7-12-2014

Imperfection-Aware Design of CNFET Digital VLSI Circuits

Abbasali Arabi Mazraehshahi

Abbasali Arabi Mazraehshahi

Candidate

Electrical and Computer Engineering

Department

This dissertation is approved, and it is acceptable in quality and form for publication:

Approved by the Dissertation Committee:

Payman Zarkesh-Ha , Chairperson

Mani Hossein-Zadeh

Zayd Leseman

James Plusquellic

Imperfection-Aware Design of CNFET Digital VLSI Circuits

by

Abbasali Arabi Mazraehshahi

B.S., Shahid Bahonar University of Kerman, 2002
M.S., Amirkabir University of Technology, 2007

DISSERTATION

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctor of Philosophy
Engineering

The University of New Mexico
Albuquerque, New Mexico

May 2014

Dedication

To my dear parents for their love

Acknowledgments

First and foremost, I would like to thank my advisor Dr. Payman Zarkesh-Ha. He is a great teacher, an excellent mentor, an intelligent researcher and most importantly trustworthy friend. I learnt a great deal from our numerous technical discussions. He is indeed the most hardworking person I have ever known and I will always admire his professionalism, enthusiasm for new ideas and dedication to research.

I would like to thank Dr. Mani Hossin-Zadeh, Dr. Zeyd Lesseman and Dr. James Plusqulic for serving on my dissertation committee.

A special thanks to all those who are responsible for the underlying academic and administrative work. Firstly, I am very grateful to Elmyra Grelle for always being so nice and helpful. I thank Edwina Chavez-Salazar, Cornelia Platero, Kyla Sorensen, Yvone Nelson, and Douglas Weintraub for their academic assistance. I am grateful to Dave Modisette, Mike Majedi, Frank Mercer, and Computer support group for technical assistance.

I would like to thank all the members of our research group at the VLSI Design Research Lab at UNM, specially Rakesh Mahto and Javad Ghasemi, for being such great friends.

I am grateful to all my friends for making my life exciting in Albuquerque. A very special thanks to Alireza Ghaffarkhah, Nima Khademi Kalantari, Mostafa Esmaeili, Behshad Roshanzadeh, Mehran Daneshbod, Hamed Nasrabadi, Mina Zare, and all the others that I may have missed.

I owe my deepest gratitude to my family, for their constant love and uninterrupted support. Especially, I thank my parents for being a driving force in pursuing my dreams.

Imperfection-Aware Design of CNFET Digital VLSI Circuits

By

Abbasali Arabi Mazraehshahi

B.S., Shahid Bahonar University of Kerman, 2002
M.S., Amirkabir University of Technology, 2007

Ph.D., University of New Mexico, 2014

ABSTRACT

Carbon nanotube field-effect transistor (CNFET) is one of the promising candidates as extensions to silicon CMOS devices. The CNFET, which is a 1-D structure with a near-ballistic transport capability, can potentially offer excellent device characteristics and order-of-magnitude better energy-delay product over standard CMOS devices. Significant challenges in CNT synthesis prevent CNFETs today from achieving such ideal benefits. CNT density variation and metallic CNTs are the dominant type of CNT variations/imperfections that cause performance variation, large static power consumption, and yield degradation.

We present an imperfection-aware design technique for CNFET digital VLSI circuits by:

1) Analytical models that are developed to analyze and quantify the effects of CNT density variation on device characteristics, gate and system levels delays. The analytical models, which were validated by comparison to real experimental/simulation data, enables us to

examine the space of CNFET combinational, sequential and memory cells circuits to minimize delay variations. Using these model, we drive CNFET processing and circuit design guidelines to manage/overcome CNT density variation.

2) Analytical models that are developed to analyze the effects of metallic CNTs on device characteristics, gate and system levels delay and power consumption. Using our presented analytical models, which are again validated by comparison with simulation data, it is shown that the static power dissipation is a more critical issue than the delay and the dynamic power of CNFET circuits in the presence of m-CNTs.

3) CNT density variation and metallic CNTs can result in functional failure of CNFET circuits. The complete and compact model for CNFET probability of failure that consider CNT density variation and m-CNTs is presented. This analytical model is applied to analyze the logical functional failures. The presented model is extended to predict opportunities and limitations of CNFET technology at today's Gigascale integration and beyond.

Table of Contents

List of Figures

List of Tables

Chapter 1

1 Introduction

1.1 Motivation

Over the past 50 years, Moore's law has been the driving force behind the semiconductor industry. Anticipating the fundamental limits of conventional complementary metal oxide semiconductor (CMOS) technology in the near future [1, 2], has forced the semiconductor industry to look for a successor technology to CMOS. Carbon nanotubes (CNTs) have drawn considerable attention due to their superior device characteristics during the past two decades. Carbon nanotube field effect transistor (CNFET) devices fabricated with ideal CNT synthesis can potentially offer more than an order of magnitude benefit in energy-delay product over Silicon CMOS. However, significant challenges in CNT synthesis prevent CNFETs today from achieving such ideal benefits [3].

1.2 Carbon nanotube basics

CNTs are made from graphene sheet, with one or more than one layers forming single-walled nanotubes (SWNTs) or multi-walled nanotubes (MWNTs), respectively [4]. CNTs have a typical diameter of 1-5 nm and can be several hundred microns long. Figure 1.1 shows a graphene sheet, when folded into a cylinder forms a SWNT.

Ballistic transport of carriers can be achieved in SWCNT because of their quasi 1-D structure, which restricts the movement of carriers only along the axis of the CNT and eliminates the wide angle scatterings of carriers. Based on their structural symmetry,

SWCNTs can be either metallic (m-CNT) or semiconducting (s-CNT) [4]. Both m-CNTs and s-CNTs have found many applications in nanoelectronics. Because m-CNTs band structure has zero or very little energy gap, m-CNTs are undesirable for transistor applications. In contrast, s-CNTs have been used to create carbon nanotube field effect transistors.

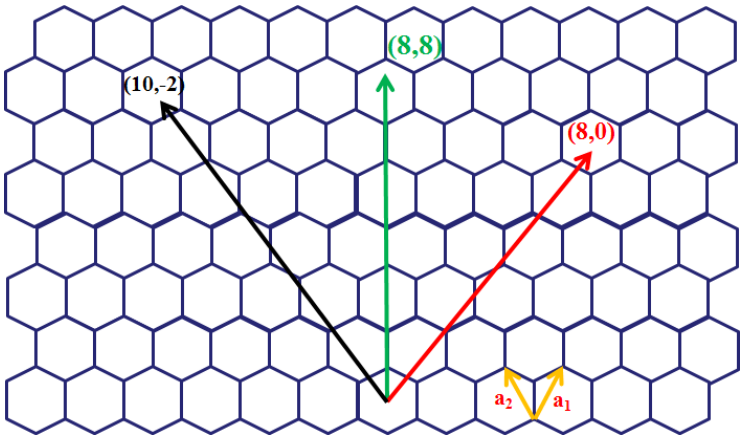


Figure 1.1. Schematic honeycomb structure of a graphene sheet. Carbon atoms are at the vertices. SWNTs can be formed by folding the sheet along lattice vectors. The two basis vectors *a1* and *a2*, and several examples of the lattice vectors are shown [4, 5].

1.3 Carbon nanotube field effect transistor

The unique electronic properties of CNFETs make them a promising candidate as an extension to CMOS devices. CNFET devices fabricated with ideal CNT synthesis can potentially provide more than an order of magnitude benefit in Energy-Delay Product (EDP) over Silicon CMOS at 16nm technology node [6, 7]. Franklin *et al.* in [8] have demonstrated a sub-10 nm CNFET, which outperforms its competing Si devices by more than four times in terms of normalized current density at low operating voltages of 0.5 V, thereby making them ideal for both high performance and low power applications. However, significant challenges in CNT synthesis prevent CNFETs from achieving such ideal benefits today [3, 5, 9].

CNFETs can be classified based on the operation of the device as, Schottky Barrier CNFET, MOSFET-like CNFET, and Band-To-Band-Tunneling CNFET [10]. In this dissertation, we consider top-gated MOSFET-like CNFETs (MOS CNFETs) [11]. For simplicity, we will refer to MOS CNFET and SWCNT as just CNFET and CNT respectively, from here on.

The major difference between a CNFET and Si CMOS is the channel material. Figure 1.2 illustrates the device structure of a CNFET with four ideal semiconducting CNT in the channel. These s-CNTs can be grown on or transferred to a substrate. An s-CNT acts as a transistor channel, which its conductivity can be modulated by a gate. The gated regions of the CNTs can be undoped, and the source and drain regions are heavily doped. By conventional lithography, the gate, source, drain contacts, and interconnects are defined, whereas the inter-CNT spacing or pitch is not limited by lithography [39].

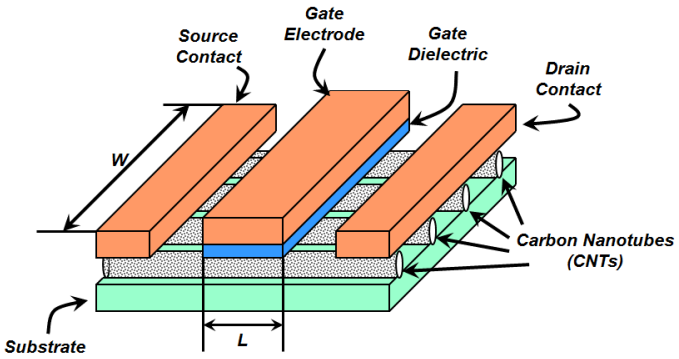


Figure 1.2. CNFET structure with four CNTs in the channel.

1.4 CNFET Technology Challenges

Form the first demonstration of single CNFET at Delft University [13] and IBM [14] in 1998, to first CNFET computer [15] at Stanford University in 2013, tremendous progress has been made in CNFET manufacturing process. First, CNFET based computer works at frequency of 1kHz with the channel length of 1µm [15].

CNFETs not only suffer from standard process variations, which are in common with CMOS technologies, but they also have their unique source of imperfections. Paul *et al.* [16] concluded that the impact of some of these sources of imperfections may be less severe for CNFETs than for silicon CMOS, because of the superior career transport and electrostatic properties of CNFETs. In addition to the above sources of variations, CNFETs are also subject to CNT-specific variations/imperfection.

1.4.1 Low CNT density

To provide sufficient drive currents, practical circuits require CNFETs with multiple CNTs in their channel [14]. In order to achieve optimal energy-delay tradeoffs, CNFETs with CNT density of about 250 CNTs/μm are needed [17]. Multiple-growth [18] and multiple-transfer [19] techniques have been demonstrated to improve the CNT density. Experimental results demonstrate that CNT density can be improved from 2-8 CNTs/μm, and 20-45 CNTs/μm for multiple-transfer and multiple-growth techniques respectively.

1.4.2 Mispositioned CNTs

A large fraction (99.5%) of the CNTs grown on single-crystal quartz substrates normally grow aligned *i.e.* straight and parallel to each other [20]. A non-negligible fraction of misaligned CNTs may pass through a layout region where a CNT was not intended to pass. In the extreme cases, mispositioned CNTs can cause functional failures of logic gates [21]. Patil *et al.* developed layout techniques to create CNFET circuits immune to mispositioned CNTs [21].

1.4.3 CNT diameter variation

Based on different CNT growth techniques CNT diameter varies from 0.5 to 3.5 nm. Since CNTs' bandgap is inversely proportional to the diameter of them, this variation affects the

device parameters, which are made out of CNTs [4]. Normally, a CNFET containing a single CNT in its channel is highly sensitive to CNT diameter variation [16, 22]. However, in practical circuits with multiple CNTs in their channel, due to statistical averaging, the impact of CNT diameter variation can be significantly reduced [16].

1.4.4 Schottky Barrier Contact

The interface between the carbon nanotubes and metals that are used as source/drain of a CNFET forms a Schottky Barrier (SB) [23-26]. The SBs at the source and drain side of transistors result in a significant reduction in the drain current in the transistors. Therefore, for a high performance operation of the CNFET devices, suitable metals are required, which can be used as source and drain contacts and also provide ohmic source and drain contacts [27].

1.4.5 CNT Density Variation

The process of CNT growth always results in random and undetermined CNT placement. Inability of growing perfectly aligned and uniformly distributed CNTs that impose a fundamental limit on current CNFET technology, have been widely observed in published results for CNT growth [20, 28].

1.4.6 Metallic CNTs

Depending on the chirality of the carbon nanotube, the growing CNT can either be metallic or semiconductor. Since the conductivity of m-CNTs cannot be controlled by the gate, they cannot be used to make CNFETs. Therefore, a CNFET made entirely from m-CNTs will have a current that is only dependent on the drain–source voltage, not on the gate voltage. m-CNTs can cause significant problems in CNFET circuits [27, 35-37].

Significant works have been done to overcome the non-ideality of CNT synthesis processes. Still, the current CNT synthesis processes are far from ideal to utilize the benefits of this revolutionary material. The impact of unique source of CNFET imperfections have been investigated and analyzed by different research groups. The impact of CNT diameter variations for CNFET that contains only one CNT compared to the conventional CMOS variations such as channel length and oxide thickness was reported by [16]. Due to good electrostatic control and near ballistic transport in CNFETs, Paul *et al.* concluded that CNFETs are generally less sensitive to the conventional CMOS variations.

CNT density variations and metallic-CNT-induced variations are the main sources of variations/imperfections in CNFET technology [9, 35-44]. In Figure 1.3 Zhang *et al.* [41] compared the variations in ON-state current of CNFET caused by each of the aforementioned sources of CNT-specific variations for a minimum-width CNFET at 32nm technology. Normally, a CNFET containing a single CNT in its channel is highly sensitive to CNT diameter variations [16]. However practical circuits require CNFETs with multiple CNTs in their channel [45]. Because of statistical averaging the impact of CNT diameter, doping and alignment variations is significantly reduced for CNFETs with multiple CNTs under the gate [17, 41, 46-47]. As shown in the figure, CNT density variation and metallic-CNT-induced variation contribute most significantly to the overall CNFET ON-state current variations.

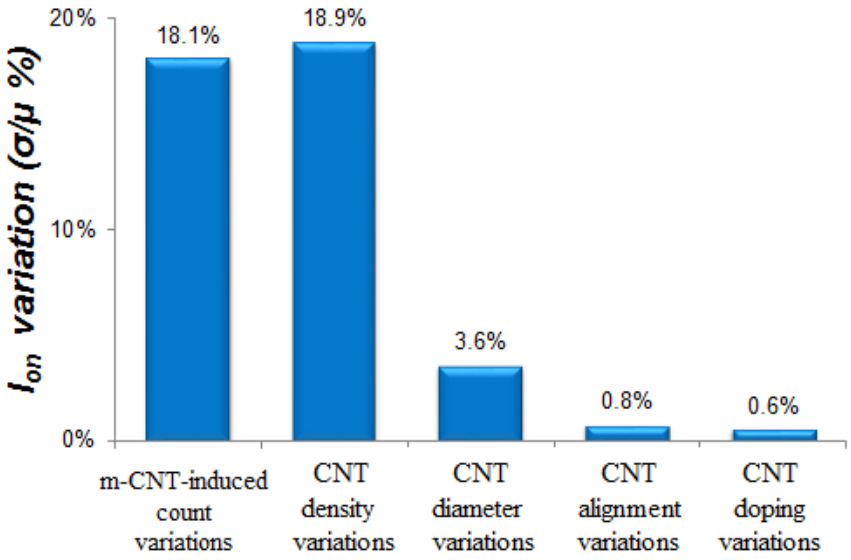


Figure 1.3. Simulated σ(Ion)/μ(Ion) of a minimum-width CNFET at 32nm technology node (using CNFET device models in [3]), contributed by various sources of CNT-specific variations[41].

1.5 Contributions

This dissertation presents an imperfection-aware design of CNFET digital VLSI circuits. The major contributions of this dissertation are listed below:

1. Analytical models are developed to analyze and quantify the effects of CNT density variation on device characteristics, gate and system levels delays. The analytical models enable us to examine the space of CNFET combinational, sequential, and memory circuits to minimize delay variations. Using these models, we drive CNFET circuit design guidelines to manage/overcome CNT density variations.
2. Analytical models are also developed to analyze the effects of m-CNTs on device characteristics, gate, and system levels delay and power consumption. Using these model, we also drive CNFET design processing guidelines.
3. CNT density variation and m-CNTs can result in functional failure of CNFET circuits. The complete and compact model for CNFET probability of failure that consider CNT density variations and m-CNTs is presented. This analytical model

is applied to analyze the logical functional failures. The presented model is extended to predict opportunities and limitations of CNFET technology at today's Gigascale integration and beyond, assuming that the CNT placement during the growth process still stays in a random and non-uniform fashion that we have today.

1.6 Existing Research Topics

The scope of this dissertation is closely related to the following existing research topics.

1.6.1 Circuit analysis and design space for overcoming variations in CNFET circuits

A) ***CNT density variation***

1. Deng *et al.* [17] considered m-CNT-induced variations by assuming an ideal scenario that all m-CNTs are removed and all s-CNTs are kept intact during a removal process, CNT density variation has not been considered.
2. [46, 47] did not consider CNT density variations in their quantified variation analysis.
3. For the first time, Zhang *et al.* [41] quantified the impact of CNT variations on delay variations of CNFET. In [41], Zhang adapted an existing Monte Carlo-based statistical static timing analysis approach by using a variation-aware timing model [9] for CNFET logic gates based on the CNFET device model and CNT count variations model [38]. Zhang *et al.* in [38] presents parameterized model based on the probability mass function (PMF), which is extracted numerically from experimental results for CNT density variations. It is shown in [38] that for large size CNFET, the CNT count distribution can be approximated by a Gaussian distribution. However, the model is not analytical and it depends on the CNT spacing distribution that has to be extracted from experimental CNT growth.

B) ***Metallic CNT***

1. In [39] a probabilistic model which incorporates processing and design parameters and enables quantitative analysis of the impact of m-CNTs on leakage, noise margin, and delay variations of CNFET-based digital logic circuits has been presented.
2. Ashraf *et al.* in [27, 48] presented analytical models to estimate the functional yield of logic gates designed using different configurations of CNFET.

Although these two works presented some good models that describe the effect of m-CNTs on circuit parameters, there is still no simple physical model that can predict the system-level requirements for acceptable range of m-CNTs.

1.6.2 Imperfection-immune CNFET circuit analysis to overcome CNT density variation and m-CNTs

Transistor-level redundancy is used in CNFET devices to enhance the transistor reliability. The probability of short defects for CNFET has been discussed in many previous publications [35-37, 49-51]. Based on the probability of short defects, Lin *et al.* have presented a novel metallic-CNT tolerant CNFET using asymmetric correlated CNTs (ACCNT) that use uncorrelated stacks of series CNFETs to tolerate short defects caused by m-CNTs and correlated branches of parallel CNFETs to increase the device drive strength without degrading the failure rate [35-37]. There is still no complete and compact model for CNFET probability of failure that includes 1) the probability of open defects (void CNFET), 2) the probability of short defects (due to m-CNT), and 3) the CNT count density variations.

1.7 Organization of the Dissertation

The rest of the dissertation is organized as follows:

Chapter 2 presents an analytical model, based on binomial probability distribution. The model then is validated using experimental/simulated data. It will be used to analyze and quantify the effects of CNT density variation on device, gate, and system levels of CNFET VLSI systems.

Chapter 3 presents analytical models, which was validated using simulated data, to analyze and quantify the effects of m-CNTs on device, gate, and system levels of CNFET VLSI systems.

Chapter 4 presents an integrated and compact model for probability of failure in CNFETs that includes CNT density variation and m-CNTs.

Chapter 5 concludes the dissertation by highlighting the contribution of this research as well as presents some possible extensions of this work.

Chapter 2

2 CNT Density Variation

CNT density variation is one the major sources of imperfection in CNFET VLSI circuits. Since this imperfection plays an important role in the performance and robustness of CNFET circuits, accurate analysis and characterization of CNFET technology is essential. In this chapter, based on binomial probability distribution, we proposed analytical models, validated using experimental/simulated data, to analyze and quantify the effects of CNT density variation on device, gate, and system levels of CNFET VLSI systems. Using these models, we derived simple, yet useful CNFET processing and circuit design guidelines to manage CNT density variation. Later in chapter 4, we use these models as a great tool to predict the limitations and opportunities for a robust and reliable CNFET system in the presence of CNT density variation.

2.1 Introduction

Unlike in the standard CMOS process, where each mask layer is aligned precisely, the process of CNT growth always results in random and undetermined CNT placement. Figure 2.1 illustrates three CNFETs located on randomly placed CNTs. Depending on the location of CNFET, a random number of CNTs exists under the gate "window." For instance, CNFET-c has 4 CNTs in the channel, and CNFET-a has only 1 CNT, and CNFET-b has none. We refer to a CNFET without a CNT, which results in an open defect, as a "void CNFET," whereas a CNFET with at least one CNT in the channel is referred to

as a “live CNFET.” A live CNFET can be a “functional device” if there is no m-CNT in the channel; otherwise, a live CNFET will become in a short defect [52].

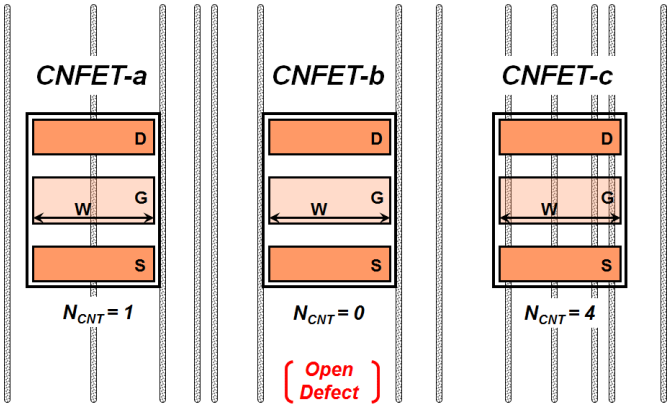


Figure 2.1. CNFETs on randomly placed CNTs. CNFET-a has only one CNT, CNFET-b has no CNT, and CNFET-c has four CNTs. CNFET-b is a void CNFET, which is always open [52].

Figure 2.1 intuitively shows that by choosing too small of a CNFET width, *W*, the chance of having an open defect increases, whereas selecting too large of a CNFET width results in more CNFETs and therefore a greater likelihood of short defects.

Because the number of CNTs dictates the CNFET device parameters, such as ON-current, OFF-current, threshold voltage, and the average gate capacitance, for characterizing CNFET systems in the presence of CNT density variation, it is vital to find the distribution of CNTs.

Experimental extraction of CNT distribution can be time consuming because such a process depends on the width of CNFET [9]. A parameterized model for CNT density variation, based on experimental data, has been developed by Zhang *et al.* [38, 53]. Although the model offers a valuable insight into the CNT density variation issue, it is based on the probability mass function that is numerically extracted from experimental results, which is not easily reproducible. It is shown in [38] that for a large size CNFET, the CNT count distribution can be approximated by a Gaussian distribution. However,

again, the model depends on the CNT spacing distribution that must be extracted from an experimental CNT growth. In this section, we derive a model that represents the distribution of the number of randomly placed CNTs in the channel of CNFETs.

2.1.1 Derivation of CNT density distribution

Suppose that in a window size of W, we try n coin-flipping experiments for CNT placement. The probability of having k CNTs, $f_B(k)$, in this window size of W in this experiment follows the standard binomial distribution:

$$f_B(k) = \binom{n}{k} p^k (1-p)^{n-k} , \tag{2-1}$$

where p=0.5 and

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} . \tag{2-2}$$

The probability distribution of CNT count therefore is simplified to:

$$f_{CNT}(k) = \frac{n!}{k!(n-k)!}(0.5)^k(0.5)^{n-k} = \frac{n!}{k!(n-k)!}(0.5)^n . \tag{2-3}$$

To find n in (2-3), we consider the fact that the average CNT counts from (2-3), $\mu = n \times p = n \times 0.5$, must be equal to the expected number of CNTs in our experiment, i.e., $\bar{N}_{CNT} = W.D$; therefore,

$$\mu = \bar{N}_{CNT} \quad \Rightarrow \quad n = 2\bar{N}_{CNT} = 2W.D . \tag{2-4}$$

Substituting (2-4) into (2-3) gives the CNT count probability density distribution:

$$f_{CNT}(k) = \frac{(2\bar{N}_{CNT})!}{k!(2\bar{N}_{CNT}-k)!}(0.5)^{2\bar{N}_{CNT}} , \tag{2-5}$$

where $f_{CNT}(k)$ is the probability of having k CNTs under the CNFET gate, and $\bar{N}_{CNT}$ is the expected number of CNTs, which is equal to $W \cdot D$. An example of a CNT count distribution using (2-5) is shown in Figure 2.2 for $\bar{N}_{CNT} = 3$, which can represent a CNFET with either

a 0.3 µm width and density of 10 CNT/µm, or a CNFET with a width 30 nm and density of 100 CNT/µm.

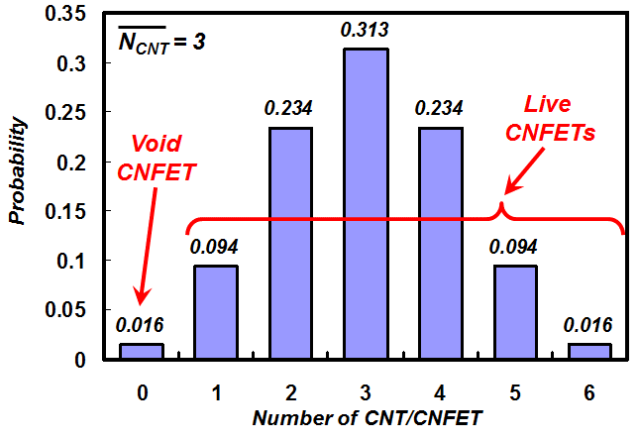


Figure 2.2. Analytical CNT count probability distribution using binomial distribution in (2-5) when the expected number of CNTs is set to be 3.

From the binomial distribution in (2-5), the standard deviation of CNT count distribution can be determined by:

$$\sigma = \sqrt{np(1-p)} \quad \Rightarrow \quad \sigma_{CNT} = \sqrt{\frac{\bar{N}_{CNT}}{2}}\,. \tag{2-6}$$

By using (2-6), the coefficient of variation is equal:

$$\left(\frac{\sigma}{\mu}\right)_{CNT} = \frac{\sqrt{\frac{\bar{N}_{CNT}}{2}}}{\bar{N}_{CNT}} = \frac{1}{\sqrt{2\bar{N}_{CNT}}}\,, \tag{2-7}$$

The index of dispersion (IDC) is defined as the ratio of the variance to the mean of a random variable [9]. Therefore, the index of dispersion of CNT count distribution is:

$$\left(\frac{\sigma^2}{\mu}\right)_{CNT} = \frac{\frac{\bar{N}_{CNT}}{2}}{\bar{N}_{CNT}} = 0.5\,. \tag{2-8}$$

In general, even though k is always an integer number, $\bar{N}_{CNT}$ may become a non-integer value, depending on the values of W and D. In this case, the continuous version of equation (2-5) can be used based on Gamma function, as:

$$f_{CNT}(k) = \frac{\Gamma(2\bar{N}_{CNT}+1)}{k!\cdot\Gamma(2\bar{N}_{CNT}-k+1)}(0.5)^{2\bar{N}_{CNT}}, \tag{2-9}$$

where

$$\Gamma(x) = \int_0^\infty t^{x-1}\, e^{-t} dt\ . \tag{2-10}$$

Note that the CNT count probability distribution in (2-5) when k=0 gives the probability of a void CNFET, or an open defect, P_O,

$$P_O = (0.5)^{2\bar{N}_{CNT}}\ . \tag{2-11}$$

Equation (2-11) provides a *fundamental limit* on probability of open defects in CNFET as a function of expected number of CNTs [52]. Figure 2.3 validates our analytical model for CNT count probability distribution by its precise correlation with two sets of experimental data extracted from [37, 38].

To analyze the non-negative integer random variable, *IDC* is a useful tool to describe the dispersion or variability [9]. Equation (2-8) shows that the *IDC* of our analytical model is always equal to 0.5 and independent of experimental data. This has been confirmed by experimental data reported by other researcher too. For example Zhang *et al.* reported the *IDC* of 0.5 in their approximated Gaussian model based on experimental data [9]. Equation (2-8) validates our analytical model for CNT count probability distribution by its precise correlation with the Stanford experimental-based model. It also confirms our prime assumptions *i.e.* at any point, the chance of having a CNT, p, is 50% that we made to drive this analytical formula.

2.2 CNT density variation impacts on CNFET devices

In this section, we analyze the impact of CNT density variation on CNFET device characterizations. The device-level model for CNFET used in this sections is based on a CNFET SPICE model [3] evaluated at the 32 nm technology node. The supply voltage for

CNFET is assumed to be 0.9V. The CNT density variation in (2-5) is used to implement CNFET variations in SPICE.

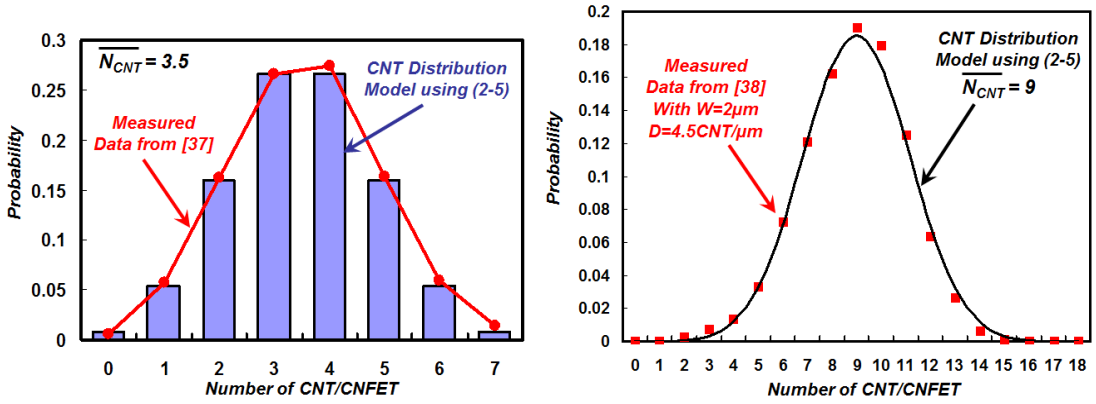


Figure 2.3. Analytical model of CNT count probability distribution versus experimental data from [37, 38].

2.2.1 ON-state current, OFF-state current and device parasitic capacitance

ON-state and OFF-state currents, I_{ON} and I_{OFF}, often impose critical constraints in VLSI design. A higher I_{ON} is required to get a better performance, while a lower I_{OFF} is desirable to get lower leakage power. Also device parasitic capacitance, C_{CNFET}, plays an important role in circuit performance, and therefore variation of device capacitance can drastically affect the system performance fluctuation.

I_{ON} and I_{OFF} are defined as:

I_{ON}: The I_{ON} for 32 nm N-type (P-type) CNFET is defined as the value of the drain current (I_D) at V_{GS} = 0.9V (V_{GS} = -0.9V) and V_{DS} = 0.9 V (V_{DS} = -0.9V).

I_{OFF}: The I_{OFF} for 32 nm N-type (P-type) CNFET is defined as the value of the drain current (I_D) at V_{GS} = 0V (V_{GS} = 0 V) and V_{DS} = 0.9V (V_{DS} = -0.9V).

Figure 2.4 shows the SPICE simulation results for I_{DS}–V_{GS} characteristics of 32-nm N-type CNFETs with expected 5 CNTs in the channel, where the number of CNTs varies from 1

to 10. The solid red line in the middle represents the nominal case (μ=5), and the dashed lines are for devices with various number of CNT.

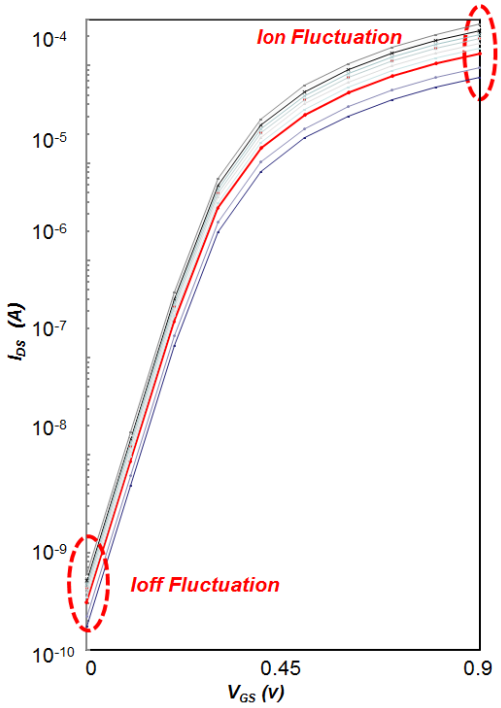


Figure 2.4. DC characteristic fluctuations of 32nm N-type CNFET due to CNT density variation. I_{DS}-V_{GS} characteristics fluctuations for expected 5 CNTs in the channel using SPICE simulation [54].

For a CNFET with *N* CNTs under the gate, the $I_{ON,N}$, $I_{OFF,N}$, and $C_{CNFET,N}$ can be expressed as[3, 45]:

$$I_{ON,N} \approx N \times I_{ON,1}, \tag{2-12}$$

$$I_{OFF,N} \approx N \times I_{OFF,1}, \tag{2-13}$$

$$C_{CNFET,N} \approx N \times C_{CNFET,1}, \tag{2-14}$$

where $I_{ON,1}$, $I_{OFF,1}$, and $C_{CNFET,1}$ are the ON-state current, OFF-state current, and the gate capacitance of a CNFET with 1 CNT, respectively.

Equations (2-12), (2-13), and (2-14) show that, ON-state current, OFF-state current, and the gate capacitance of a CNFET, in the presence of CNT density variation depend on random variable *N*. The coefficient of variation, σ/μ, of $I_{ON,N}$, $I_{OFF,N}$, and $C_{CNFET,N}$ can be

calculated by (2-8). As a result (2-8) can be used as a guideline to choose a CNFET that meets a specific constraint. For example, to obtain a CNFET with less than 20% of I_{ON}, I_{OFF} and C_{CNFET} variations, there must be at least 13 CNTs in the CNFET [54], because

$$\left(\frac{\sigma}{\mu}\right)_{I_{ON}} = \frac{\sqrt{\frac{\overline{N}_{CNT}}{2}}}{\overline{N}_{CNT}} = 0.2 \quad \Rightarrow \overline{N}_{CNT} > 13 \,.$$

Table 2.1: The percentage of I_{ON}, I_{OFF} and C_{CNFET} variations, analytical model versus SPICE simulation [54].

NCNT	SPICE Simulation (σ/μ)%			Analytical Model (σ/μ)%
	I_{ON}	I_{OFF}	C_{GATE}	I_{ON} / I_{OFF} / C_{CNFET}
5	43.5	46.2	45.8	44.72
10	33.62	30.9	34.1	31.62
50	14.2	14.6	14.0	14.14
100	9.99	10.1	10	10.00
200	7.1	7.0	7.0	7.07

Table 2.1 shows the percentage of variation of I_{ON}, I_{OFF} and C_{CNFET} using analytical model versus extracted data from SPICE simulation using Monte Carlo analysis. As shown the predicted percentage of variation matches very well with the SPICE simulation results.

2.2.2 Threshold voltage

Threshold voltage fluctuation is the source of variation in performance, delay, and leakage power in Gigascale integrated systems. A drastic change in the threshold voltage can lead to a large short-circuit current, large leakage power, and poor performance. We use SPICE Monte-Carlo simulations to determine threshold voltage variation in CNFETs. The threshold voltage is measured from the device I-V characteristics using the linear extrapolation technique [55]. Based on measured data, the maximum percentage of threshold voltage variation in CNEFT due to CNT density variation is about 0.01%, which

is negligible. Also, the percentage of threshold voltage variation is independent of number of CNTs, which is a great benefit for CNFET technology [16, 54].

2.3 Impact of CNT density variation on CNFET gate delay

After device level, the next level to investigate the impact of CNT density variation on Gigascale integrated system is the CNFET gate level. In the conventional design flow, static timing analysis (STA) is used to estimate the circuit delay. The need for an effective modeling of process variations in timing analysis has led to extensive research in statistical STA (SSTA). In both STA and SSTA techniques the delay characterization for cell libraries should be clearly defined [56]. As shown in section 2.2, the CNT density variation severely changes the individual device parameters such as I_{ON}, I_{OFF} and C_{CNFET}. Therefore, accurate characterization of each gate in the presence of CNT density variation is vital for future CNFET technology.

Figure 2.5 illustrates the effects of CNT density variation on each gate delay and consequently on circuit delay using SPICE simulations [57]. This figure confirms that the gate delay distribution depends on the number of CNTs under the gate. For example since G1 has more CNTs under the gate compared to G2, the percentage of gate delay variation in G1 is smaller than in G2. Also, due to few CNTs under the G3, the percentage of path delay variation from its inputs to its output of G3 is worse.

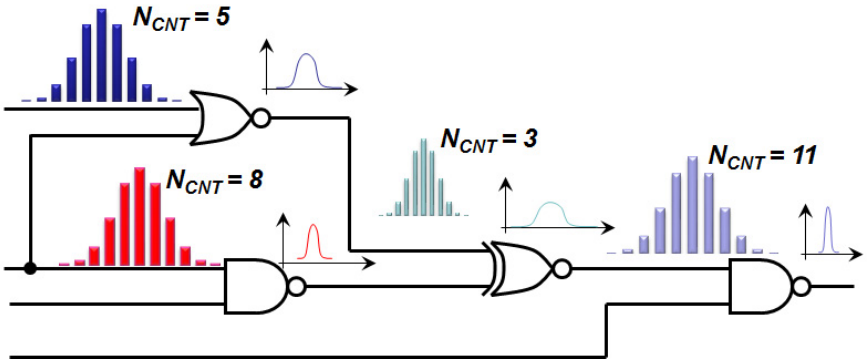


Figure 2.5. Simulated effects of CNT density variation on gate and circuit delay [57].

2.3.1 Transient response model approach

Although there are different ways to calculate gate delay such as transient response model, RC delay model, and linear delay model [58, 59], in this section we will use transient response model to predict the gate-delay variation due to CNT density variation.

Propagation delay time, t_{pd}, is defined as the time delay from the input voltage crossing 50% to the output voltage crossing 50% [58]. The propagation delay of a logic gate is determined by the current that it can deliver and the capacitance that it drives. It has been shown that the propagation delay of a logic gate, t_{pd}, can be estimated from [58]:

$$t_{pd} = \frac{C_{Load} \times VDD/2}{I_{Drive}}, \tag{2-15}$$

where C_{Load} is the load capacitance, I_{Drive} is the CNFET drive current, and V_{DD} is the power supply voltage. For a CNFET with N CNTs under the gate, the drive current $I_{ON,N}$ can be expressed as[3, 45]

$$I_{Drive} = I_{ON,N} \approx N \times I_{ON,1} \tag{2-16}$$

where $I_{ON,1}$ is the ON current of a CNFET with a single CNT.

2.3.1.1 Mapping the CNT density variation to the gate delay variation

As a baseline, we first consider an inverter driving a fixed load capacitance. We assume that the self-loading capacitance is negligible in comparison with the large load

capacitance, C_L. Substituting (2-15) into (2-16) gives the propagation delay of an inverter to drive a fixed load capacitance as a function of the number of CNTs.

$$t_{pd} \cong \frac{(C_L)\times VDD/2}{I_{ON,N}}, \tag{2-17}$$

where $I_{ON,N}$ is N times $I_{ON,1}$

$$t_{pd} \cong \frac{(C_L)\times VDD/2}{N\times(I_{ON,1})} = \frac{K}{N}. \tag{2-18}$$

The only random variable in (2-18) is N, which represents the number of CNTs in the CNFET gate. All other parameters are constant and therefore can be represented by a constant parameter, K.

As shown in section 2.1, depending on the location of the CNFET, there is a random number of CNTs under the gate "window" which drastically can change the transistor's ON-current and consequently gate delay. Figure 2.6 shows the mapping of the CNT density function using Equation (2-18) to the propagation gate delay function. The probability that the number of CNTs under the gate is within some interval $N_{CNT1} \leq \overline{N}_{CNT} \leq N_{CNT2}$ is equal to the probability that the gate delay is within the interval $K/N_{CNT2} \leq t_{pd} \leq K / N_{CNT1}$.

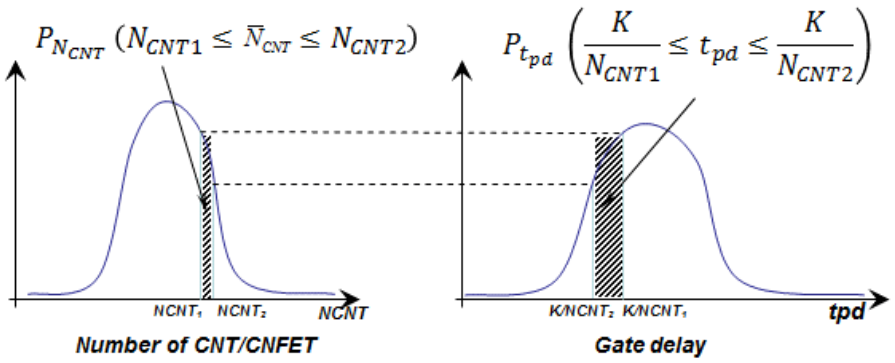


Figure 2.6. Mapping the distribution of number of CNTs per CNFET to the gate delay distribution [57].

To drive an analytical relationship between the number of CNTs per CNFET and gate delay distribution, let's assume that X is a random variable with the mean value of μ_x and the

standard deviation of σ_x. Generally, the distribution of random variable *Y=1/X* is not necessarily normal [60], however if the variation is small ($\sigma_x/\mu_x<<1$), then *Y=1/X* can be well approximated by a normal distribution [61]. It has been shown in [61] that for *Y=1/X*, we have:

$$\mu_Y = \frac{1}{\mu_X}. \tag{2-19}$$

To find standard deviation for *Y*, when the standard deviation is small, we take derivative of Y with respect to X as follow:

$$\frac{dY}{dX} = \frac{-1}{X^2} \quad \Rightarrow \quad dY = \frac{-dX}{X^2}. \tag{2-20}$$

Considering that the derivative can be represented by small changes, (2-20) can be rewritten as:

$$\Delta Y \approx \frac{-\Delta X}{X^2}. \tag{2-21}$$

Assuming that the standard deviation compared to the mean value is very small ($\sigma/\mu<<1$) Δ can be substitute with σ in (2-21), which yields

$$\sigma_Y = \frac{\sigma_X}{\mu_X{}^2}. \tag{2-22}$$

To illustrate the accuracy of (2-19) and (2-22), an example of a random variable which can represent the numbers of CNTs under the CNFET gates with mean (μ_X=80), and standard deviation (σ_X=6.32) is calculated form (2-6) and is shown in Figure 2.7-a. Figure 2.7-b shows the mean and standard deviation of Y=1/X using MATLAB simulation [57].

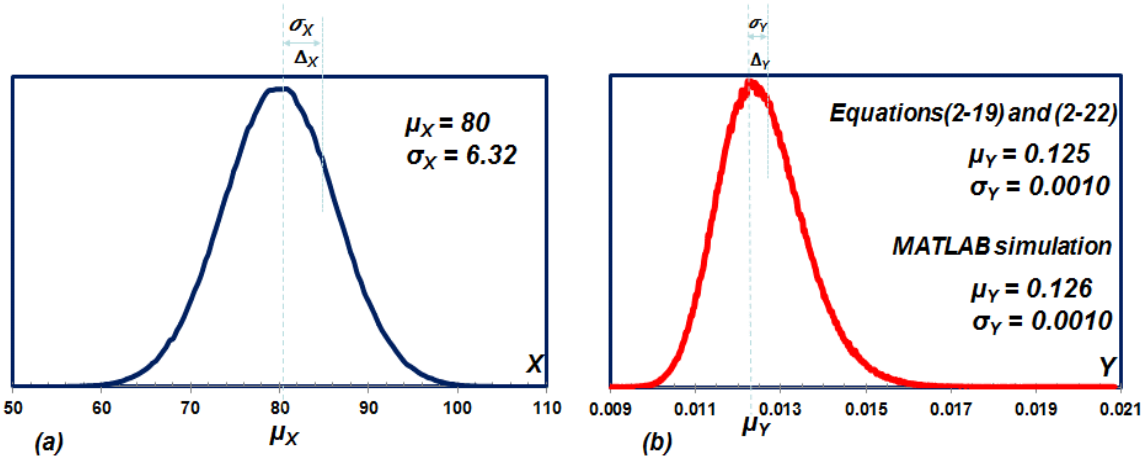


Figure 2.7. Mean and standard deviation, analytical model verification versus MATLAB data [57].

Using (2-22) the standard deviation for propagation delay can be determined by:

$$\sigma\, t_{pd} \approx \mathrm{K}\frac{\sigma_{\bar{N}}}{(\bar{N})^2} \tag{2-23}$$

where $\bar{N}$ is expected number of CNTs under the gate. To determine the percentage of delay variation, using (2-19) and (2-23) gives:

$$(\sigma/\mu)_{t_{pd}} \approx \left(\frac{K\frac{\sigma_{\bar{N}}}{(\bar{N})^2}}{K\frac{1}{\bar{N}}}\right) = \left(\frac{\sigma_{\bar{N}}}{\bar{N}}\right) = \left(\sqrt{2\,\bar{N}}\right)^{-1} \tag{2-24}$$

The device-level model for CNFET used in the this section is based on the CNFET SPICE model developed by Deng *et al.* for a minimum-width CNFET at 32nm technology node [3]. The CNT density variation in (2-5) is used to implement CNFET variations in SPICE. Monte Carlo simulations were performed to evaluate the effect of variations of $\bar{N}_{CNT}$ on the mean and standard deviation of delay characteristics of logic gates for several gates including: INV1X, INV4X, NAND2-1X, NOR2-1X, NAND3-1X, and NOR3-1X. It is assumed that each gate drives a 200fF fixed load. In these simulations, 10 different samples of $\bar{N}_{CNT}$within the range 20-250 CNTs under the gate were considered. The ON-current depends on the actual "effective" gate width which is determined by the number of CNTs

under the gate and the spacing between them [62]. The supply voltage in accordance with the ITRS roadmap for 32nm technology is 0.9V [63]. 1000 samples were taken into consideration in our simulations.

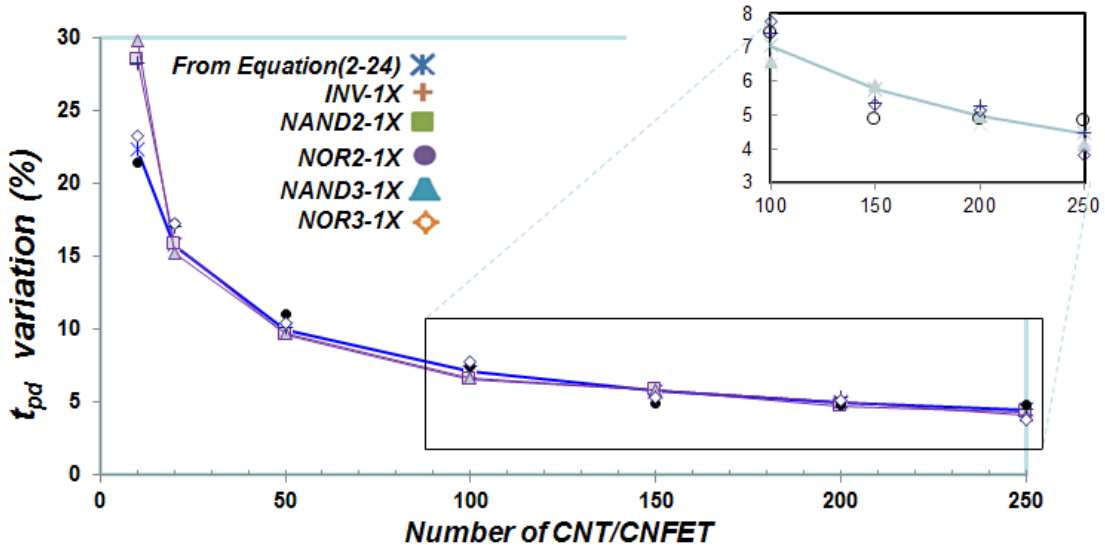


Figure 2.8. Predication of gate delay variation versus simulated data for INV1X, NAND2-1X, NOR2-1X, NAND3-1X, and NOR3-1X CNFET gates to drive a 200fF constant load [57].

Figure 2.8 show the percentage of gate delay prediction using (2-24) versus simulated data for different logic gates. The results validate the accuracy of our analytical model for prediction the percentage of logic gates delay by its precise correlation with percentage of variation for CNT density distribution. Equation (2-24) predicts the gate delay variation very accurately, when the average number of CNTs under the gate are larger than 50, which is a typical range [45].

Table 2.2 shows the average and standard deviation of the gate delay using analytical model versus simulated data extracted from SPICE simulation on a 1X inverter using Monte Carlo simulation. As shown the predicted delay variation matches very well with the SPICE simulation results.

Table 2.2: Analytical model verification versus SPICE simulation, assuming I_{on}=17µA, C_{LOAD}=200fF, V=0.9v [57]

NCNT	Analytical Model			SPICE Simulation		
	µ(pS)	σ(pS)	σ/µ(%)	µ(pS)	σ(pS)	σ/µ(%)
10	523	117	22.3	554	156	28.2
20	261	41.37	15.8	264	44.8	17
50	104	10.47	10	104	10.5	10.1
100	52	3.70	7.07	51	3.83	7.44
150	34	2.01	5.77	34	1.86	5.35
200	26	1.31	4.99	26	1.37	5.28
250	20	0.94	4.47	20	0.92	4.5

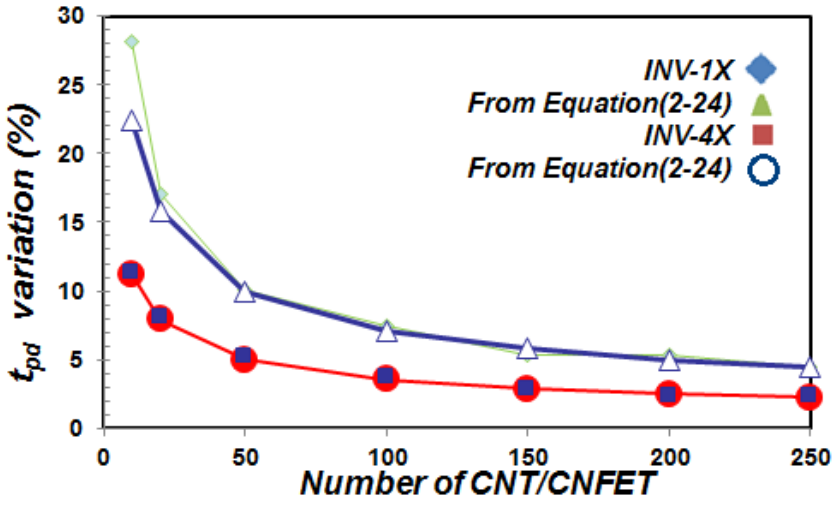


Figure 2.9. Prediction of gate delay variation versus simulated data for INV1X and INV4X to drive a 200fF constant load [57].

Percentage of gate delay variation versus simulated data for INV1X and INV4X driving a 200fF load capacitance is shown in Figure 2.9. Simulated result shows that the percentage of delay variation in INV1X is two times more than INV4X which is expected based on (2-24). Although $\frac{(C_L)\times VDD/2}{(I_{ON,1})}$ which was denoted as *K* in (2-18), is a technology dependent parameter, (2-24) shows the percentage of delay variation is independent of technology

node. In other words, irrespective of the technology node, the number of CNTs under the gate "window" plays a vital role in delay variation which depends on CNT density variation [57].

2.3.2 RC delay model approach

To drive an analytical formula that considers all aspects and parameters of an actual design, our previous approach, which was based on transient response model and Equation (2-15) was not successful. However, the assumption of I_{Drive} to be equal to I_{ON} works for simple gates. The difference between I_{Drive} and I_{ON} to calculate the complex gate delay in the presence of self-loading and interconnect capacitance, forced us to use the RC delay model and well-known Elmore delay technique to analyze the effect of CNT density variation on CNEFT gates delay.

The RC delay model treats a transistor as a switch in series with a resistor. This approximation works remarkably well for delay estimation. The effective resistance is the ratio of V_{DS} to I_{DS} averaged across the switching interval of interest [58]. A unit CNFET that contains one CNT under the gate is defined to have effective resistance R_{ON}. A CNFET with N CNTs under the gate has resistance of $\frac{R_{ON}}{N}$, because it delivers N times as much current. As shown in Figure 2.10, each N-type and P-type CNFET has gate and self-loading capacitance. We define C_{IN} and C_{SL} to be the gate capacitance and self-loading of a unit transistor, respectively. C_{SL} and C_{IN} depend on the number of CNTs under the CNFET channel. Hence, a CNFET with N CNTs under the gate has an input capacitance of $N \times C_{IN}$, a self-loading capacitance of $N \times C_{SL}$ and a resistance of $\frac{R_{ON}}{N}$ as modeled and depicted in Figure 2.10.

Because the RC delay model treats a transistor as a switch in series with a resistor, any complex gate can be model as a switch in series with a resistor, therefore, there is no limitation for modeling combinational or sequential circuits [58]. Without loss of generality, the RC model of inverter and transmission gates have been shown in Figures 2.11-a and 2.11-b, respectively. The unit inverter of Figure 2.11-a is composed of an N-type CNFET and a P-type CNFET of unit size to achieve equal rise and fall resistance. The effective resistance of the transmission gate is modeled as a parallel connection of the resistances R_{ON}-n and R_{ON}-p of the N-type and P-type CNFET devices [58, 59].The simulated value of R_{TG}= R_{ON}-n || R_{ON}-p as a function of V_{out} is relatively constant[59].

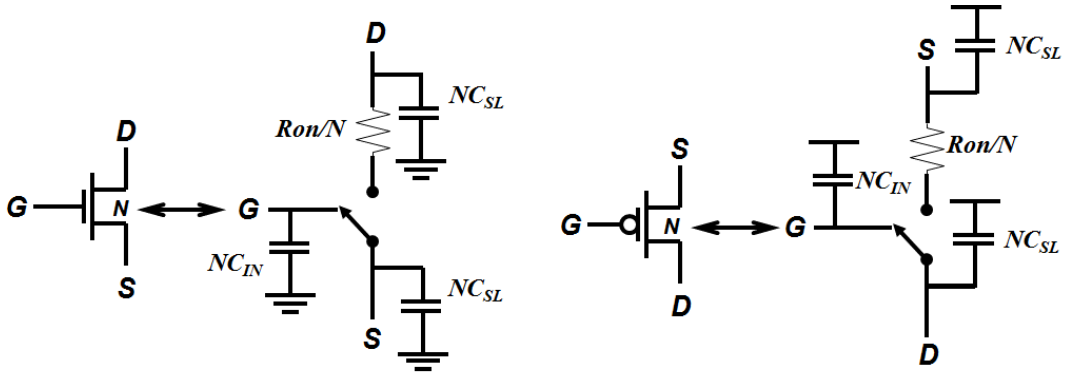


Figure 2.10. Equivalent RC model circuits for N-type and P-type CNFET

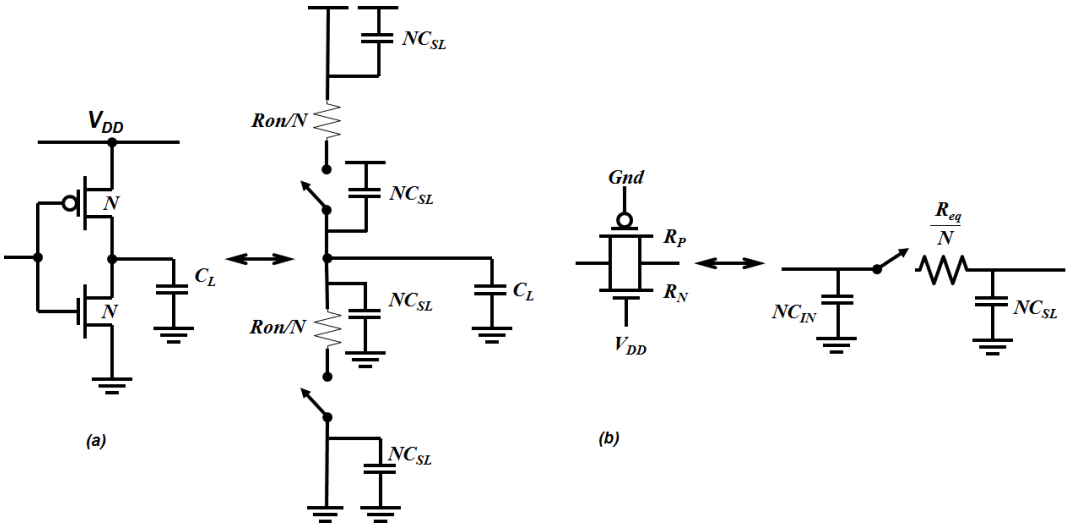


Figure 2.11. Equivalent RC circuit for a) an inverter, b) a transmission gate

To drive a general formula that consider all aspects and parameters of actual design, as a baseline, we first consider a gate, which has been modeled as an RC switch that does not drive any external node, as shown in Figure 2.12-a.

The propagation delay of gate as a function of the number of CNTs is:

$$t_{pd} = \ln 2 \times \left(\frac{R_{ON}}{N}\right) \times (N \times C_{SL}) = \ln 2 \times R_{ON} \times C_{SL} \ . \qquad (2\text{-}25)$$

Equation (2-25) shows that in this case, the delay propagation is constant and independent of the number of CNTs under the gate channel. SPICE simulation also confirms the accuracy of equation (2-25). Because the standard deviation of a constant value is zero, the percentage of delay variation due to CNT density variation is 0%. Table 2.3 shows the average and standard deviation of the gate delay using analytical model versus simulated data extracted from SPICE simulation on a 1X inverter using Monte Carlo simulation.

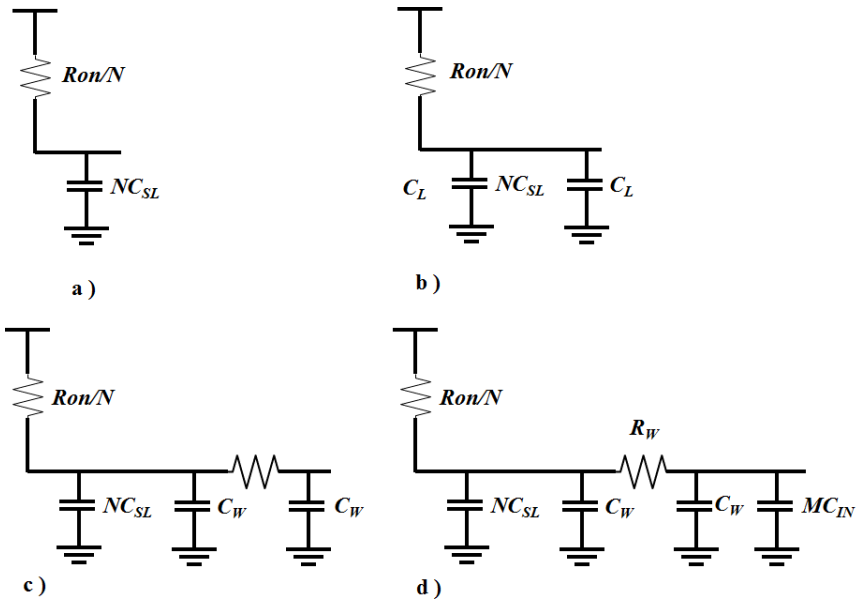


Figure 2.12. Modeling of a combinational circuit as a , a)RC switch, b) RC switch driving a constant load, c) RC switch along a wire, and d) RC switch driving another gate(s) along the wire.

For the second stage, to drive the general formula, a RC switch that drives a fixed load capacitance with negligible wire capacitance has been considered and is shown in Figure 2.12-b. Equation (2-26) gives the delay propagation as a function of the number of CNTs,

$$t_{pd} = \ln 2 \times \left[R_{ON} \times C_{SL} + \frac{R_{ON} \times C_L}{N}\right] = \ln 2 \times \left[R_{ON} \times C_{SL} + \frac{R_{ON} \times C_L}{N}\right]. \quad (2\text{-}26)$$

The mean of propagation delay is equal to:

$$\mu_{t_{pd}} = \ln 2 \times \left[R_{ON} \times C_{SL} + \frac{R_{ON} \times C_L}{N}\right]. \quad (2\text{-}27)$$

Table 2.3: Analytical model verification versus SPICE simulation for INV1X assuming R_{on}=36.2kΩ, C_{SL}=0.22fF, C_{LOAD}=0

NCNT	SPICE Simulation			Our Analytical Model		
	μ(ps)	***σ(s)***	***σ/μ***	***μ(ps)***	***σ(s)***	***σ/μ***
40	0.55	3.1e-27	0	0.78	0	0
50	0.55	3.1e-27	0	0.78	0	0
80	0.55	3.1e-27	0	0.78	0	0
100	0.55	3.1e-27	0	0.78	0	0
150	0.55	3.1e-27	0	0.78	0	0
200	0.55	3.1e-27	0	0.78	0	0

By assuming that the standard deviation compared to the mean value is very small, we use (2-19) and (2-22) to calculate the standard deviation of t_{pd}. Hence, the coefficient variation of gate delay due to CNT density variation can be found by:

$$(\sigma/\mu)_{t_{pd}} \approx \left(\frac{R_{ON} \times C_L\left(\frac{\sigma_{\bar{N}}}{(\bar{N})^2}\right)}{R_{ON} \times C_{SL} + R_{ON} \times C_L\left(\frac{1}{\bar{N}}\right)}\right) \quad (2\text{-}28)$$

Assuming that the self-loading capacitance is negligible in comparison to the load capacitance ($C_L \gg C_{SL}$), (2-28) can be simplified to (2-24), shown in Section 2.3.1.1

The average and standard deviation of the gate delay using analytical model versus simulated data extracted from SPICE simulation on a 1X inverter using Monte Carlo simulation has shown in Table 2.4. As shown the predicted delay variation matches very well with the SPICE simulation results.

An RC switch driving a long wire with lumped capacitance and resistance, shown in Figure 2.12-c, has also been considered for the third stage to drive the general formula. The delay propagation as a function of the number of CNTs is:

$$t_{pd} = \ln 2 \times \left[(R_{ON} \times C_{SL} + R_W \times C_W) + \frac{2R_{ON} \times C_W}{N} \right], \tag{2-29}$$

where R_W and C_W are wire resistance and wire capacitance respectively.

The mean of propagation delay is equal to:

$$\mu_{t_{pd}} = \ln 2 \times \left[(R_{ON} \times C_{SL} + R_W \times C_W) + \frac{2R_{ON} \times C_W}{N} \right]. \tag{2-30}$$

Because the only random variation in (2-30) is N, from (2-19) and (2-23), the propagation delay variation can be found by:

$$(\sigma/\mu)_{t_{pd}} \approx \left(\frac{2R_{ON}C_L\left(\frac{\sigma_{\overline{N}}}{(\overline{N})^2}\right)}{R_{ON}C_{SL}+R_W C_W+2R_{ON}C_W\left(\frac{1}{\overline{N}}\right)} \right). \tag{2-31}$$

As we infer from (2-31), resistance and capacitance of wire affect the coefficient of variation. Using the (2-31), a three-dimensional plot of the percentage of variation for inverter delay as a function of the expected number of CNTs and the wirelength is shown in Figure 2.13.

In the fourth and final stage in driving the general formula, we consider one gate driving another gate along the long interconnect, shown in Figure 2.12-d. Equation (2-32) gives the propagation delay as a function of the number of CNTs in the driver and load gates:

$$t_{pd} = \ln 2 \times \left[(R_{ON}C_{SL} + R_W C_W) + \frac{2R_{ON}C_W}{N} + MR_w C_{IN} + \frac{M}{N} R_{ON} C_{IN} \right], \qquad (2\text{-}32)$$

where C_{IN} is the input gate capacitance of the loading gate.

Table 2.4: Analytical model verification versus SPICE simulation for INV1X, assuming R_{on}=36.2kΩ, C_{SL}=0.22fF, C_{LOAD}=100fF

NCNT	Analytical Model			SPICE Simulation		
	μ(pS)	σ(pS)	σ/μ(%)	μ(pS)	σ(pS)	σ/μ(%)
40	62.96	6.99	11.10	65.41	7.66	11.71
50	50.48	5.00	9.90	52.21	5.12	9.81
80	31.75	2.47	7.78	33.15	2.85	8.60
100	25.51	1.77	6.94	26.17	1.97	7.53
150	17.19	0.96	5.58	17.57	0.97	5.52
200	13.03	0.63	4.83	13.30	0.65	4.88

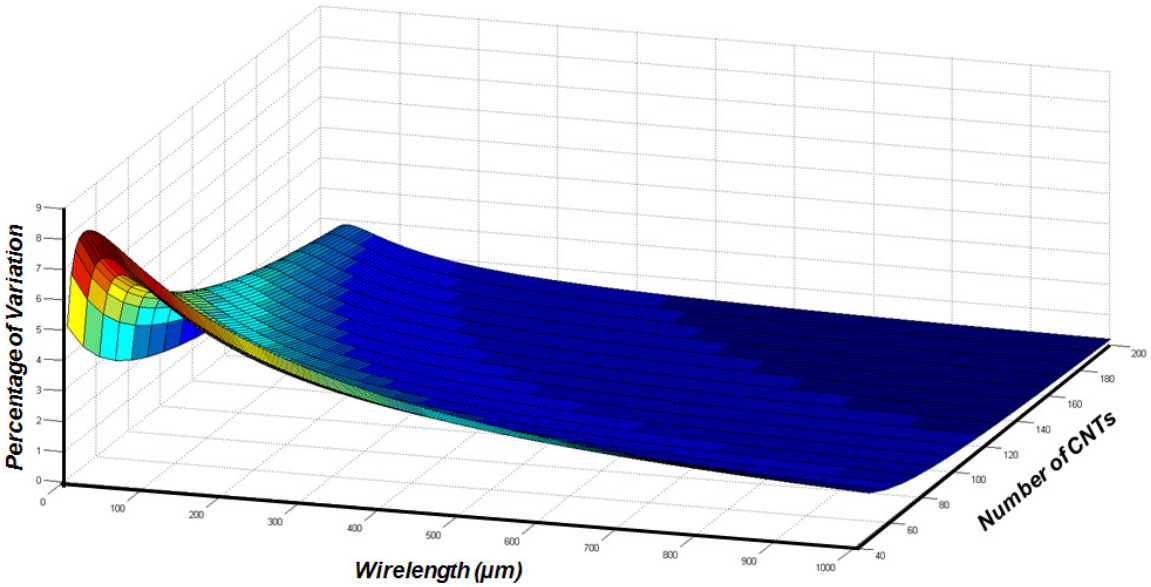


Figure 2.13. Percentage of variation for inverter delay as a function of the expected number of CNTs and the wirelength.

To drive an analytical relation between the number of CNTs per CNFET in driver and load gates and gate delay distribution, let's assume that X and Y are random variables with mean values of μ_X and μ_Y as the standard deviations of σ_X and σ_Y, respectively. Generally, the

distribution of random variable $Z=X/Y$ is not necessarily normal [60]. However, if the variation is small ($\sigma_Y/\mu_Y<<1$), then $Z=X/Y$ can be well approximated by a normal distribution [61]. It has been shown in [61] that for $Z=X/Y$, we have:

$$\mu_Z = \frac{\mu_X}{\mu_Y} . \tag{2-33}$$

To find the standard deviation for Z, when the standard deviation is small, we take a derivative of Z as follows:

$$W = \frac{X}{Y} \quad \Rightarrow dW = \frac{dX}{Y} - \frac{XdY}{Y^2} . \tag{2-34}$$

Considering that the derivative can be represented by small changes, (2-34) can be rewritten as:

$$\Delta W = \frac{\Delta X}{Y} - \frac{X\Delta Y}{Y^2} . \tag{2-35}$$

Assuming that the standard deviation compared to the mean value is very small ($\sigma/\mu<<1$) Δ can be substituted with σ in (2-35), which yields

$$\sigma_w{}^2 = \left(\frac{\sigma_x}{\mu_y}\right)^2 + \left(\frac{\mu_x\sigma_y}{\mu_y}\right)^2 . \tag{2-36}$$

To illustrate the accuracy of (2-33) and (2-36), pairs of random variables, which can represent the numbers of CNTs under the CNFET driver and load gates, are selected. Table 2.5 shows the mean and standard deviation of these pairs using an analytical model versus MATLAB simulation.

By assuming that the standard deviation compared to the mean value is very small, we use (2-36) to calculate the standard deviation of propagation delay. Hence, the variation coefficient due to the CNT density variation can be found by:

$$(\sigma/\mu)_{t_{pd}} \approx \left(\frac{2R_{ON}C_L\left(\frac{\sigma_{\overline{N}}}{(\overline{N})^2}\right)+\sigma_{\overline{M}}R_wC_{IN}+\frac{\overline{M}}{2\overline{N}^3}(\overline{M}+\overline{N})+R_{ON}C_{IN}\sqrt{\frac{\overline{M}}{2\overline{N}^3}(\overline{M}+\overline{N})}}{(R_{ON}C_{SL}+R_WC_W)+\frac{2R_{ON}C_W}{\overline{N}}+\overline{M}R_wC_{IN}+\frac{\overline{M}}{\overline{N}}R_{ON}C_{IN}} \right) . \tag{2-37}$$

Using Equation (2-37), a three-dimensional plot of the percentage of variation of inverter delay as a function of the expected number of CNTs in the driver gate and the wirelength for different fan-outs are shown in Figure 2.14.

Table 2.5: Mean and standard deviation, analytical model versus MATLAB simulation

(X/Y)	Analytical Model		MATLAB Simulation	
	Mean(μ)	Sigma(σ)	Mean(μ)	Sigma(σ)
(40/40)	1	0.158	1.01	0.164
(40/80)	0.50	0.0685	0.50	0.070
(40/100)	0.40	0.0529	0.40	0.054
(40/120)	0.33	0.0430	0.33	0.044
(40/160)	0.25	0.0313	0.25	0.031

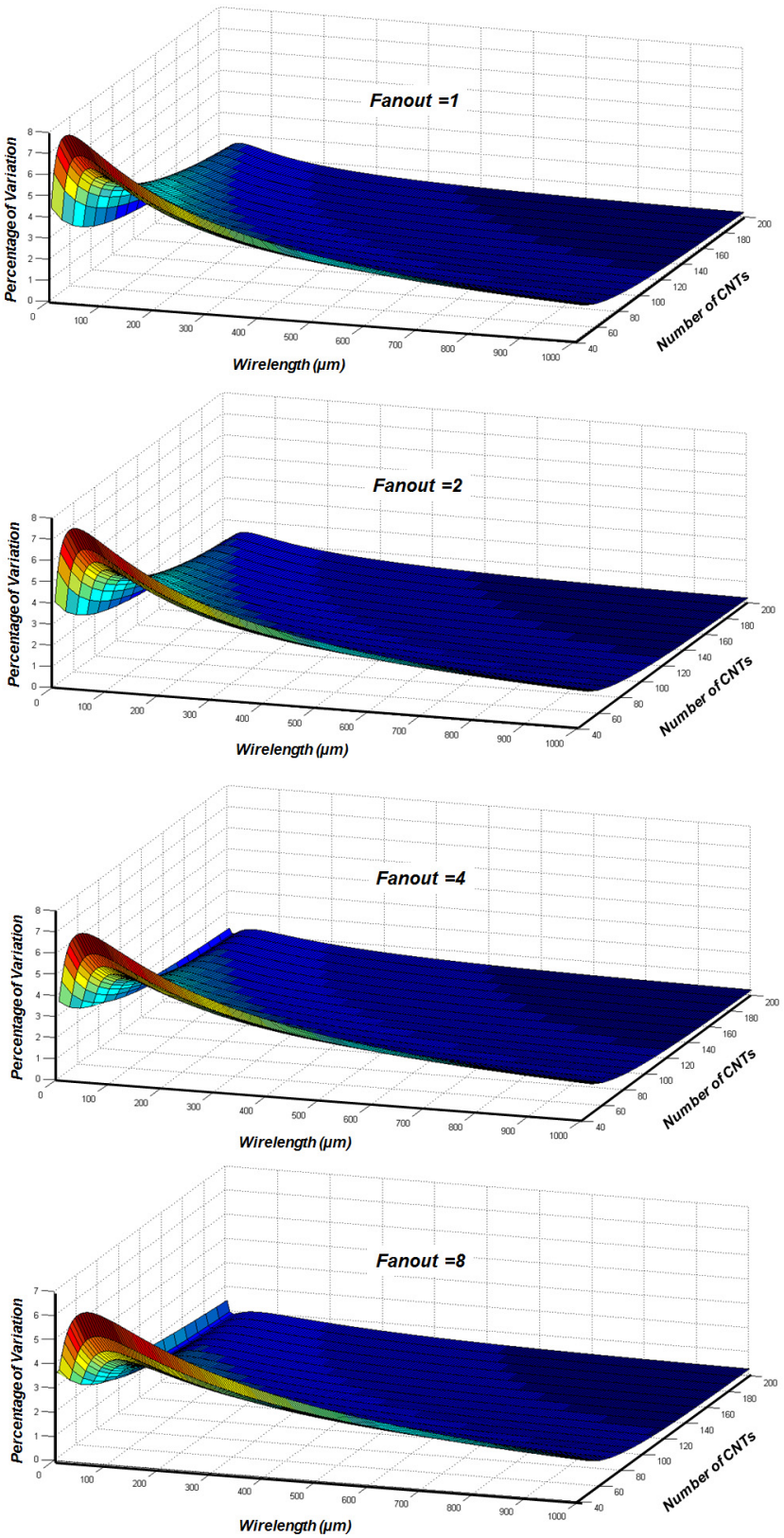


Figure 2.14. Percentage of delay variation as a function of the expected number of CNTs and the wirelength. a) Fan-out=1; b) Fan-out=2; c) Fan-out=4; d) Fan-out=8.

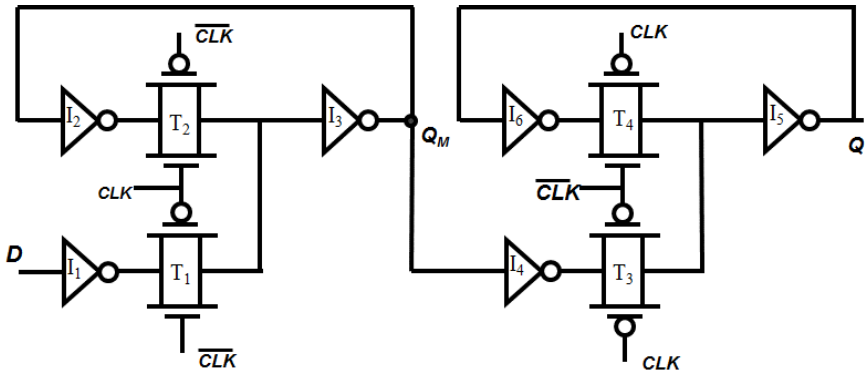


Figure 2.15. Master slave positive edge-triggered Flip-Flop [59]

2.4 Impact of CNT density variation on sequential circuits

Memory elements, including Flip-Flops and latches, are common in digital circuits. Therefore, it is important to quantify the effects of CNT density variation on CNFET-based memory elements. There are many approaches for constructing Flip-Flops. One very common technique involves the use of transmission gates and inverters. A complete transistor-level implementation of a master-slave positive edge-triggered Flip-Flop is shown in Figure 2.15 [59].

Flip-Flops are characterized by three important timing parameters: *setup time*, *hold time,* and the *propagation delay* (t_{c2q}). Therefore, we quantify these parameters under the effect of CNT density variation.

2.4.1 Setup time

Setup time has been defined as the minimum amount of time the data should be held steady before the clock event. We assume that the propagation delay of each inverter and transmission gate in Figure 2.15 is t_{pd_INV} and $t_{pd\text{-}TG,}$ respectively. For the transmission gate multiplexer-based Flip-Flop, input D must propagate through I1, T1, I3, and I2 before the rising edge of the clock. Hence, the setup time is [59]:

$$t_{SETUP} = 3 \times t_{pd-INV} + t_{pd-TG}. \tag{2-38}$$

Consider the equivalent circuit of the Flip-Flop cell to calculate the setup time shown in Figure 2.16-a. The equivalent circuit to calculate delay from the input of I2 to the input of T2 has been modeled as an RC switch and is shown in Figure 2.16-b, which is the same as Figure 2.12-a. Therefore, based on (2-24), the value of the setup time is constant and independent of the number of CNTs under the gate channel. The standard deviation of a constant value is zero; hence, the percentage of setup time variation due to CNT density variation is 0%. The SPICE simulation confirms that the percentage of setup time variation due to CNT density variation is negligible.

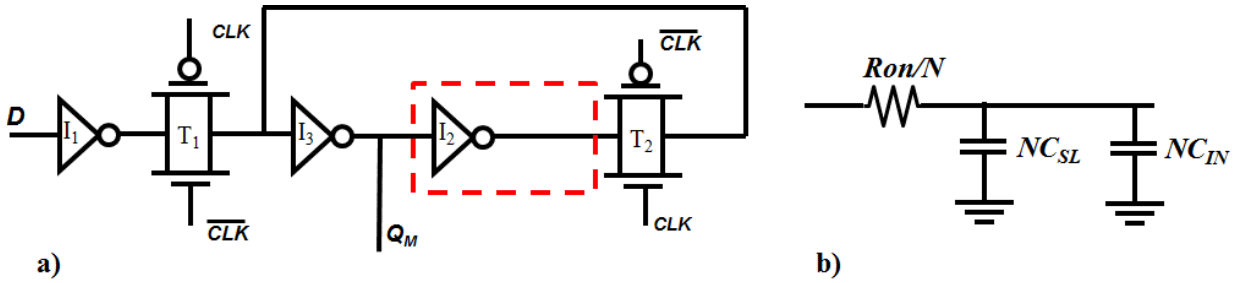


Figure 2.16. a) Equivalent circuit to calculate the setup time; b) RC switch model for the last stage to measure setup time delay.

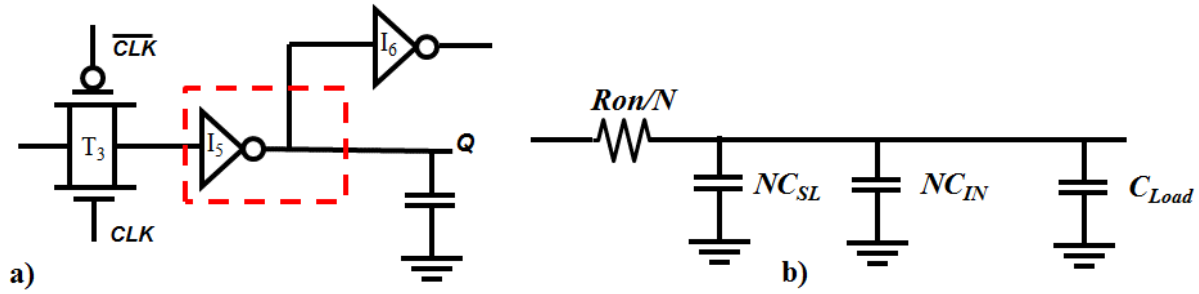


Figure 2.17. a) Equivalent circuit to calculate the propagation delay b) RC switch model for the last stage to measure the propagation delay

2.4.2 Propagation delay

Figure 2.17-a shows the equivalent circuit to calculate the propagation delay. The input of I5 to the output of the Flip-Flop has been modeled as an RC switch and is shown in Figure 2.17-b, which is the same as Figure 2.12-b. Because Flip-Flops usually drive many nets and consequently have a large fan-out, the amount of load capacitance is considerable in

comparison with the self-loading and input capacitance of I6. Therefore, based on the equivalent circuit shown in Figure 2.12-b, the coefficient variation of the propagation delay due to CNT density variation can be found by using (2-24). Table 2.6 shows the percentage of propagation delay variation using Equation (2-24) versus simulated data extracted from SPICE using the Monte Carlo simulation.

Table 2.6: Analytical model verification for percentage of propagation delay variation versus SPICE simulation, assuming C_{LOAD}=200fF.

NCNT	Analytical Model - σ/μ (%)	SPICE Simulation - σ/μ (%)
40	11.18	11.71
50	10	9.81
80	7.91	8.60
100	7.07	7.53
150	5.77	5.52
200	5	4.88

2.4.3 Hold time

In this Flip-Flop configuration, the hold time is 0 [59]. However, for other Flip-Flop configurations, the hold time can be quantified by the inverter gate delay [58, 59]. Therefore, by modeling to the RC switch, the effect of CNT density variation on hold-time can be quantified.

2.5 Impact of CNT density variation on static random access memory

In this section, we study the effects of CNT density variation on performance and stability of 6-T static random access memory (SRAM) cells. Because of robustness, low power operation, and short access time, 6T SRAM is commonly used [64]. The 6T SRAM cell, shown in Figure 2.18, is comprised of a pair of weak cross-coupled inverters holding the

state and a pair of access transistors to read and write the state [58, 59]. To ensure read stability as well as writability, the transistors must satisfy ratio constraints. The N-type CNFET pull-down transistor in the cross-coupled inverters must be the strongest. The access transistors are of intermediate strength, and the P-type CNFET pull-up transistors must be weak [58, 59]. To satisfy these constraints, N1 and N3 have 3×N CNTs under the gate. Also, P1 and P2 have N CNTs. The access transistors have 2×N CNTs under the gate channel [65]. To achieve good layout density, all of the transistors use the same bundle of CNTs.

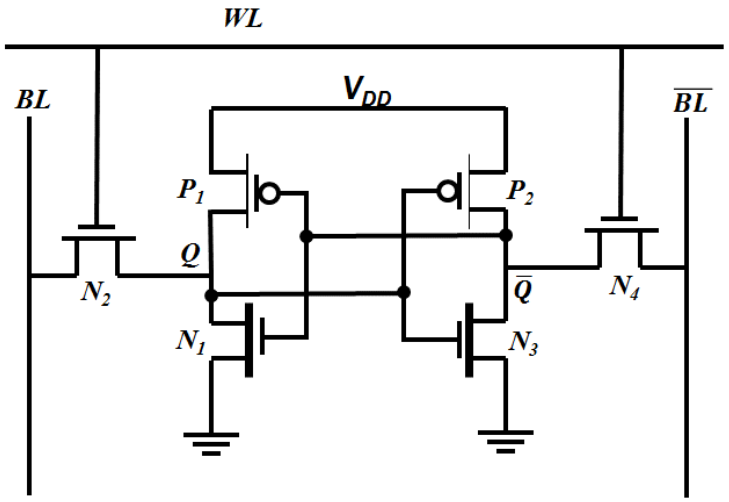


Figure 2.18. The 6-T static random access memory cell.

2.5.1 Write delay

Write delay has been defined as the time from the 50% activation of wordline (WL) to the time internal nodes Q and $\bar{Q}$ reach 50% of their final values [66]. To analyze the effect of CNT density variation on write delay, without loss of generality, assume Q is initially 0 and that we wish to write a 1 into the cell. Bit line (BL) is precharged high and is left floating. Bit-line bar ($\overline{BL}$) is pulled low by a write driver. The capacitance at node Q consists of

$$C_Q = 2 \times NC_{SL-N2} + 3 \times NC_{SL-N1} + NC_{SL-P1} + 3 \times NC_{IN-N3} + NC_{IN-P2} = K \times N \quad (2\text{-}39)$$

where C_{SL} and C_{IN} are self-loading and input capacitance, respectively. As we see, K is constant and N is the only random variable in C_Q. Because the node Q charges through access transistor N2, the RC switch model for writing operation is similar to Figure 2.12-a, and the write delay is independent of the number of CNTs. As we expect, the percentage of write-delay variation due to CNTS density variation is 0%. Table 2.7 shows the extracted percentage of write-delay variation from SPICE. As we see, the percentage of write-delay variation is negligible.

Table 2.7: Extracted percentage of write-delay variation due to CNT density variation form SPICE simulation.

NCNT	SPICE Simulation- σ/μ (%)
50	0.00930
80	0.00447
100	0.00392
150	0.00166
200	0.00112
250	0.00079

2.5.2 Read delay

Read delay is defined as the time required to develop a 100mV differential voltage between BL and $\overline{BL}$ after WL reaches 50% of its final swing [67]. Again, to analyze the effect of CNT density variation on read delay, without loss of generality, assume Q is initially 0 and thus $\bar{Q}$ is initially 1. The BL and $\overline{BL}$ are both initially precharged high and are left floating. $\bar{Q}$ and $\overline{BL}$ both should remain 1[58, 59]. The equivalent half-circuit of the SRAM cell is shown in Figure 2.19-a, where C_{BL} is the bit-line capacitor, and R_{N4} is the equivalent resistance of N4 [68].

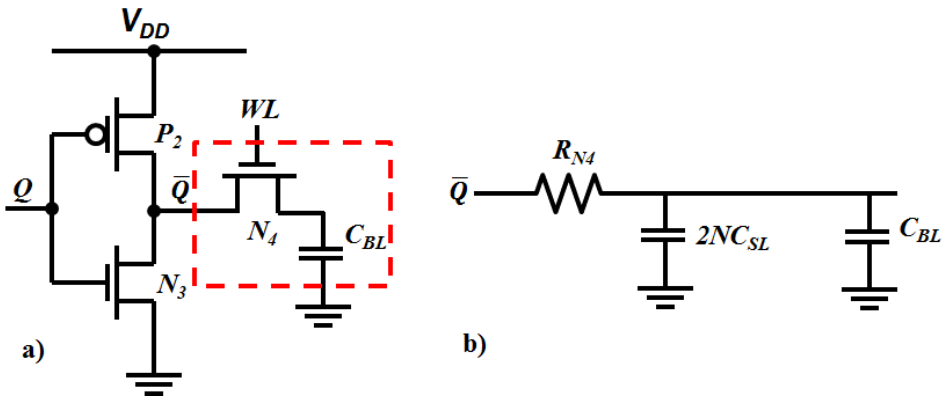


Figure 2.19. a) Equivalent half-circuit of SRAM for read delay [68]; b) Equivalent RC switch for read delay.

Because access transistor N4 is responsible for discharging C_{BL}, the read delay variation equation due to CNT density variation can be simplified as (2-25) based on Figure 2.19-b. The self-loading capacitance is negligible in comparison to the large C_{BL}; therefore, the percentage of read delay variation is equal to (2-23). The percentage of read-delay variation using (2-23) versus simulated data extracted from SPICE is shown in Table 2.8.

Table 2.8: Analytical model verification for percentage of read-delay variation versus SPICE simulation, assuming C_{BL}=200fF.

NCNT	Analytical Model- σ/μ (%)	SPICE Simulation- σ/μ (%)
50	10	10.1
80	7.91	7.54
100	7.07	6.63
150	5.77	5.31
200	5	4.81
250	4.47	4.15

2.5.3 Static noise margin

The side of the maximum nested square between the normal and mirrored voltage transfer curves (VTCs) for a pair of cross-coupled inverters has been defined as static noise margin

(SNM) [39, 69,70]. Because the stability and writability of the SRAM cells are quantified by the hold margin, the read margin, and the write margin, we analyze the effects of CNT density variation on these criteria. The static noise margin while the cell is holding its state and being neither read nor written is defined as *hold margin*. As shown in Figure 2.20-a, noise source Vn is applied to each of the cross-coupled inverters while the access transistors are OFF and do not affect the circuit behavior [58].

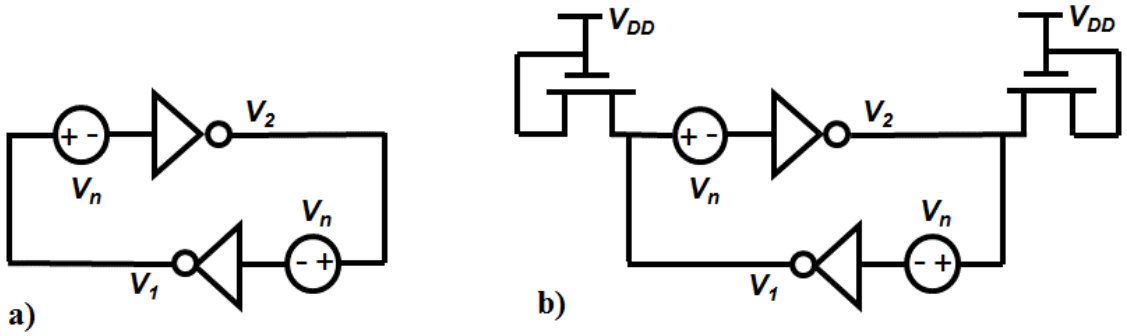


Figure 2.20. Cross-coupled inverters with noise sources for: a) hold margin; and b) read margin [58].

The BL and $\overline{BL}$ are initially precharged, and the access transistor tends to pull the low node up when the cell is being read. The tendency of pulling up the low node distorts the voltage transfer characteristics. The static noise margin under these circumstances that is shown in Figure 2.20-b has been defined as the *read margin* [58].

The read/hold margins depend on threshold voltage, power supply, and Cell Ratio. As we saw in section 2.2, threshold voltage is independent of CNT density variation. The cell ratio is 3/2, which is defined as the ratio of the number of CNTs in the drive transistors to that of access transistors. The power supply is fixed. Therefore, CNT density variation does not have any effect on read/hold margins and the percentage of variation due to the CNT density variation is 0%.

During writing the new state, access transistors N2 and N4 must overpower the pull-up P1 and P2 transistors to create a single, stable state. The *write margin* is determined by a

similar fashion as a read margin, with one access transistor pulling to 0 and the other to 1. If the |Vn| is too large, a second stable state will exist, preventing the function of writes [58]. The write margin is inversely proportional to the ratio of the number of CNTs in the pull-up transistors to the access transistors-1/2 in this case-and to the threshold voltage [65]. Therefore, the CNT density variation does not have any effect on write margins and percentage of variation due to CNT density variation is 0%. Table 2.9 shows the extracted percentage of hold margin, read margin, and write margin variations from SPICE. As we expect, these percentages of variation are negligible.

Table 2.9: Extracted percentage of hold-margin, read-margin, and write-margin variations, due to CNT density variation form SPICE simulation.

NCNT	SPICE Simulation $\sigma/\mu(\%)$		
	Hold Margin	Read Margin	Write Margin
50	0.090479	0.011128	0.0015285
80	0.000263	0.000987	0.0627869
100	0.128025	0.007852	0.020050
150	0.18649	0.000398	0.038366
200	0.40003	0.040172	0.0234980
250	0.15409	0.008596	0.0000084

2.6 CNT density variation impacts on system level

After device and gate levels, the last level to investigate the impact of CNT density variation on a Gigascale integrated system is the CNFET system level. The path delay is referred to the total delay for the logic to propagate through a logic path. This corresponds to the sum of the delays through the various logic cells and nets along the path [58, 59, 71]. Based on the placement of CNFET gates, if the gate delays along a path are independent,

they tend to average out in the overall path delay [38, 72]. Figure 2.21-a shows an example of a data path between two Flip-Flops. Without loss of generality, we assume the path between Flip-Flops goes through *n* NAND2 gates. Figure 2.21-b shows the placement of NAND2 gates where they use an independent bundle of CNTs along the Y-axis. Each gate has an independent normal-delay distribution P_1, P_2,...,P_n, with the same mean μ and standard deviations σ. The mean of the path delay relative to the mean of individual gate delays is given by[38]:

$$\mu_{PATH} = \sum_{i=1}^{n} \mu_{Gi} = n\mu_{Gi} \,, \tag{2-40}$$

where *n* is the number of stages and μ_{Gi} is the mean of gate delay. Because the gates have been placed on independent bundles of CNTS, the standard deviation of path delay is:

$$\sigma_{PATH} = \sqrt{\sum_{i=1}^{n} {\sigma_{Gi}}^2} = \sigma_{Gi}\sqrt{n} \,. \tag{2-41}$$

The coefficient of variation, σ/μ, of the path delay relative to the gate delay by using (2-40) and (2-41) is given by[38]:

$$(\sigma/\mu)_{PATH} = \frac{1}{\sqrt{n}}(\sigma/\mu)_{Gate} \,. \tag{2-42}$$

The placement of NAND2 gates, where they use the same bundle of CNTs along the Y-axis, is shown in Figure 2.21-c. Because the gates used the same bundle of CNTs, the delay distributions are correlated; therefore, the standard deviation of path delay is equal to sum of them [38]:

$$\sigma_{PATH} = \sum_{i=1}^{n} \sigma_{Gi} = n\,\sigma_{Gi} \,. \tag{2-43}$$

The mean of the path delay that is independent of the placement follows Equation (2-43). Therefore, the coefficient deviation for the path delay becomes:

$$(\sigma/\mu)_{PATH} = (\sigma/\mu)_{Gate} \tag{2-44}$$

In both cases, by increasing the number of gates in the path, the standard deviation of the path delay increases. However, with the assumption of independence, the ratio σ/μ scales as $\frac{1}{\sqrt{n}}$, whereas it remains constant under the fully correlated assumption [38].

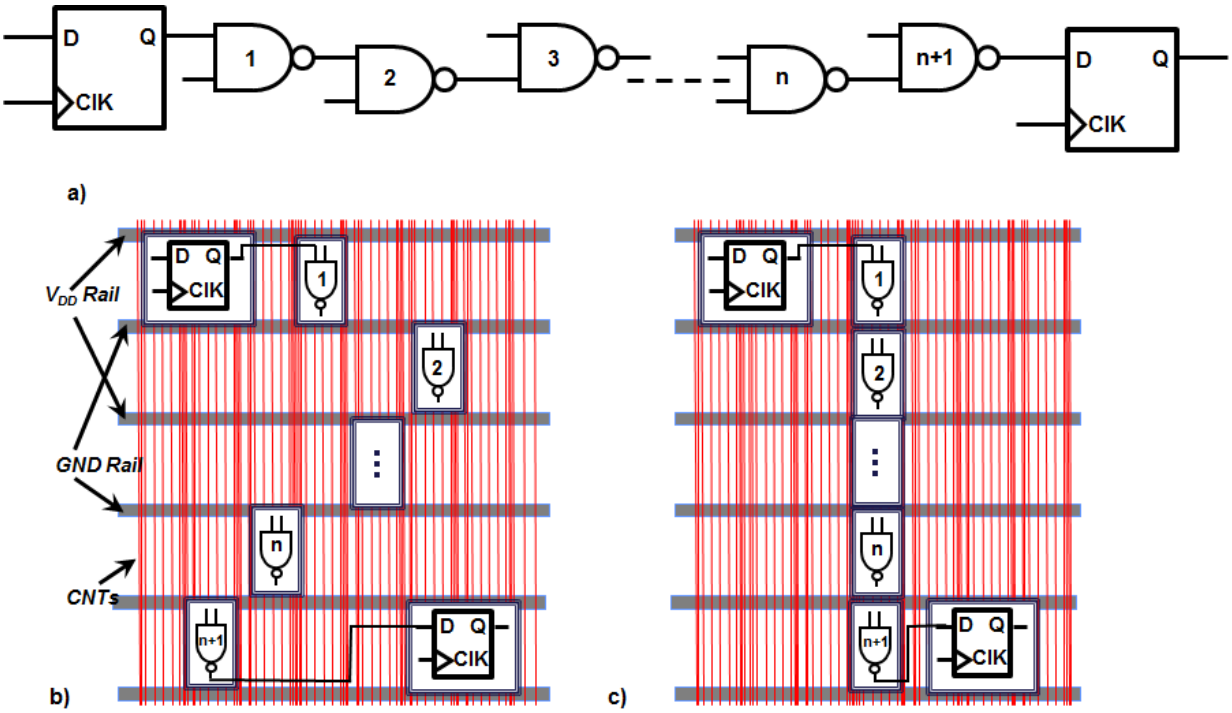


Figure 2.21. a) Timing path; b) Timing path placed through uncorrelated CNTs; c) Timing path placed through correlated CNTs.

Chapter 3

3 CNFET VLSI systems in the presence of Metallic CNTs

The presence of m-CNTs affects the delay and power dissipation of CNFET-based circuits. Since there is no CNT fabrication technique that guarantees ideal CNT growth with exclusively semiconducting CNTs, the impact of m-CNTs on CNFET circuits must be quantified and carefully considered. In this chapter, we present analytical models for gate delay and power consumption in CNFET logic gates in the presence of m-CNTs. These models can be great tools in predicting the limitations and opportunities for a robust and reliable CNFET system in the presence of m-CNTs.

3.1 Introduction

The presence of m-CNTs can severely change the individual gate power dissipation and delay. The large percentage of m-CNTs is not tolerable from the noise margin and the power consumption perspectives [37, 39]. Accurate characterization of gate delay, path delay, static power, and dynamic power consumptions in CNFET technology to determine an acceptable range of m-CNTs is necessary for future CNFET technology design and development. Figure 3.1 illustrates the effects of m-CNTs on delay and functionality in a CNFET logic block using SPICE simulations. For example, in case (1), where all the gates composed of only semiconductor CNTs (s-CNTs), the output node has full swing from V_{DD} to 0. However, the waveforms in cases (2) and (3) show that the presence of m-CNTs degrades the output voltage swing and consequently increases the path delay. The high

percentage of m-CNTs in G1 (P_{M_G1}=15%) ultimately results in a failure which is shown in case (4) in Figure 3.1.

Zhang *et al.* in [39] presented a probabilistic model, which incorporates processing and design parameters and enables quantitative analysis of the impact of m-CNTs on leakage, noise margin, and delay variations of CNFET-based digital logic circuits. Ashraf *et al.* in [27, 48] presented some analytical models to estimate the functional yield of logic gates using different configurations of CNFETs. Although these previous works presented some good models that describe the effect of m-CNTs on circuit parameters, there is still no simple analytical model that can predict the system-level requirement for acceptable range of m-CNTs.

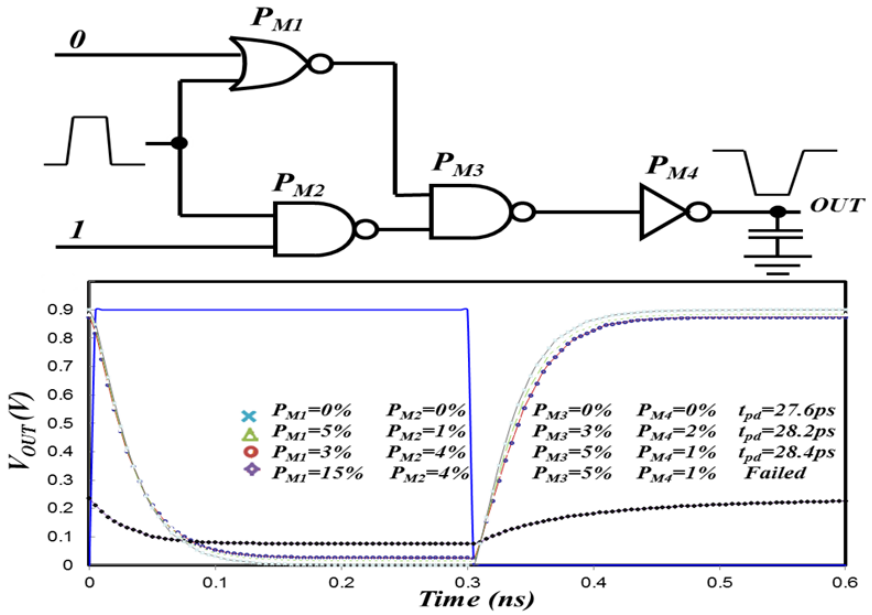


Figure 3.1. Simulated effects of m-CNTs on circuit delay contributed by various percentage of m-CNTs on gates.

3.2 Metallic CNTs in CNFET

Depending on the chirality of the carbon nanotube, the growth of CNT can result in either metallic or semiconductor CNTs [4]. Since the conductivity of m-CNTs cannot be controlled by the gate, they cannot be used to make CNFETs. Therefore, a CNFET made

entirely from m-CNTs will have a current that is only dependent on the drain–source voltage, not on the gate voltage. Instead, CNFETs made up entirely from s-CNTs will exhibit a good gate-control, often with ION–IOFF ratios of up to 10^6 [3, 17].

Due to random distributions of CNTs, about 33% of the CNTs are normally grown as metallic CNT [4]. However, with the advancement of CNT fabrications, the current CNT synthesis techniques yield about 1% or lower m-CNTs [29-32]. There has been work on m-CNT removal after CNT growth by electrical burning [33] or selective etching [34]. However, because of the non-ideality of all of these techniques, a fraction of m-CNTs still survive after m-CNT removal process. These remaining m-CNTs can cause significant problems in CNFET circuits [27, 35-37].

As mentioned in chapter 1, a CNFET containing a single CNT in its channel is highly sensitive to CNT diameter variations. Therefore, to provide sufficient drive currents, practical circuits require CNFETs with multiple CNTs in their channel [45]. These CNFETs will generally consist of a mix of m-CNTs and s-CNTs. Lin *et al.* have presented some experimental data of a CNFET fabricated on a mix of s-CNTs and m-CNTs [35-37]. As expected, they showed that the device with m-CNTs has an extremely low ION–IOFF ratio. Using such CNFETs to construct gates leads to a poor noise margin, which will result in signal loss over just a few cascaded stages as shown in Figure 3.1 case (4). Therefore, it will be impossible to create functional large-scale circuits using CNTs without addressing the m-CNTs problem.

3.2.1 Impact of m-CNTs on I_{ON}-I_{OFF} ratio

Since a higher I_{ON} is required to achieve better performance and, a lower I_{OFF} is desirable to get lower leakage power, a CNFET which is made only form s-CNTs with the overall

device I_{ON}–I_{OFF} ratio of ~ 10^6 is desirable. By assuming that the I_{ON} of s-CNT is the same as the I_{ON} of m-CNT [27, 35-37, 39, 48], the I_{ON}–I_{OFF} ratio of the CNFET degrades to approximately the inverse of probability of m-CNTs. For example, only 1% m-CNTs in CNFET can reduce the overall device I_{ON}–I_{OFF} ratio of ~ 10^6 to about 100. This is not acceptable in many VLSI applications.

Although the low I_{ON}–I_{OFF} ratio for CNFETs is undesirable (due to its high leakage current), a logic gate made of such leaky CNFTE can still be functional if the probability of m-CNTs is very small as simulated data shown in Figure 3.1 case (2) and (3).

3.2.2 Impact of m-CNTs on the output voltage swing

Figure 3.2 shows the schematic of an inverter that consists of one P-type CNFET and one N-type CNFET with mix of aligned correlated m-CNTs and s-CNTs under the gate window. To analyze the impact of m-CNTs on the output voltage swing, the voltage transfer characteristics (VTC) of the CNFET inverter is simulated and shown is Figure 3.3. As shown in Figure 3.3, the presence of m-CNTs can drastically affect the VTC and the output voltage swing of the CNFET inverter. Logic swing (V_{LS}), which is defined as the magnitude of voltage differences between the static high output voltage (V_{OH}) and static low output voltage (V_{OL}) levels, is given by [73]:

$$V_{LS} = V_{OH} - V_{OL} \ . \tag{3-1}$$

In the presence of m-CNTs the logic swing decreases because m-CNTs conduct current even when the corresponding CNFET is turned off. This undesirable conductivity lowers the V_{OH} and increases the V_{OL} of the gate, which results in reducing the output voltage swing as shown in Figure 3.3.

To determine V_{OH} in a CNFET inverter of Figure 3.5, where there are N_M m-CNTs under the gate, we model the m-CNTs as separate always-ON CNFETs as shown in Figure 3.4. While there is no simplified formula for CNFET similar to α-power law in MOSFETs[74], both experimental and simulation results show that if contacts are ideal (ohmic contacts) and the channel is long, then the I-V characteristics of CNFETs are similar to MOSFETs [3, 75-77].

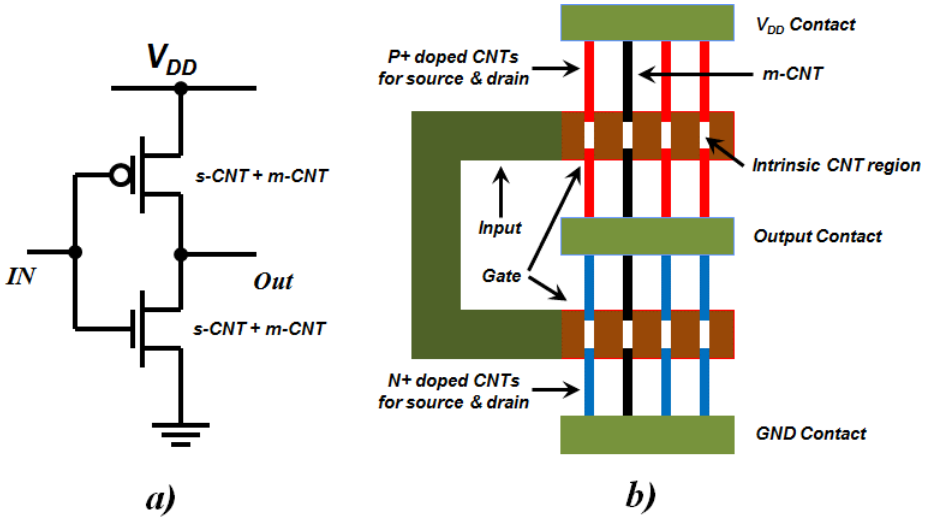


Figure 3.2. Schematics of a logic inverter made up of one P-type CNFET and one N-type CNFET laid out on aligned correlated mix of m-CNTs and s-CNTs [39].

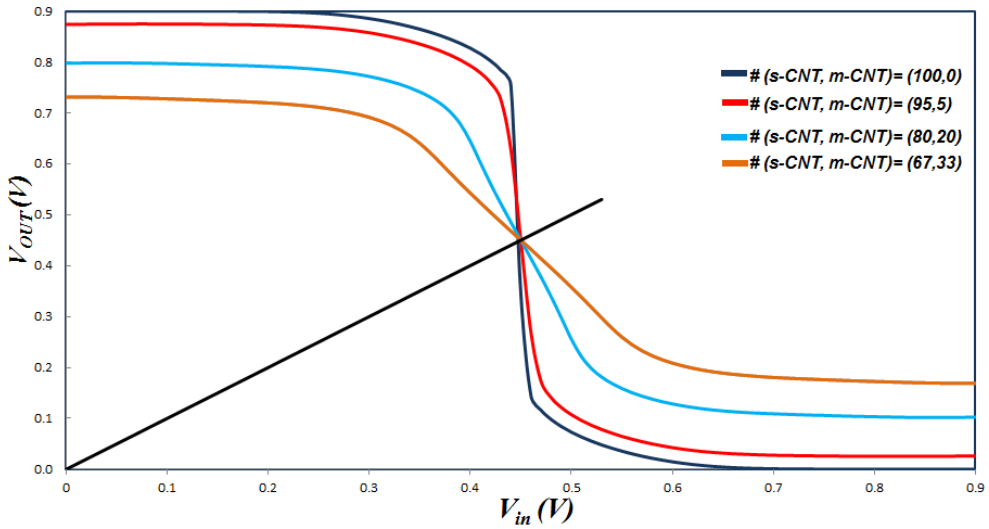


Figure 3.3. Simulated Voltage Transfer Characteristics of CNFET inverter in the presence of m-CNTs.

To drive an analytical formula for V_{OH} based on the number of m-CNTs in pull-down network, we assume that the pull-up CNFET is in linear region and the CNFET with m-CNTs in the pull down transistor is in saturation region as illustrated in Figure 3.5-a. Furthermore,

$$I_{DS,P} = I_{DS,N} \ . \tag{3-2}$$

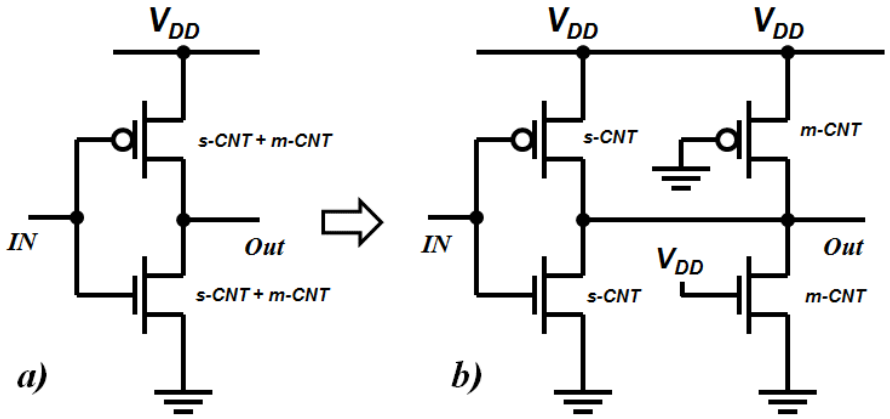


Figure 3.4. a) Schematic of imperfect CNFET inverter, b) Modeling of m-CNTs in CNFET as always ON transistors.

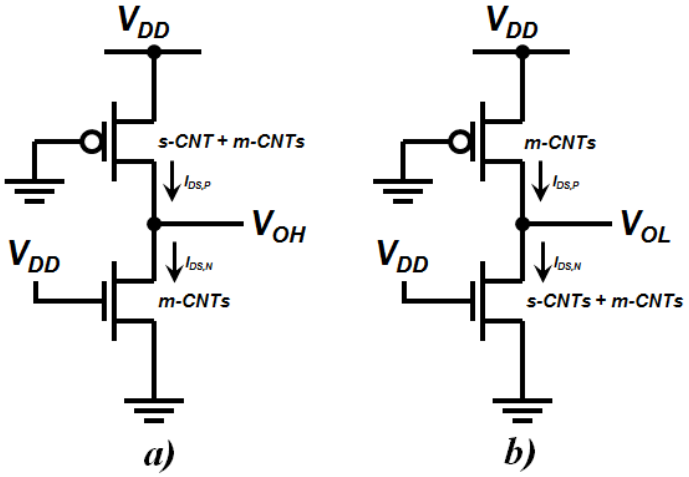


Figure 3.5. Imperfect CNFET inverter models for a) V_{OH}, b) V_{OL}.

The currents in pull-up and pull-down networks are given by (3-3) and (3-4), respectively

$$I_{DS,P} = K\ N_T \left[(V_{SG} - V_t)(V_{DD} - V_{OH}) - \frac{(V_{DD}-V_{OH})^2}{2}\right], \tag{3-3}$$

$$I_{DS,N} = I_{ON,N} = \frac{K}{2}\ N_M (V_{GS} - V_t)^2, \tag{3-4}$$

where K is the CNFET transconductance parameter, N_T is the total number of CNT including s-CNTs and m-CNTS, N_M is the number of m-CNT in pull-down transistor, V_{GS} is the gate to source voltage of transistors, and Vt is threshold voltage. The threshold voltage is measured from the device I-V characteristics using the linear extrapolation technique [55]. To simplify our analytical model we assume that the absolute value of threshold voltage for both N-type and P-type CNFET are equal [3]. Considering the fact that the absolute value of V_{GS} in (3-3) and (3-4) is V_{DD}, and substituting (3-3) and (3-4) into (3-2) yields:

$$K\ N_T\left[(V_{DD}-V_t)(V_{DD}-V_{OH})-\frac{(V_{DD}-V_{OH})^2}{2}\right]=$$

$$\frac{K}{2}\ N_M(V_{DD}-V_t)^2\ . \tag{3-5}$$

From (3-5), V_{OH} can be found analytically as:

$$V_{OH}=V_{DD}-(V_{DD}-V_t)\left(1-\sqrt{1-\frac{N_M}{N_T}}\right). \tag{3-6}$$

Similarly, by analyzing Figure 3.5-b, the analytical model for V_{OL} can be found as:

$$V_{OL}=(V_{DD}-V_t)\left(1-\sqrt{1-\frac{N_M}{N_T}}\right). \tag{3-7}$$

Substituting (3-6) and (3-7) into (3-1) yields:

$$V_{LS}=V_{DD}-2(V_{DD}-V_t)\left(1-\sqrt{1-\frac{N_M}{N_T}}\right)\ . \tag{3-8}$$

Table 3.1 validates our analytical models for V_{OH} and V_{OL} based on (3-6) and (3-7) with simulated data extracted from SPICE for several gates including: INV1X, NAND2-1X, NOR2-1X, NAND3-1X, and NOR3-1X. As shown the analytical models data matches very well with the SPICE simulation results.

3.3 Impact of m-CNTs on gate delay

Propagation delay time, t_{pd}, is defined as the time delay from the input voltage crossing 50% to the output voltage crossing 50% [58]. The propagation delay of a logic gate is determined by the output current that can deliver to its load capacitance. It has been shown that the propagation delay of a logic gate, t_{pd}, can be estimated from [58]:

$$t_{pd} = \frac{C_{Load} \times \Delta V/2}{I_{Drive}}, \tag{3-9}$$

where C_{Load} is the load capacitance, I_{Drive} is the gate drive current, and ΔV is the maximum output voltage swing, which in the ideal case is V_{DD}. To calculate the gate delay for CNFET inverter in the presence of m-CNTs, the output gate voltage swing is determined by using V_{LS} from (3-8).

In a complementary CMOS-based circuit that has pull-up and pull-down networks, the gate drive current is given by [58, 59]

$$I_{Drive} = I_{ON} - I_{OFF} \ . \tag{3-10}$$

The same equation can be used for a complementary circuit build with N-type and P-type CNFETs [27, 39, 48]. For a CNFET with N_T CNTs under the gate, the $I_{ON,N}$ and $I_{OFF,N}$ can be expressed as [3, 45]:

$$I_{ON,N} \approx N_T \times I_{ON,1} \ , \tag{3-11}$$

$$I_{OFF,N} \approx N_T \times I_{OFF,1} \ , \tag{3-12}$$

where $I_{ON,1}$ and $I_{OFF,1}$ are the ON and OFF current of a CNFET with 1 CNT, respectively. Experimental result shows $I_{ON} >> I_{OFF}$ [3, 17, 27, 35-37, 39, 48]. Therefore, under an ideal scenario, when all the N_T CNTs under the CNFET gate are semiconducting and no m-CNTs exist, the drive current is:

$$I_{Drive} \approx N_T \times (I_{ON,1} - I_{OFF,1}) \approx N_T \times I_{ON,1} \ . \tag{3-13}$$

Table 3.1: Analytical model verification against SPICE simulation, assuming V_t =0.1862V, V_{DD}=0.9V N_T=100, N_M=P_M*N_T

P_M(%)	Analytical Model		SPICE Simulation									
			INV_1X		NAND2_1X		NAND3_1X		NOR2_1X		NOR3_1X	
	VOH	VOL	VOH	VOL	VOH	VOL	VOH	VOL	VOH	VOL	VOH	VOL
0	0.900	0	0.900	0	0.900	0	0.900	0	0.900	0	0.900	0
1	0.896	0.004	0.895	0.005	0.898	0.011	0.898	0.016	0.889	0.002	0.884	0.002
5	0.882	0.018	0.874	0.026	0.888	0.052	0.892	0.075	0.848	0.012	0.825	0.008
10	0.863	0.037	0.848	0.052	0.876	0.102	0.885	0.150	0.798	0.024	0.750	0.015
20	0.825	0.075	0.798	0.102	0.853	0.206	0.869	0.301	0.694	0.047	0.599	0.031

Substituting (3-13) into (3-9) gives the propagation delay of an ideal CNFET inverter to drive a fixed load capacitance:

$$t_{pd} = \frac{C_{Load} \times VDD/2}{N_T \times I_{ON,1}} . \quad (3\text{-}14)$$

However, in the presence of m-CNTs, due to the current coming from the OFF network, I_{OFF} is given by:

$$I_{OFF} = N_M \times I_{ON,1} . \quad (3\text{-}15)$$

The I_{Drive} of imperfect CNTFET is therefore simplified to:

$$I_{Drive} \approx N_T \times I_{ON,1} - N_M \times I_{ON,1},$$

$$I_{Drive} \approx (N_T - N_M) \times I_{ON,1} . \quad (3\text{-}16)$$

Figure 3.6 illustrates the SPICE simulation results, showing the effect of m-CNTs on I_{Drive} during discharging and charging of output load capacitance. As shown in Figure 3.6, increasing the number of m-CNTs decreases the driver current by the integer number of ON-current of a single CNT.

Substituting (3-6), (3-7), and (3-16) into (3-9) gives the propagation delay of an imperfect CNFET inverter that drives a fixed load capacitance as a function of the total number of CNTs and the number of m-CNTs:

$$t_{pd} = \frac{C_{Load} \times V_{LS}/2}{(N_T - N_M) \times I_{ON,1}} . \quad (3\text{-}17)$$

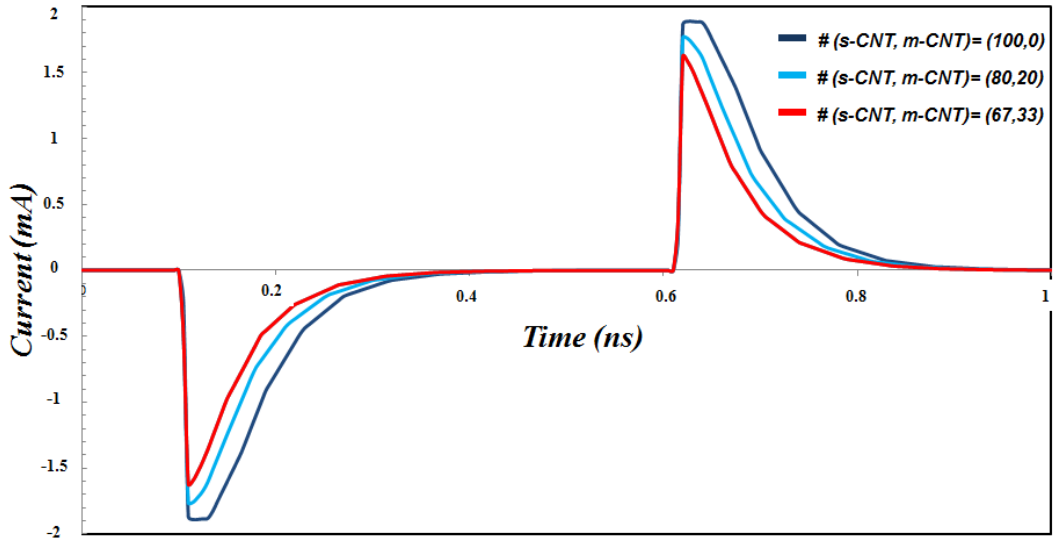


Figure 3.6. Effects of m-CNTs on I_{Drive} in (3-16) during discharging and charging of the output capacitance.

Increasing the number of m-CNTs results in decreasing of ΔV and I_{Drive} in (3-17). Since reduction of I_{Drive} is normally more than reduction of ΔV due to presence of m-CNTs, the gate delay typically increases by increasing the number of m-CNTs. The amount of increasing in delay is given by subtracting (3-17) from (3-14), which yields:

$$\Delta t_{pd} = \frac{C_{Load} \times V_{LS}/2}{(N_T - N_M) \times I_{ON,1}} - \frac{C_{Load} \times VDD/2}{N_T \times I_{ON,1}} . \quad (3\text{-}18)$$

Assuming the difference between V_{LS} and V_{DD} for small P_M is negligible, (3-18) can be simplified to:

$$\Delta t_{pd} = \frac{C_{Load} \times VDD/2}{N_T \times I_{ON,1}} \times \frac{P_M}{1 - P_M} . \quad (3\text{-}19)$$

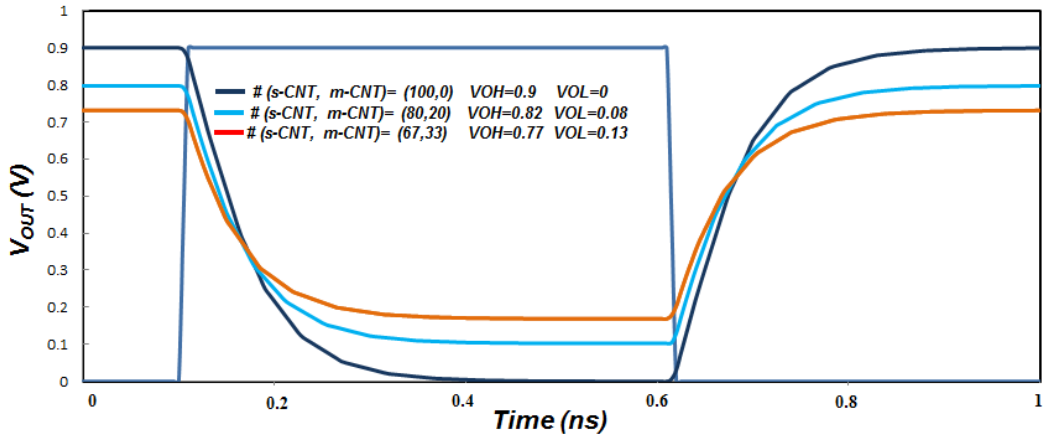


Figure 3.7. Simulated effects of m-CNTs on inverter gate delay driving a constant load.

3.4 Impact of m-CNTs on power consumption

Similar to CMOS circuits, the power dissipation in CNFET circuits consists of mainly dynamic and static components. Dynamic power dissipation is due to charging and discharging of load capacitances as the gate switches, while current leakage through imperfect transistors leads to static or leakage power dissipation.

To determine the static power dissipation of a CNFET inverter, first the DC currents supplied by V_{DD} for the high and low output, which denoted by $I_{DD}(OH)$ and $I_{DD}(OL)$, must be determined. Equation (3-20) gives the static power dissipation, based on the DC power supply currents, $I_{DD}(OH)$ and $I_{DD}(OL)$ [73]

$$P_{Static} = \frac{I_{DD}(OH)+I_{DD}(OL)}{2} V_{DD}, \tag{3-20}$$

where in an ideal CNFET we have

$$I_{DD}(OH) = I_{DD}(OL) \approx 0 . \tag{3-21}$$

Thus the static power dissipation in an ideal CNFET inverter is negligible. However, in the presence of m-CNTs there is a constant current flowing from V_{DD} to ground. The amount of this leakage current is proportional to the number of m-CNTs in pull-up and pull-down network, *i.e.*

$$I_{DD}(OH) = I_{DD}(OL) = N_M \times I_{ON,1}, \quad (3\text{-}22)$$

where N_M is the number of m-CNTs in pull-up and pull-down transistors. The static power dissipation for an imperfect CNFET inverter as a function of the number of m-CNTs is determined by substituting (3-22) into (3-20):

$$P_{Static} = N_M \times I_{ON,1} \times V_{DD} . \quad (3\text{-}23)$$

Equation (3-23) confirms that m-CNTs severely aggravate static power consumption, which may not be acceptable in many applications.

Similar to CMOS inverter, the dynamic power dissipated in a CNFET inverter is equal to:

$$P_{Dynamic} = \frac{1}{2}\alpha f_c C_L V^2{}_{Max} \quad (3\text{-}24)$$

where C_L is the load capacitance, α is the activity factor, f_c is the clock frequency, and V_{Max} in the maximum voltage swing in the load capacitor, which is equal to V_{LS} in our CNFET inverter. Using (3-8) for V_{LS} the dynamic power dissipation in CNFET gate can be evaluated by

$$P_{Dynamic} = \frac{1}{2}\alpha f_c C_L {V_{LS}}^2 . \quad (3\text{-}25)$$

3.5 CNFET VLSI system in the presence of m-CNTs

The analytical models for output high, output low, and logic swing voltages as well as delay, static and dynamic power dissipations presented in (3-6), (3-7), (3-9), (3-17), (3-23), and (3-25) provide a great tool in understanding of the effect of m-CNTs on CNFET systems.

The V_{OH}, V_{OL}, and V_{LS} of a CNFET inverter as a function of the probability of m-CNTs, P_M, are shown in Figure 3.8. Assuming that for a "reliable" inverter the V_{LS} must be at least $V_{DD}/2$, with Vt=0.1862V, then the maximum allowable P_M=53%.

$$\frac{V_{DD}}{2} = V_{DD} - 2(V_{DD} - V_t)\left(1 - \sqrt{1 - \frac{N_M}{N_T}}\right) \Rightarrow \frac{N_M}{N_T} = 0.53\ . \qquad (3\text{-}26)$$

The delay of a typical CNFET inverter as a function of P_M is shown in Figure 3.9. As shown in this figure, at first the delay increases by increasing of P_M. This is due to the presence of P_M in the numerator of the fraction in (3-19). At some point the delay decreases due to the 1-P_M term in the denominator of the function in (3-19), when P_M becomes very large, beyond 50%.

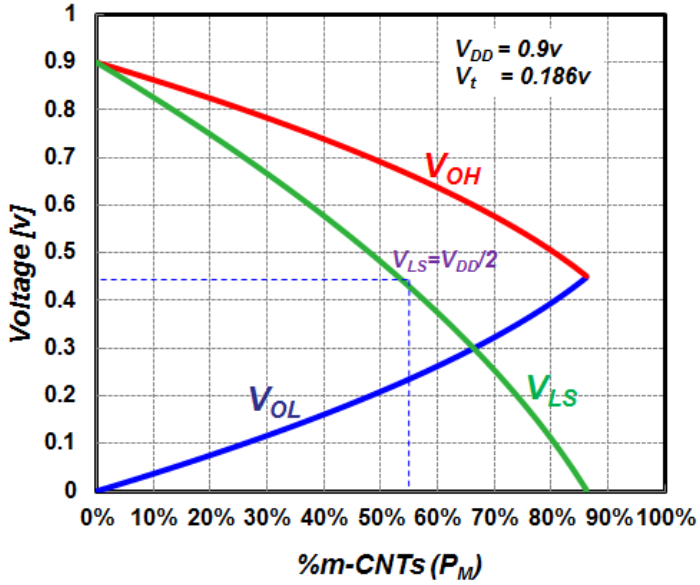


Figure 3.8. V_{OH}, V_{OL}, and V_{LS} of a CNFET inverter as a function of the probability of m-CNTs.

The increasing of path delay due to m-CNTs can be analytically calculated by using (3-19). For example if 15 gates exists per path, and only 10% increases in delay is acceptable, by using (3-19) the maximum tolerable P_M is:

$$0.1 = 15 \times \frac{P_M}{1 - P_M} \Rightarrow P_M = 0.0066. \qquad (3\text{-}27)$$

The static and dynamic power dissipation in the CNFET inverter is shown in Figure 3.10-a, where it clearly confirms that even a very small amount of m-CNTs can significantly ruin the static power dissipation, while the dynamic power stays almost unchanged.

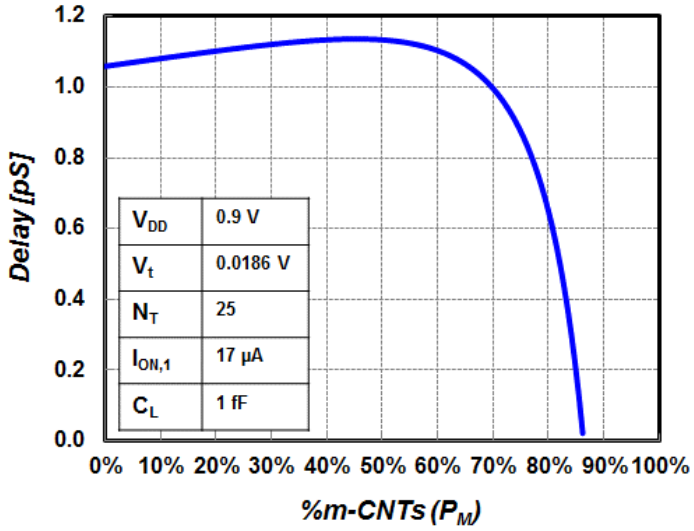


Figure 3.9. CNFET inverter delay as a function of the probability of m-CNTs.

To determine an acceptable range of P_M based on the static power, the magnified version of Figure 3.10-a is shown in Figure 3.10-b. As the plot of Figure 3.10-b shows, P_M>0.06% is unacceptable, because within this range the static power exceeds the dynamic power. Using our presented analytical models and based on the previous examples, it is shown that the static power dissipation is a more critical issue than the delay and the dynamic power of CNFET circuits in the presence of m-CNTs.

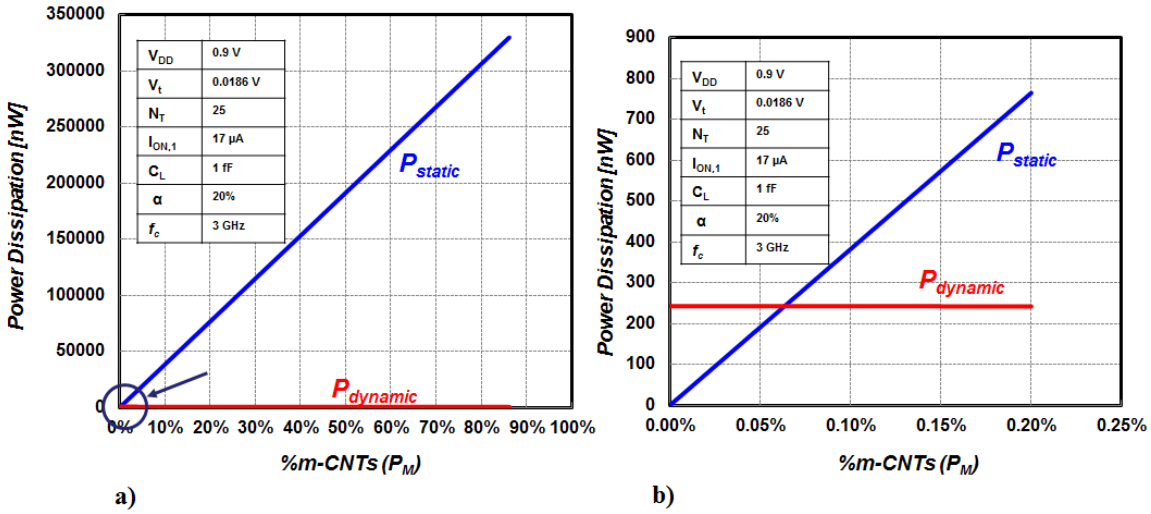


Figure 3.10. Static and dynamic power dissipation in CNFET inverter as a function of the probability of m-CNTs

Chapter 4

4 Fault analysis of CNFET VLSI system in the presence of CNT density variation and Metallic CNTs

As semiconductor process geometries shrink to nanometer regime, the transistors and interconnects become exceedingly susceptible to failure due to their physical limitations. Implementing a resilient circuit using imperfect nanoelectronic devices, such as CNFETs, requires a detailed failure-based design optimization and cell characterization integrated into the Electronic Design Automation (EDA) tools. In this chapter, we developed a basic method of cell failure characterization for failure-based design optimization in nanoelectronic EDA tools. Based on the probability of "open" and "short" defect in CNFET transistors, two types of failure models were developed in section 4-1; 1) input vector-based failure probability, and 2) worst case failure probability.

The worst case logic gate probability of failure is the primary mathematical model that can be used for chip-level failure analysis in EDA tools. The transistor-level redundancy can be easily incorporated into the model to compute the failure probability of gates with composite array of transistors. Also, an integrated and compact model for probability of failure in CNFETs that includes 1) void CNFETs, 2) CNT density variation, and 3) m-CNTs is presented based on binomial probability distribution. Comparison with experimental data shows that the compact model successfully predicts the failure probability in CNFET devices. Moreover, the model is used in a new design space to

explore tradeoffs, key limitations, and opportunities for today's Gigascale CNFET integrated systems. Utilizing the compact model to determine the probability of failure, we proposed guidelines for CNFET manufacturing processes as well as for CNFET standard cell development to ensure that the device operates at its optimal design point. The guidelines include the maximum allowable probability of m-CNT, and the minimum number of CNTs in the CNFET channels, to achieve a target yield.

4.1 Logic gate characterization for failure

As we move toward revolutionary nanoelectronic devices, such as quantum, molecular transistors, nanowire and CNFET, we anticipate less reliable devices [12]. For instance, in comparison to the 10^{-9} to 10^{-7} failure rates in CMOS technology [78], the failure rates in emerging nanotechnologies are projected to be in the order of 10^{-2} to 10^{-1} due to their size and imperfection synthesis process [79-81]. It is, therefore, imperative to accept the fact that the underlying hardware will no longer be perfect, and enable design of robust systems that are resilient to hardware imperfections [82].

Manufacturing failures can be codified into two categories: "open" and "short" defects. In CNTFET fabrication process, m-CNTs make the CNFET a short circuit device and CNT density variation may cause an always open CNFET when there is no CNT under the device channel. Therefore, we consider two types of failure probabilities for nanoelectronic devices: probability of short defects, P_s, and probability of open defects, P_o.

Probabilities of short and open defects are independent quantities, since short and open defects are caused by independent process defects. Probability of short can be thought of as the probability of failure for an "off" transistor, $P_f(\text{off}) = P_s$. Similarly, probability of open can be thought as of the probability of failure for an "on" transistor, $P_f(\text{on}) = P_o$.

Logic gate characterization is the basic components of standard cell ASIC design flow in current EDA tools, where the delay of standard cells are pre-calculated for whole chip timing analysis. Traditionally, gate characterization simply provided a look up table of cell delays as a function of load capacitances to be used in static timing analysis (STA).

However, in order to perform "static failure analysis" (SFA), as proposed in this section, failure characterization must be included into logic gate characterization. This section presents a simplified model to characterize logic gate failure based on the probability of short defects, P_s, and probability of open defects, P_o.

4.1.1 Probability of failure and transistor sizing

To derive the probability of failure in a logic gate, first we analyze the probability of failure in a transistor with size *N*. Assume that the probability of short defects, P_s, and probability of open defects, P_o, is defined for a unit size transistor (size 1×). As shown in Figure 4.1, a 2× size transistor can be represented by two transistors in parallel. In this case, the overall transistor fails as an open circuit, only if both left and right transistors have open failures. Therefore, the composite probability of open failure, $\hat{P}_O$, can be determined by:

$$\hat{P}_O = P_O \times P_o = {P_O}^2 \ . \tag{4-1}$$

Similarly, the probability of short failure in the 2× transistor can be determined by considering that the overall transistor succeed (does not fail as a short circuit), only if both left and right transistors do not have short failure. Therefore, the composite probability of short failure, $\hat{P}_S$, can be determined by:

$$1 - \hat{P}_S = (1 - P_S) \times (1 - P_S) = (1 - P_S)^2 \ \Rightarrow \ \hat{P}_S = 1 - (1 - P_S)^2 \ . \tag{4-2}$$

Equations (4-1) and (4-2) can be easily extended for *N*× size transistor as shown in Figure 4.1.

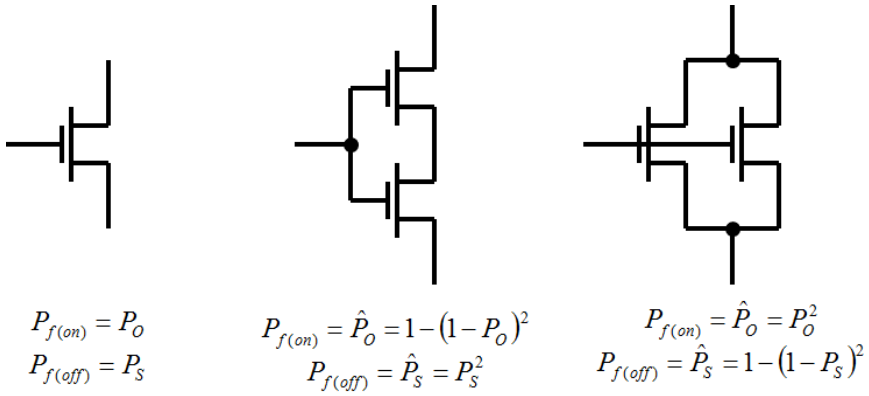


Figure 4.1. Probability of failure for *N*× transistor [83].

4.1.2 Input vector-based failure analysis

In general, the probability of failure in a logic gate depends on the input vector. For instance, if the input vector turns on a transistor in which is defectively short, the logic gate doesn't experience any failure. To develop a model for a logic gate failure, consider a simple inverter circuit shown in Figure 4.2, where the composite probability of open and short failures form N-type field effective transistor (NFET) are $\hat{P}_{O,n}$, and $\hat{P}_{S,n}$, and the composite probability of open and short failures form P-type field effective transistor (PFET) are $\hat{P}_{O,p}$, and $\hat{P}_{S,p}$, respectively.

When the input of the inverter shown in Figure 4.2 is at zero, the inverter succeeds (does not fail) only if the NFET is not short and PFET is not open to maintain output at V_{DD}. This can be presented by:

$$1 - P_f(0) = \left(1 - \hat{P}_{S,n}\right)\left(1 - \hat{P}_{O,p}\right) \quad \Rightarrow \quad P_f(0) = 1 - \left(1 - \hat{P}_{S,n}\right)\left(1 - \hat{P}_{O,p}\right), \quad (4\text{-}3)$$

where $P_f(0)$ is the probability of failure, when the input of the inverter is at zero, and $\hat{P}_{S,n}$, and $\hat{P}_{O,p}$, are the composite probability of short failure in NFET and open failure in PFET, respectively. Similarly, when the input of the inverter shown in Figure 4.2 is at V_{DD}, the inverter succeeds (does not fail) only if the NFET is not open and PFET is not short to maintain output at zero. This can be presented by:

$$1 - P_f(1) = (1 - \hat{P}_{O,n})(1 - \hat{P}_{S,p}) \quad \Rightarrow \quad P_f(1) = 1 - (1 - \hat{P}_{O,n})(1 - \hat{P}_{S,p}), \quad (4\text{-}4)$$

where $P_f(1)$ is the probability of failure when the input of the inverter is at V_{DD}, and $\hat{P}_{O,n}$, and $\hat{P}_{S,p}$, are the composite probability of open failure in NFET and short failure in PFET respectively.

Likewise, the probability of failure for a NAND gate as a function of input vector can be derived as shown in Figure 4.3. Depending on the input vector, the NAND gate maintains the correct output logic (does not fail) only if the transistors in the paths are properly functional. Similarly the probability of logic gate failure can be determined for any logic gate as a function of its input vector.

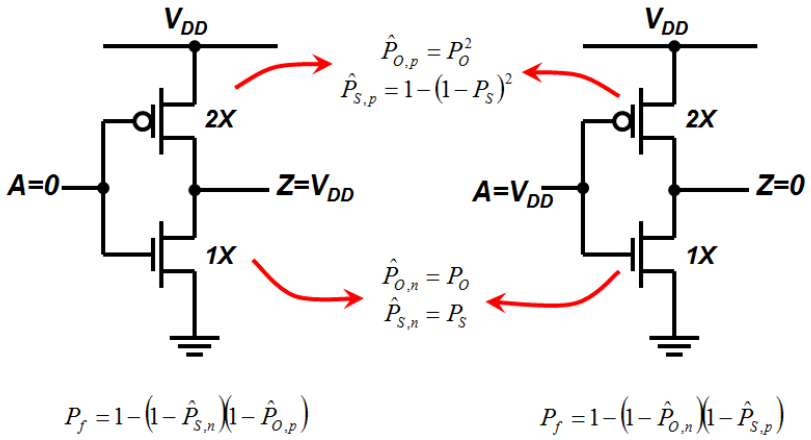
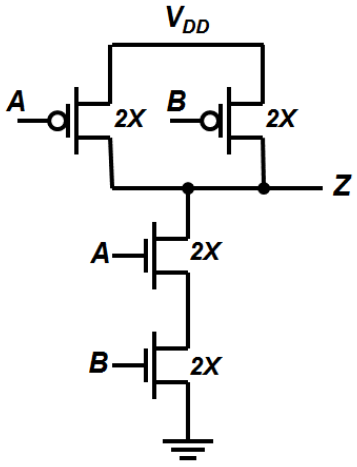


Figure 4.2. Probability of failure in an inverter when input is 0 and 1 [83].



A	B	Z	Probability of Failure
0	0	1	$P_f = 1-(1-\hat{P}_{S,n}^2)(1-\hat{P}_{O,p}^2)$
0	1	1	$P_f = 1-(1-\hat{P}_{S,n})(1-\hat{P}_{O,p})$
1	0	1	$P_f = 1-(1-\hat{P}_{S,n})(1-\hat{P}_{O,p})$
1	1	0	$P_f = 1-\left[(1-\hat{P}_{O,n})^2\right]\left[(1-\hat{P}_{S,p})^2\right]$

$$\hat{P}_{O,n} = \hat{P}_{O,p} = P_O^2$$
$$\hat{P}_{S,n} = \hat{P}_{S,p} = 1-(1-P_S)^2$$

Figure 4.3. Probability of failure in a NAND gate for various input vectors [83].

4.1.3 Worst case failure analysis

Although, in general the probability of failure in a logic gate is a function of its input vector, in practice the logic gate should be functional for any input vector to ensure the reliability of the whole system. Therefore the logic gate succeeds (does not fail) only if all the transistors in a logic gate have no open defect and have no short defect. For the inverter in Figure 4.2, the worst case probability of failure, P_f, is given by:

$$P_f = 1 - \left(1 - \hat{P}_{S,n}\right)\left(1 - \hat{P}_{O,n}\right)\left(1 - \hat{P}_{S,p}\right)\left(1 - \hat{P}_{O,p}\right) . \quad (4\text{-}5)$$

Similarly, for the NAND gate in Figure 4.3, the worst case probability of failure, P_f, is given by:

$$P_f = 1 - \left(1 - \hat{P}_{S,n}\right)^2\left(1 - \hat{P}_{O,n}\right)^2\left(1 - \hat{P}_{S,p}\right)^2\left(1 - \hat{P}_{O,p}\right)^2 . \quad (4\text{-}6)$$

In this chapter, we consider the worst case probability of failure for logic gate characterization.

Figure 4.4 illustrates the worst case probability of failure for various sizes of inverter, NAND, and NOR gates using CNFET process technology, where the probability of short defects (due to m-CNT), P_S, is assumed to be 2% and the probability of open defects (due to CNT density variation), P_O, is assumed to be 1%. As shown in this figure, the probability of failure in NAND gates is higher than that in inverters of the same size. Also, the probability of failure in NOR gates is higher than that in NAND gates of the same size. As the size of the gate increases the probability of failure also increases.

The data in Figure 4.4 shows that the probability of failure in logic gates using CNFET is prohibitively large (for instance 50% for a 4x NAND gate). This suggests that the gates using CNFET devices are not practically reliable if they are simply connected to construct a logic gate.

Ps = 0.02 and Po = 0.01

Size	INV	NAND	NOR
1x	0.07	0.15	0.20
2x	0.11	0.28	0.33
3x	0.17	0.38	0.45
4x	0.22	0.48	0.55
5x	0.26	0.55	0.64
6x	0.30	0.62	0.70
7x	0.35	0.68	0.76
8x	0.38	0.73	0.80
9x	0.42	0.77	0.84
10x	0.45	0.80	0.87

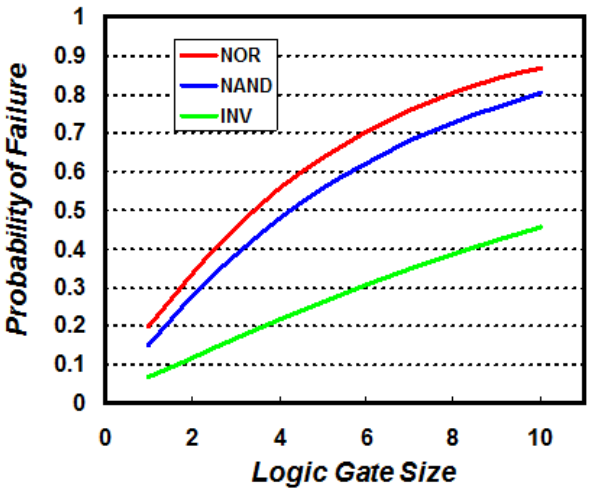


Figure 4.4. Probability of failure in an INV, NAND, and NOR gate in CNFET technology [83].

4.2 Transistor-level redundancy for CNFET

Utilizing unreliable logic elements for reliable computing is a concept that dates back to Von Neumann's seminal work in 1956, where he studied redundancy techniques to increase reliability of faulty logic gates [84]. Device-level redundancy also became a method for realizing fault tolerant system in 1956, where Moore *et.al*. proposed a method to design realizable circuits with unrealizable devices [49, 50].

Transistor-level redundancy is used in CNFET devices to enhance the transistor reliability. To improve the open defect rate, CNFETs are built in a bundle form as shown in Figure 4.5-a [21, 38]. Moreover, to improve the short defect rate due to m-CNT, multiple bundles of CNFETs are connected in series as shown in Figure 4.5-b [35-37, 39].

Transistor-level redundancy is based on series and parallel connections of devices as shown in Figure 4.7. To develop a model for such complex redundant circuit, we start with a basic 2-transistor series and parallel connections shown in Figure 4.6, considering that the probability of short defects for a single transistor is P_s, and the probability of open defects for a single transistor is P_o.

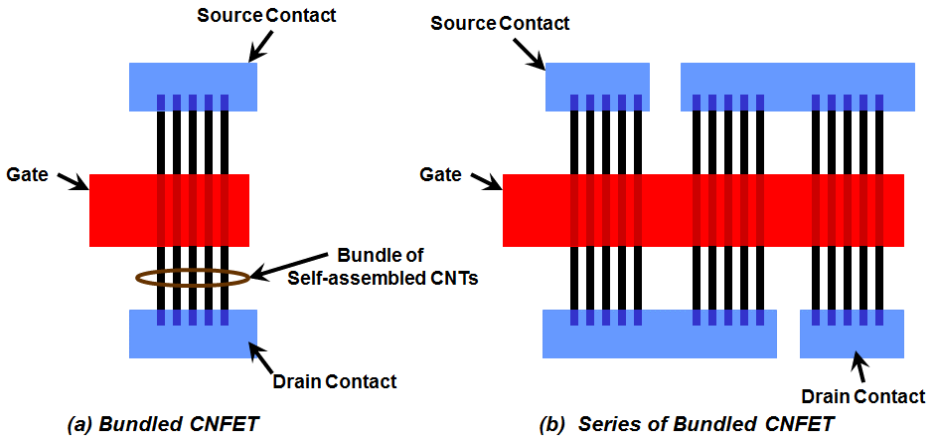


Figure 4.5. Bundled CNFET and series of bundled CNFET structures [83].

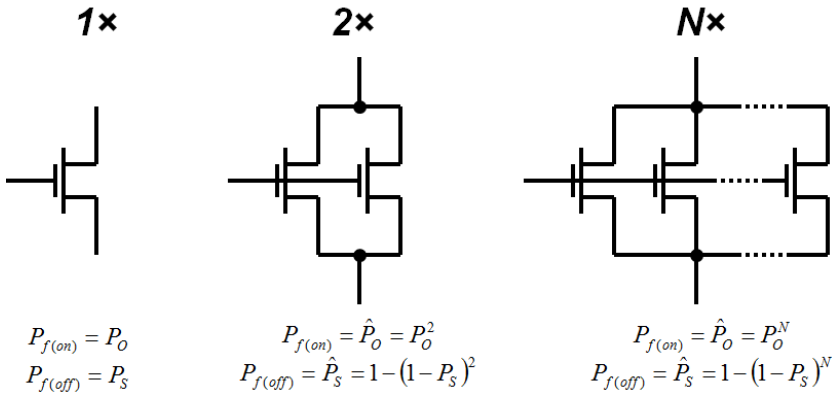


Figure 4.6. Series and parallel redundancy in basic CNFET transistors [83].

The probability of failure for parallel connection of transistors is equivalent to the transistor sizing model explained in section 4.1, where the composite probability of short and open defects can be computed using (4-1) and (4-2). The probability of series connection of transistors, however, can be derived in a similar fashion. Refer to Figure 4.6, the composite series transistor fails as a short circuit, only if both top and bottom transistors have short failures. Therefore, the composite probability of short failure, $\hat{P}_S$, can be determined by:

$$\hat{P}_S = P_S \times P_S = {P_S}^2 \ . \tag{4-7}$$

Similarly, the probability of open failure in the series transistors can be determined by considering that the composite transistor succeeds (does not fail as a short circuit), only if

both top and bottom transistors do not have open failure. Therefore, the composite probability of open failure, $\hat{P}_O$, can be determined by:

$$1 - \hat{P}_o = (1 - P_o) \times (1 - P_o) = (1 - P_o)^2 \Rightarrow \hat{P}_o = 1 - (1 - P_o)^2 . \quad (4\text{-}8)$$

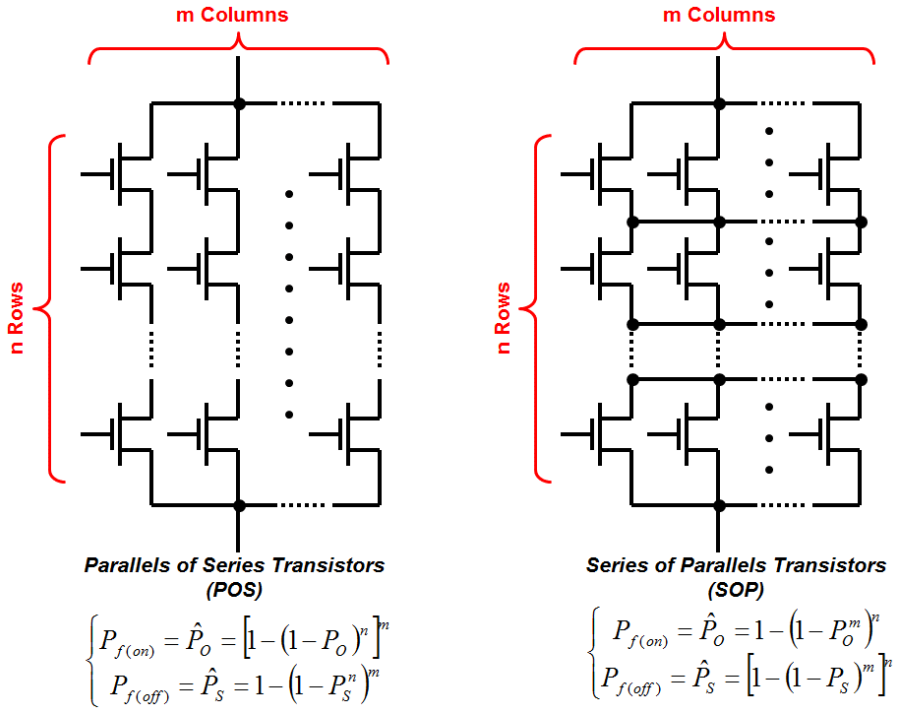


Figure 4.7. Series and parallel transistor-level redundancy to improve CNFET reliability [52, 83].

In general, the composite probability of open and short failures for series-of-parallel (SoP) and parallel-of-series (PoS) constructions can be derived as shown in Figure 4.7. For CNFET process integration, a series connection of bundled CNFET (or SoP configuration) is used to overcome the issue with high defective rates [83].

4.3 Failure analysis for single-transistor CNFET

The process of CNT growth always results in random and undetermined CNT placement. In chapter 2, we derived the CNT count probability density distribution, (2-5),

$$f_{CNT}(k) = \frac{(2\bar{N}_{CNT})!}{k!(2\bar{N}_{CNT}-k)!}(0.5)^{2\bar{N}_{CNT}} , \quad (4\text{-}9)$$

where $f_{CNT}(k)$ is the probability of having k CNTs under the CNFET gate, and $\bar{N}_{CNT}$ is the expected number of CNTs. The CNT count probability distribution in (4-9) when k=0 gives the probability of a void CNFET, or an open defect, P_O,

$$P_O = (0.5)^{2\bar{N}_{CNT}} . \quad (4\text{-}10)$$

Equation (4-10) provides a *fundamental limit* on probability of open defects in CNFET as a function of expected number of CNTs. However, a CNFET may also become defective when at least one m-CNT appears under the gate. In this section, we derive the overall failure model that includes both short and open defects.

Note that in this chapter, to have a better numerical representation and easier computation, we always use probability of failure instead of yield. It is easier to represent 10^{-9} or 1 part-per-billion (1 PPB) for probability of failure rather than 0.999999999 for yield.

4.4 Derivation of CNFET probability of failure

For a CNFET with N_{CNT} in the channel, the device is "not short" only when all N_{CNT} carbon nanotubes are not metallic. Therefore, if the probability of short CNFET is, P_S, then in a CNFET with N_{CNT} carbon nanotubes in the channel we have:

$$1 - P_S = (1 - P_M)^{N_{CNT}} \quad \Rightarrow \; P_S = 1 - (1 - P_M)^{N_{CNT}} , \quad (4\text{-}11)$$

where P_M is the probability of m-CNT occurrence. P_M can be either measured through experiments or predicted from bandgap stochastic models developed in [85], where the probability of zero bandgap represents the probability of m-CNTs. Figure 4.8 uses (4-11) to illustrate how the probability of short failure increases as the number of CNTs in the channel of a CNFET increases.

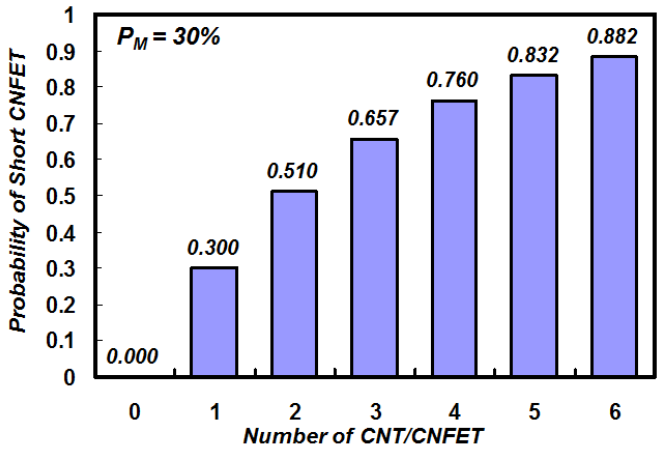


Figure 4.8. Probability of short CNFET distribution when the probability of m-CNT is set to be 30%.

Equation (4-11) has been used in many previous publications [35-37, 49–51] to compute the probability of short defects for CNFET. However, as discussed in Section 2-1, once we choose the CNFET device width W over the CNT density of D due to the random CNT placement, the exact number of CNTs in the channel of a CNFET may not always be $N_{CNT} = W.D$. Therefore, (4-11) does not count for CNT density variations. To include the CNT density variation and more accurately compute the average probability of short defects, we use the CNT count distribution in (4-9) in conjunction with the probability of short defects in (4-11). Since these two probabilities are independent, we can find the average probability of short defect by:

$$\bar{P}_S = \sum_{k=1}^{2N_{CNT}} f_{CNT}(k) P_S(k),$$

$$= \sum_{k=1}^{2N_{CNT}} \frac{(2\bar{N}_{CNT})!}{k!(2\bar{N}_{CNT}-k)!} (0.5)^{2\bar{N}_{CNT}} [1-(1-P_M)^k]. \quad (4\text{-}12)$$

The total probability of CNFET failure is simply the sum of the probabilities of open and short defects. This is true because the short and open defects are mutually exclusive. A short failure and an open failure cannot occur at the same time. Therefore, the total probably of failure in a single CNFET can be determined by

$$P_f = P_o + \bar{P}_s = (0.5)^{2N_{CNT}} + \bar{P}_S, \quad (4\text{-}13)$$

where P_O is the probability of open defect from (4-10) and $\bar{P}_s$ is the average probability of short defect from (4-12). Equation (4-13) is a complete and compact model for CNFET probability of failure that includes: 1) probability of open defects, 2) probability of short defects (due to m-CNT), and 3) CNT count density variations [52].

4.4.1 Single-transistor CNFET optimization and tradeoffs

In Section 2.1, it was intuitively discussed that there is a tradeoff for choosing the CNFET window size W. Due to the random placement of CNTs, too small a window results in more open defects, and too large a window results in more CNTs and, therefore, more short defects. Equation (4-13) precisely confirms this tradeoff.

Figure 4.9 illustrates the probability of CNFET failure and its components: the probability of open and short defects versus the expected number of CNTs in the channel based on the compact model given in (4-13). It is assumed that the probability of a m-CNT occurrence, P_M, is 30%, which is typical in current technology without m-CNTs removal processing [86].

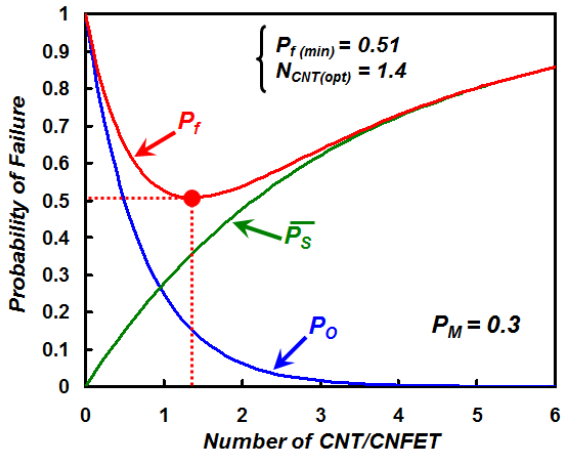


Figure 4.9. Probability of CNFET failure and its components: probability of open and short defects versus the expected number of CNTs in the channel based on the compact model in (4-13) [52].

As shown in Figure 4.9, for a small number of CNTs, the probability of open defect is dominant, and for a large number of CNTs, the probability of short defect is dominant. In

this particular case, when P_M= 30%, the minimum probability of failure is 51%, which occurs at the optimum point of $\overline{N}_{CNT}$=1.4 CNTs in the channel. This means that the optimum CNFET channel width is 0.14μm in CNT growth technology, where the achievable CNT density is about 10 CNT/μm. Based on Figure 4.9, the probability of failure in a single-transistor CNFET cannot be better than 51%, which is certainly unacceptable for any VLSI design application today.

However, to achieve a reasonable reliability in a single transistor CNFET, P_M must be significantly reduced. Table 4.1 represents how reduction of P_M affects the optimum CNFET failure rate. It also shows the optimum expected number of CNTs in the channel $\overline{N}_{CNT}$ to obtain the optimum point.

Table 4.1: Optimum number of CNTs and minimum probability of failure for various probabilities of m-CNTs[52]

P_M	$N_{CNT}(OPT)$	$P_f(min)$
0.3	1.4	0.51
0.1	2.0	0.248
0.001	5.0	0.00597
1.00E-09	15.0	1.59E-08
1.00E-10	17.0	1.76E-09

4.4.2 Single-transistor CNFET for Gigascale integration

Figure 4.10 illustrates the optimum point for a single-transistor CNFET to obtain 1 PPB failure rate to be used in a Gigascale integrated system with one billion transistors. Based on the model presented in this section, the failure rate of 1 PPB can only be achieved when P_M=5.6×10^{-11} and $\overline{N}_{CNT}$=17.5. With the current growth technology, it is impractical to

obtain such a low probability of m-CNT occurrence as it requires precise control over the nanotube chirality during growth and excessive m-CNT removal after growth [88].

Moreover, Figure 4.10 shows that even if P_M can be reduced to such a low level, the probability of open defects becomes a critical issue for CNFETs in today's Gigascale integration, where a large number of CNTs will be required. In this example, to reach the failure rate of 1 PPB, the expected number of CNTs in the CNFET channel becomes 17.5. This means that the CNFET channel width must be 1.75 µm in CNT growth technology, where the achievable CNT density is about 10 CNT/µm. This will also impose a large area penalty if today's CNT growth technology is used.

Note that the targeted failure rate of 1 PPB can result indifferent Gigascale chip yields. If all the transistors are made "uncorrelated," then the total chip failure rate is

$$P_f(chip) = 1 - \left(1 - P_f\right)^N = 1 - (1 - 10^{-9})^{10^9} = 63\%, \qquad (4\text{-}14)$$

where N is the number of single-transistor CNFETs in the chip (i.e., one billion), and P_f is the failure rate of a CNFET (i.e., 1 PPB).

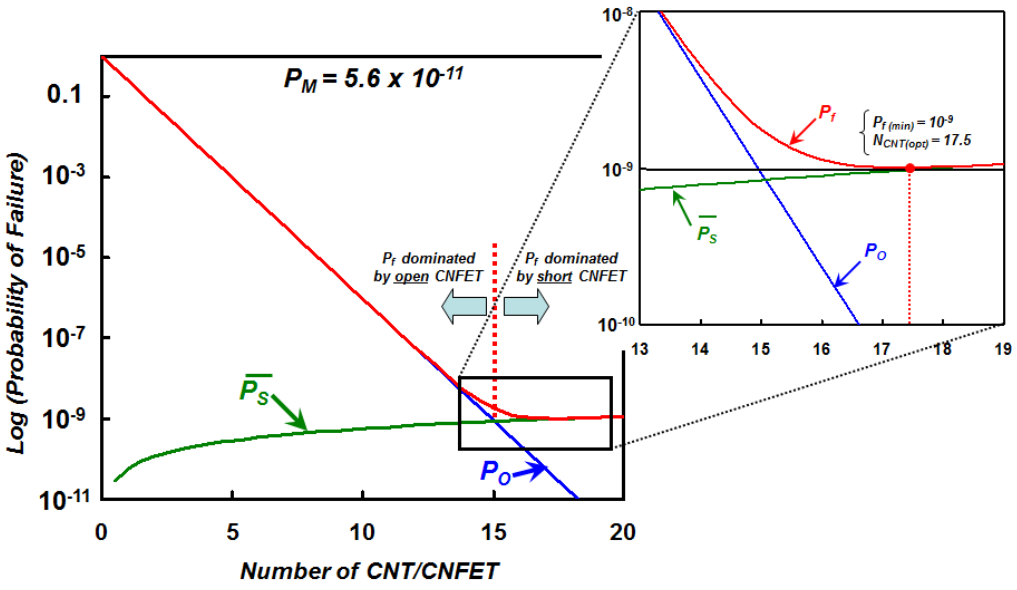


Figure 4.10. Optimum point for a single-transistor CNFET to obtain 1-PPB failure rate based on the compact model in (4-13) [52].

The total chip yield can be increased by making some of the CNFETs "correlated" [35-37]. Assuming that at least 100 CNFETs can be made correlated [35-37], then the total chip failure rate is

$$P_f(chip) = 1 - \left(1 - P_f\right)^{N/S} = 1 - (1 - 10^{-9})^{10^9/100} \approx 1\%, \qquad (4\text{-}15)$$

where S is the number of correlated CNFETs that fit along the length of a group of CNTs and it is assumed to be 100 in this example.

To illustrate the correlated and uncorrelated CNFET implementations, an example of a CNFET circuit is shown in Figure 4.11. The CNFETs that are placed on the same bundle of CNTs are correlated [Figure 4.11 (b)], whereas the CNFETs on different bundles are uncorrelated [Figure 4.11 (c)].

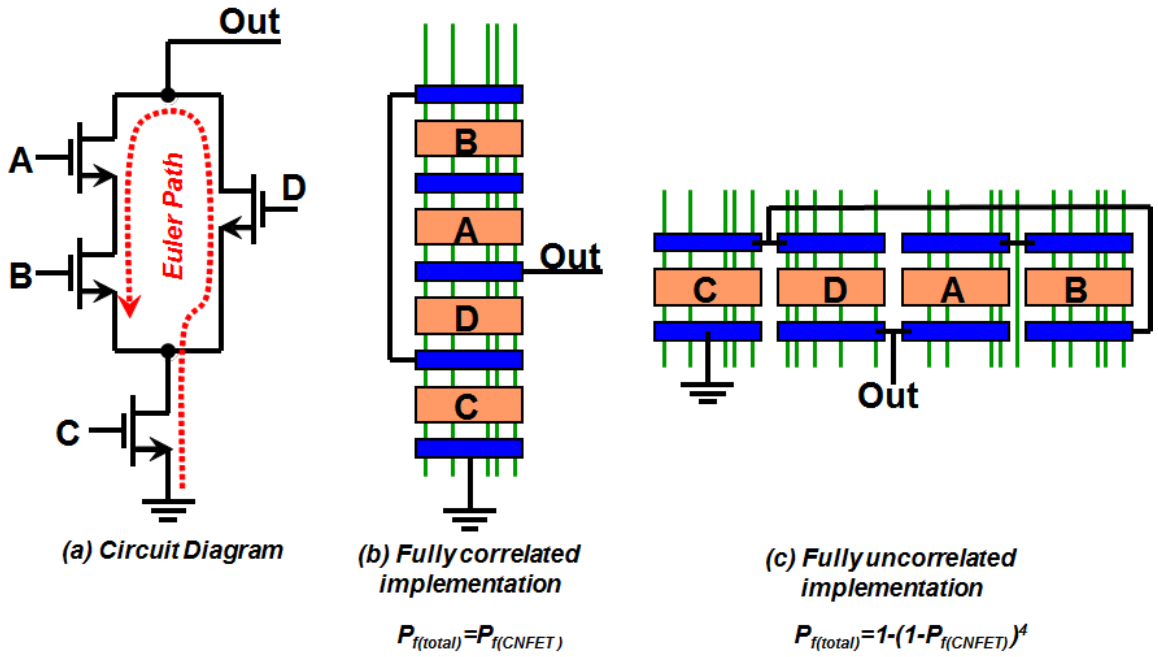


Figure 4.11. Correlated and uncorrelated CNFET implementation of a circuit. (a) Circuit diagram. (b) Fully correlated implementation $P_{f(total)} = P_{f(CNFET)}$. (c) Fully uncorrelated implementation $P_{f(total)} = 1 - (1 - P_{f(CNFET)})^4$ [52].

4.5 Device limit in single-transistor CNFET

Assuming that all the technological constraints in CNT fabrication can be eliminated, the probability of failure in a single transistor CNFET will be governed by its key limitation, which is the probability of void CNFET, (4-10). This is true because the growth of CNTs always results in a random and non-uniform placement. Without the possibility of precise CNT placement under the gate of CNFET, the probability of void CNFET will remain a key limitation for the reliability of single-transistor CNFET. Therefore, even if all m-CNTs can be removed (P_M=0), then the probability of failure in a single-transistor CNFET will eventually become

$$P_f(ideal) = P_o = (0.5)^{2\bar{N}_{CNT}}, \tag{4-16}$$

where $\bar{N}_{CNT}$ is the expected number of CNTs in the channel of CNFET. Figure 4.12 shows that the probability of failure approaches to P_O from (4-13), as P_M approaches zero.

Based on this limitation, to get a 1 PPB failure rate in an ideal single-transistor CNFET, it must contain at least 15 CNTs in the channel. This projection could not be found without the CNT count distribution model developed in Section 2.1. With the minimum expected number of 15 CNTs, a lower bound on the smallest CNFET width W can be predicted, assuming the highest possible density of CNTs. It is shown that as CNTs are brought closer below a certain range (~<3−4 nm), the electrostatic nanotube–nanotube interactions lead to reduction in current drive and ultimately increase in delay of the CNFET [89]. Based on the results shown in [89], at the density of 1 CNT/4 nm or D=250 CNT/μm, the charge screening effect results in about 5% increase in CNFET delay. Assuming that D=250 CNT/μm is an acceptable maximum density limit and considering that the CNFETs contains the minimum of 15 CNTs, then the device limit for a minimum CNFET width will

be 60nm for an ideal single transistor CNFET device projected for a Gigascale integrated system.

Note that this lower bound on CNFET device width is independent of the CNFET technology node (CNFET channel length), because the constraint is due to random placement of CNTs only, and not on fabrication defects that exist even in today's standard CMOS technology process.

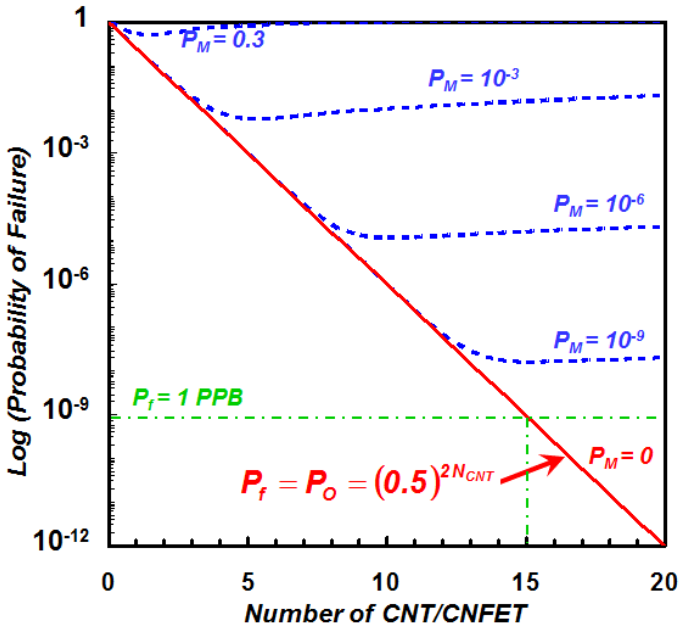


Figure 4.12. Device limit in a single-transistor CNFET, where the probability of failure approaches the probability of open defect CNFET as P_M moves toward zero [52].

4.6 Design guidelines for optimum CNFET

Using (4-13), the probability of CNFET failure for three sets of probabilities of m-CNTs, P_M=0.3, 0.1, and 0.03 is shown in Figure 4.13. The optimum point for CNFET design is shown by a filled circle on these plots.

Now consider that in a standard cell library, it is required to design a CNFET with the target probability of failure of P_{Target}=0.25. Based on the graphs in Figure 4.13, it is not possible to achieve this target probability of failure when P_M=0.3. On the other hand, Figure

4.13 shows that the target probability of failure can be achieved when P_M=0.1 or less. Since fabrication of CNT with lower probability of metallic is harder, the optimum design in this example will be $P_{M(opt)}$=0.1 and N_{CNT}(opt)=2.0. This is a direct result of exercising (4-13).

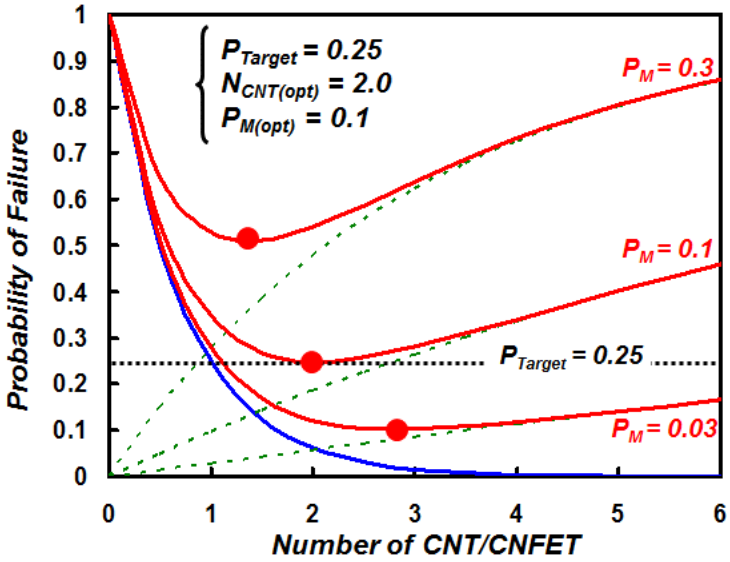


Figure 4.13. Probability of CNFET failure and the optimum points for P_M=0.3, 0.1, and 0.03 [52].

4.6.1 Derivation of optimum CNFET design point

Although (4-13) can be used to numerically find the optimum design point for CNFET, it is desirable to develop a closed-form model that can be used as a guideline for CNFET design in a given technology node. Equation (4-9) can be approximated by using the expected value of number of CNTs in the channel of CNFET, N_{CNT}, instead of its density distribution. Hence, assuming that $P_M << 1$, we can approximate $\bar{P}_S$ as:

$$\bar{P}_S \approx 1-(1-P_M)^{N_{CNT}} \approx 1-(1-N_{CNT}\,P_M) = N_{CNT}\,P_M \ . \qquad (4\text{-}16)$$

Therefore, the overall probability of failure in (4-13) can be approximated by:

$$P_f = P_o + \bar{P}_s \approx (0.5)^{2N_{CNT}} + N_{CNT}\,P_M \ . \qquad (4\text{-}17)$$

The optimum point can then be found when the derivative of P_f in (4-17) is zero, or

$$\frac{\partial P_f}{\partial N_{CNT}} = (0.5)^{2\,N_{CNT}} Ln(0.25) + P_M = 0\ . \tag{4-18}$$

Equation (4-18) along with the equation for the target probability of failure in (4-17) can be used to determine the optimum N_{CNT} and P_M as shown below:

$$\begin{cases} (0.5)^{2N_{CNT}} + N_{CNT} P_M = P_{Target} \\ \dfrac{\partial P_f}{\partial N_{CNT}} = (0.5)^{2N_{CNT}} Ln(0.25) + P_M = 0 \end{cases} \tag{4-19}$$

The approximate solutions to the system of two equations and two unknowns in (4-19) can be further simplified as below:

$$\begin{cases} N_{CNT} \approx \dfrac{W\left(-P_{Target}\right)}{Ln(0.25)} \\ P_M \approx \dfrac{P_{Target}}{N_{CNT}} \end{cases} \tag{4-20}$$

where *W(x)* is Lambert W-Function defined below [90]:

$$f(x) = x.e^x \implies W(x) = f^{-1}(x). \tag{4-21}$$

Equation (4-20) shows that to get more reliable CNFET (*i.e.*, lower P_{Target}), larger device width and lower m-CNT density is required.

4.6.2 Comparison with exact numerical solutions

To verify the accuracy of the simplified model, the solutions given in (4-20) are compared to the exact numerical solutions to the model in (4-13) for a wide range of P_{Target}. The solid lines in Figure 4.14 represent the simplified model in (4-20), and the filled circles represent the exact numerical solutions to (4-13). Figure 4.14 demonstrates an excellent match for smaller P_{Target}, which is a more practical for VLSI applications.

4.7 Design guidelines for CNFET standard cell library

The optimum probability of m-CNT, P_M, from (4-20) can be used as a guideline for CNFET process manufacturing, and the optimum number of CNT under the CNFET gate, N_{CNT}, given by (4-20) can be used as a guideline in designing the standard cell library for CNFET design automation tools. The application of the proposed model is explored in the following experiment.

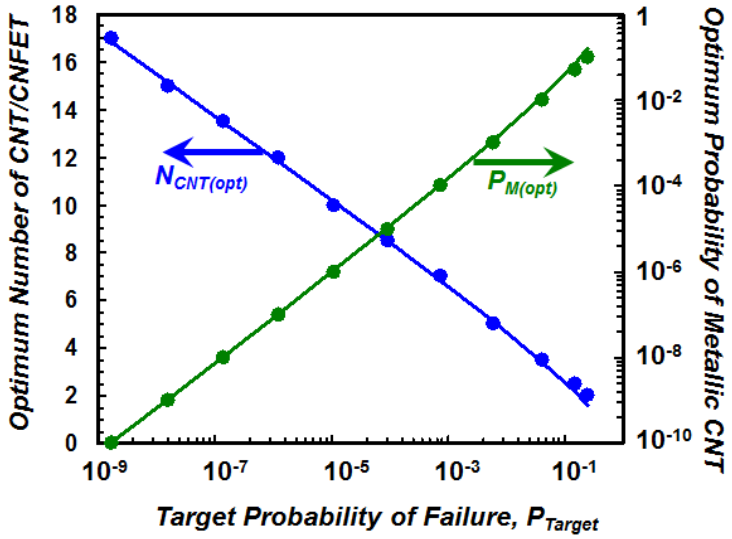


Figure 4.14. Comparison of the closed-form approximation model against exact numerical solutions.

4.7.1 Experimental results

An experimental design with 16,646 cells (133,172 CNFET devices) using the open source 45nm Nangate standard cell library [91] is chosen for this experiment. To obtain 90% yield, and considering that at least 100 CNFETs can be made to correlate as described in [53], the target probability of failure for each CNFET, P_{Target}, can then be determined by:

$$0.9 = (1 - P_{Target})^{133172/100} \quad \Longrightarrow \quad P_{Target} = 7.9 \times 10^{-5} \,. \qquad (4\text{-}22)$$

Equation (4-20) estimates that, for this target probability of failure, the probability of m-CNT must be at most P_M=9.2×10^{-6} and the number of CNT under the CNFET gate must

be N_{CNT}=8.6. This translates to a CNFET width of W=0.86µm if the CNT density D=10 CNT/µm.

Unfortunately, it may not be practical to make all CNFETs in the cell library exactly 0.86µm because the size of transistors in standard cell libraries varies significantly to optimize delay. Figure 4.15 shows the layout of three different cells, INVX1, NAND2X1, and DFFNEGX1, in the Nangate cell library. Even though the drive strength of all three cells is the same (X1), they contain a variety of transistor sizes ranging from 1× to 4×.

If we choose the 1× size CNFET to be 0.86µm in this experiment, then larger CNFETs (2× to 4× range) fall in non-optimal design points that result in larger probability of failure than P_{Target}. Therefore, the CNFET size variation in the standard cell library causes the total yield to be reduced.

To evaluate the impact of CNFET size variation on the total yield, the histogram of CNFET sizes in our experimental design is shown in Figure 4.16-a. In this experiment only 20% of CNFETs are 1× size, while 47% of CNFETs are 4× size. This means that only 20% of CNFETs meet the required optimum P_{Target}, and the rest of the CNFETs exhibit larger probability of failure as shown in Figure 4.16-b.

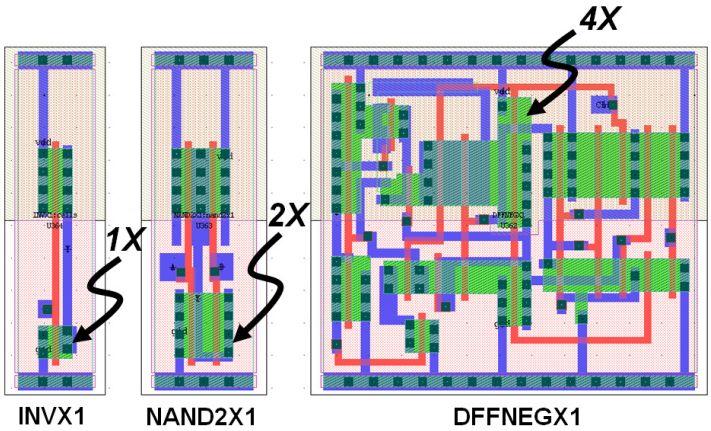


Figure 4.15. The layout of three different cells with minimum drive strength (1x) in Nangate standard cell library [91].

The issue of CNFET size variation in the cell library can be addressed by setting P_{Target} to a smaller value than 7.9×10^{-5} given by (4-22). It can be shown that, to account for the CNFET size variation, the modified target probability of failure, P'_{Target}, can be determined by:

$$P'_{Target} = \frac{P_{Target}}{S_{avg}} \tag{4-23}$$

where S_{avg} is the average normalized size of CNFET in the design. Based on Figure 4.8-a, S_{avg}=2.9 in our experimental design.

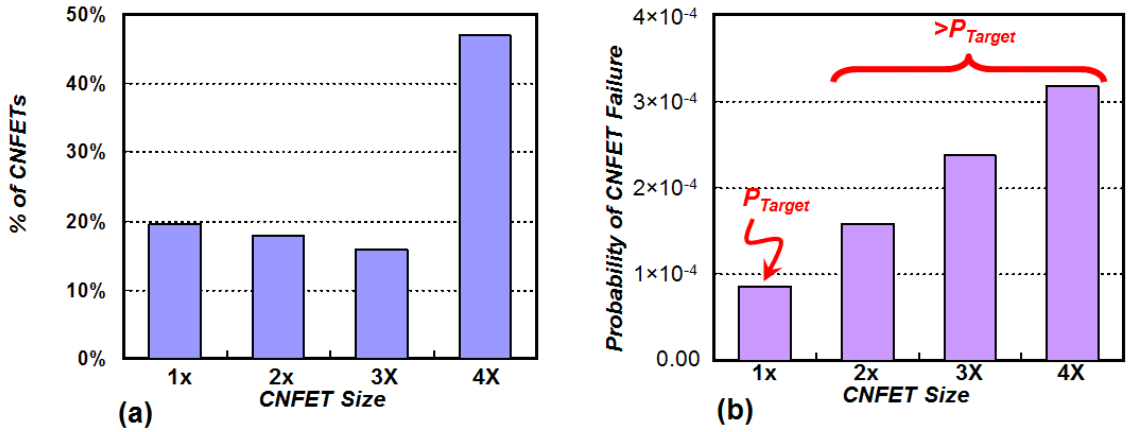


Figure 4.16. a-The histogram of CNFET sizes in the experimental design, and b-CNFET failure probability versus CNFET size.

Table 4.2 summarizes the results of the standard cell library optimization for CNFET technology used in our experimental design. In the ideal case, assuming no CNFET size variation, the total yield of 90% can be achieved with CNFET width of 0.86µm and P_M=9.2×10^{-4}. However, due to CNFET size variation in the cell library, the same design results in 74% total yield. Considering CNFET size variation in the standard cell library, the 90% total yield can be obtained if the minimum CNFET width is instead set to be 0.94µm and P_M=2.9×10^{-4}.

Table 4.2: Experimental results in a design with 133,172 CNFETs on chemically synthesized CNTs with density of *D*=10 CNT/µm using 45nm Nangate standard cell library

	P_{Target}	N_{CNT}	W_{CNFET}	P_M	*Total Yield*
Without CNFET size varaition	7.9×10^{-5}	8.6	0.86µm	9.2×10^{-6}	90%
With CNFET size varaition	7.9×10^{-5}	8.6	0.86µm	9.2×10^{-6}	74%
Modified With CNFET size varaition	2.7×10^{-5}	9.4	0.94µm	2.9×10^{-4}	90%

Chapter 5

5 Conclusion and Future Work

5.1 Conclusion

Carbon nanotube field effect transistor devices fabricated with ideal CNT synthesis can potentially offer more than an order of magnitude benefit in energy-delay product over Silicon CMOS. Although CNFETs are one of the promising candidates as extensions to CMOS devices, significant challenges in CNT synthesis prevent CNFETs today from achieving such ideal benefits. CNT density variation and m-CNTs are the dominant type of CNT variations/imperfections that cause performance variation, large static power consumption, and yield reduction.

This dissertation presented an imperfection-aware design of CNFET digital VLSI circuits. The major contribution of this dissertation were the following:

1. Analytical models are developed to analyze and quantify the effects of CNT density variation on device characteristics, gate and system levels delays. They are also validated by comparison to real experimental/simulation data. The analytical models enable us to examine the space of CNFET combinational, sequential and memory circuits to minimize delay variations. Using these model, we drive CNFET processing and circuit design guidelines to manage/overcome CNT density variations.

2. Analytical models are also developed to analyze the effects of m-CNTs on device characteristics, gate and system levels delay and power consumption. The models are also validated by comparison to simulation data. Using these model, we drive CNFET processing guidelines.
3. CNT density variation and m-CNTs can result in functional failure of CNFET circuits. The complete and compact model for CNFET probability of failure that consider CNT density variation and m-CNTs is also presented. This analytical model is applied to analyze the logical functional failures. The presented model is extended to predict opportunities and limitations of CNFET technology at today's Gigascale integration and beyond, assuming that the CNT placement during the growth process stays in a random and nonuniform fashion.

5.2 Future works

This section will briefly outline the possible extensions of this dissertation that can guide future research avenues.

In this dissertation, we considered a wide range of examples from device level, gate level and system level when proposing our CNT density variation model. The examples considered in this dissertation demonstrate different aspects of VLSI systems. Our ultimate goal is to have a unified CNT density variation aware framework for CNFET technology that incorporated into standard .Liberty format file, to be used by indusial logic synthesis and Static Timing Analysis tools.

In this dissertation, we mainly considered the effects of m-CNTs on delay and power consumption of the design. Even a very small amount of m-CNTs can drastically increase the static power dissipation of CNFET logic circuits and ruin the advantages of this

revelatory material. The ideal way to eliminate leakage current is to disconnect the module from the power supply rails, when the CNFET parts of a design in standby or sleep mode. This hybrid technique that use both CMOS and CNFET can be used for application like real time processing that for short period of time, fast processing is demanding.

In this dissertation, we considered CNT density variation and m-CNTs effects on gate delay variation separately. To the best of our knowledge, there is no work that analytically address the modeling of the both CNT density variation and m-CNTs in device, gate and system level.

6 References

[1] Horowitz, M.; Alon, E.; Patil, D.; Naffziger, S.; Rajesh Kumar; Bernstein, K., "Scaling, power, and the future of CMOS," Electron Devices Meeting, 2005. IEDM Technical Digest. IEEE International , vol., no., pp.7 pp.,15, 5-5 Dec. 2005

[2] Kuhn, K.J.; Liu, M.Y.; Kennel, H., "Technology options for 22nm and beyond," Junction Technology (IWJT), 2010 International Workshop on , vol., no., pp.1,6, 10-11 May 2010

[3] Jie Deng; Wong, H. -S P, "A Compact SPICE Model for Carbon-Nanotube Field-Effect Transistors Including Nonidealities and Its Application—Part I: Model of the Intrinsic Channel Region," Electron Devices, IEEE Transactions on , vol.54, no.12, pp.3186,3194, Dec. 2007

[4] R. Saito, G. Dresselhaus, M. S. Dresselhaus, *et al.*, Physical properties of carbon nanotubes, vol. 35. World Scientific, 1998.

[5] Design Methodologies and CAD for Emerging Nanotechnologies, Ph.D. thesis, Dept. Electric. Eng., EPFL Univ., Suisse, 2013.

[6] Patil, N.; Lin, A.; Jie Zhang; Wong, H. -S P; Mitra, S, "Digital VLSI logic technology using Carbon Nanotube FETs: Frequently Asked Questions," Design Automation Conference, 2009. DAC '09. 46th ACM/IEEE , vol., no., pp.304,309, 26-31 July 2009

[7] Lan Wei; Frank, D.J.; Chang, L.; Wong, H. -S P, "A non-iterative compact model for carbon nanotube FETs incorporating source exhaustion effects," Electron Devices Meeting (IEDM), 2009 IEEE International , vol., no., pp.1,4, 7-9 Dec. 2009

[8] Franklin, M. Luisier, S. Han, G. Tulevski, C. Breslin, L. Gignac, M. Lundstrom, and W. Haensch, “Sub-10 nm carbon nanotube transistor,” Nano letters, vol. 12, no. 2, pp. 758–762, 2012.

[9] J. Zhang, “Variation-aware design of carbon nanotube digital VLSI circuits,” Ph.D. thesis, Dept. Electric. Eng., Stanford Univ., Stanford,CA, 2011.

[10] A. Raychowdhury, A. Keshavarzi, J. Kurtin, V. De, and K. Roy, “Carbon nanotube field-effect transistors for high-performance digital circuits—dc analysis and modeling toward optimum transistor structure,” Electron Devices, IEEE Transactions on, vol. 53, no. 11, pp. 2711–2717, 2006.

[11] Y.-M. Lin, J. Appenzeller, J. Knoch, and P. Avouris, “High-performance carbon nanotube field-effect transistor with tunable polarities,” Nanotechnology, IEEE Transactions on, vol. 4, no. 5, pp. 481–489, 2005.

[12] M. S. Montemerlo, J. C. Love, G. J. Opitech, D. G. Gordon, and J. C. Ellenbogen, “Technologies and Designs for Electronic Nanocomputers,” MITRE Technical Report No. MTR 96W0000044, The MITRE Corporation, McLean, VA , July 1996.

[13] Tans, S. J., Verschueren, A. R.&Dekker, C.Room-temperature transistor based on a single carbon nanotube. Nature 393, 49–52,1998.

[14] Martel, R. A. , Schmidt, T., Shea, H. R., Hertel, T. & Avouris, P. Single-and multi-wall carbon nanotube field-effect transistors. Appl. Phys. Lett. 73, 2447,1998.

[15] M. Shulaker, G. Hills, N. Patil, H. Wei, H. Chen, P. Wong and S. Mitra,, "Carbon nanotube computer," Nature , Vol. 501, pp. 526-530, 2013.

[16] B. C. Paul, S. Fujita, M. Okajima, T. H. Lee, H.-S. Wong, and Y. Nishi, "Impact of a process variation on nanowire and nanotube device performance," Electron Devices, IEEE Transactions on, vol. 54, no. 9, pp. 2369–2376, 2007.

[17] J. Deng, N. Patil, K. Ryu, A. Badmaev, C. Zhou, S. Mitra, and H.-S.Wong, "Carbon nanotube transistor circuits: Circuit-level performance benchmarking and design options for living with imperfections," in Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International, pp. 70–588, IEEE, 2007.

[18] Hong, S. W., *et al.*, "Improved Density in Aligned Arrays of Single-Walled Carbon Nanotubes by Sequential Chemical Vapor Deposition on Quartz," Advanced Materials, Vol. 22, No. 45, pp. 1826-1830, 2010.

[19] Shulaker, M., *et al.*, "Linear Increases in Carbon Nanotube Density Through Multiple Transfer Technique," Nano Letters, Vol. 11, No. 5, pp. 1881-1886, 2011.

[20] Kang, S. J., *et al.*, "High-Performance Electronics Using Dense, Perfectly Aligned Arrays of Single-Walled Carbon Nanotubes," Nature Nanotechnology, Vol.2, No. 4, pp. 230-236, 2007.

[21] N. Patil, J. Deng, A. Lin, H. Wong, and S. Mitra, "Design methods for misaligned and mispositioned carbon-nanotube immune circuits," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, vol. 27, no. 10, pp. 1725–1736, 2008.

[22] Shahidipour, H.; Ahmadi, A.; Maharatna, K., "Effect of variability in SWCNT-based logic gates," Integrated Circuits, ISIC '09. Proceedings of the 2009 12th International Symposium on , vol., no., pp.252,255, 14-16 Dec. 2009

[23] S. Heinze, J. Tersoff, R. Martel, V. Derycke, J. Appenzeller, and P. Avouris, "Carbon Nanotubes as Schottky Barrier Transistors," Phys. Rev. Lett., vol. 89, no. 10, pp. 1068011-1068014, 2002.

[24] C. Zhou, J. Kong, and H. Dai, "Electrical measurements of individual semiconducting single-walled carbon nanotubes of various diameters," Appl. Phys. Lett., vol. 76, no. 12, p. 1597, 2000.

[25] J. Appenzeller, J. Knoch, M. Radosavljević, and P. Avouris, "Multimode Transportin Schottky-Barrier Carbon-Nanotube Field-Effect Transistors," Phys. Rev. Lett., vol. 92, no. 22, 2004.

[26] W. Kim, A. Javey, R. Tu, J. Cao, Q. Wang, and H. Dai, "Electrical contacts to carbon nanotubes down to 1 nm in diameter," Applied Physics Letters, vol. 87, no. 17, p. 173101, 2005.

[27] R. Ashraf, "Robust Circuit & Architecture Design in the Nanoscale Regime", Ph.D. thesis, Dept. Electric. Eng., Portland State Univ., 2011.

[28] Patil, N.; Lin, A.; Myers, E.R.; Ryu, Koungmin; Badmaev, A.; Chongwu Zhou; Wong, H. -S P; Mitra, S, "Wafer-Scale Growth and Transfer of Aligned Single-Walled Carbon Nanotubes," Nanotechnology, IEEE Transactions on , vol.8, no.4, pp.498,504, July 2009

[29] L. Qu, D. Feng, and L. Dai, "Preferential Syntheses of Semiconducting Vertically Aligned Single-Walled Carbon Nanotubes for Direct Use in FETs," Nano Lett., vol. 8, no. 9, pp. 2682–2687, 2008.

[30] T. Mizutani, H. Ohnaka, Y. Okigawa, S. Kishimoto, and Y. Ohno, "A study of preferential growth of carbon nanotubes with semiconducting behavior grown by plasma-enhanced chemical vapor deposition," Journal of Applied Phys., vol. 106, no. 7, pp. 073705–073705-5, Oct. 2009.

[31] M. Roberts, S. Barman, Y. Jin, J. Kim, and Z. Bao,"Self-sorted, aligned nanotube networks for thinfilm transistors," Science, vol. 321, no. 5885, pp. 101–104, 2008.

[32] S.H. Jin *et al.*, "Using nanoscale thermocapillary flows to create arrays of purely semiconducting single-walled carbon nanotubes," Nature Nanotech., vol. 8, pp. 347–355, Apr. 2013.

[33] R. Krupke, F. Hennrich, H. V. Löhneysen, and M. M. Kappes, "Separation of metallic from semiconducting single-walled carbon nanotubes,"Science, vol. 301, no. 5631, pp. 344–347, Jul. 2003.

[34] G. Zhang, P. Qi, X. Wang, Y. Lu, X. Li, R. Tu, S. Bangsaruntip, D. Mann, L. Zhang, and H. Dai, "Selective etching of metallic carbon nanotubes by gas-phase reaction," Science, vol. 314, no. 5801, pp. 974–977, Nov. 2006.

[35] A. Lin, N. Patil, H. Wei, S. Mitra, and H.-S. P. Wong, "A metallic-CNT tolerate carbon nanotube technology using asymmetric-correlated CNTs (ACCNT)," in VLSI Symp. Tech. Dig., Jun. 2009, pp. 182–183.

[36] A. Lin, N. Patil, H. Wei, S. Mitra, and H.-S. P. Wong, "ACCNT—A metallic-CNT-tolerant design methodology for carbon-nanotube VLSI: Concepts and experimental demonstration," IEEE Trans. Electron Devices, vol. 56, no. 12, pp. 2969–2978, Dec. 2009.

[37] A. Lin, J. Zhang, N. Patil, H. Wei, S. Mitra, and H.-S. P. Wong, "ACCNT: A metallic-CNT-tolerant design methodology for carbon nanotube VLSI: Analyses and design guidelines," IEEE Trans. Electron Devices, vol. 57, no. 9, pp. 2284–2295, Sep. 2010.

[38] J. Zhang, N. Patil, A. Hazeghi and S. Mitra, "Carbon Nanotube Circuits in the Presence of Carbon Nanotube Density Variations," Proc. Design Automation Conference, pp. 71-76, 2009.

[39] J. Zhang, N. Patil and S. Mitra, "Probabilistic Analysis and Design of Metallic-Carbon-Nanotube-Tolerant Digital Logic Circuits," IEEE Trans. CAD, 2009.

[40] J. Zhang, N. Patil, A. Lin, H.-S. P. Wong and S. Mitra, “Carbon Nanotube Circuits: Living with Imperfections and Variations,” Proc. Design, Automation and Test in Europe, pp. 1159-1164, 2010 (Invited).

[41] J. Zhang, N. Patil, H.-S.P. Wong and S. Mitra, “Overcoming Carbon Nanotube Variations through Co-optimized Technology and Circuit Design,” IEEE International Electron Devices Meeting, Dec. 2011.

[42] J. Zhang, N. Patil, A. Hazeghi, H.-S.P. Wong and S. Mitra, “Characterization and Design of Logic Circuits in the Presence of Carbon Nanotube Density Variations,” IEEE Trans. CAD, 2011

[43] H.-S.P. Wong, S. Mitra, D. Akinwande, C. Beasley, Y. Chai, H. Chen, X. Chen, G. Close, J. Deng, A. Hazeghi, J. Liang, A. Lin, L. Liyanage, J. Luo, J. Parker, N. Patil, M. Shulaker, H. Wei, L. Wei, J. Zhang, “Carbon Nanotube Electronics – Materials, Devices, Circuits, Design, Modeling, and Performance Projection,” IEEE Intl. Electron Devices Meeting, Washington D.C., Dec. 2011 (Invited)

[44] J. Zhang, L. Wei, N. Patil, A. Lin, H. Wei, H.-S.P. Wong and S. Mitra, “Robust Digital VLSI using Carbon Nanotubes,” Keynote paper, IEEE Trans. CAD, 2012.

[45] Patil, N.; Jie Deng; Mitra, S; Wong, H. -S P, "Circuit-Level Performance Benchmarking and Scalability Analysis of Carbon Nanotube Transistor Circuits," Nanotechnology, IEEE Transactions on , vol.8, no.1, pp.37,45, Jan. 2009

[46] Borkar, S., *et al.*, “Statistical Circuit Design with Carbon Nanotubes,” United States Patent Application 2007015506.

[47] Raychowdhury, A.; De, V.K.; Kurtin, Juanita; Borkar, S.Y.; Roy, K.; Keshavarzi, A., "Variation Tolerance in a Multichannel Carbon-Nanotube Transistor for High-Speed Digital Circuits," Electron Devices, IEEE Transactions on , vol.56, no.3, pp.383,392, March 2009

[48] Ashraf, R.; Chrzanowska-Jeske, M.; Narendra, S.G.; , "Functional Yield Estimation of Carbon Nanotube-Based Logic Gates in the Presence of Defects," IEEE Trans. Nanotechnology, vol.9, no.6, pp.687-700, Nov. 2010

[49] E. F. Moore and C. E. Shannon, “Reliable circuits using less reliable relays—Part I,” J. Franklin Inst., vol. 262, no. 3, pp. 191–208,Sep. 1956.

[50] E. F. Moore and C. E. Shannon, “Reliable circuits using less reliable relays—Part II,” J. Franklin Inst., vol. 262, no. 4, pp. 281–287, Oct. 1956.

[51] A. El-Maleh, B. Al-Hashimi, and A. Melouki, “Transistor-level based defect tolerance for reliable nanoelectronics,” in Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl., Mar. 2008, pp. 53–60.

[52] Zarkesh-Ha, P.; Shahi, A.A.M., "Stochastic Analysis and Design Guidelines for CNFETs in Gigascale Integrated Systems," Electron Devices, IEEE Transactions on , vol.58, no.2, pp.530,539, Feb. 2011

[53] J. Zhang, S. Bobba, N. Patil, A. Lin, H.-S. P. Wong, G. DeMicheli, and S. Mitra, “Carbon nanotube correlation: Promising opportunity for CNFET circuit yield enhancement,” in Proc. ACM IEEE Des. Autom.Conf., Jul. 2010, pp. 889–892.

[54] Shahi, A.A.M.; Zarkesh-Ha, P.; Elahi, M., "Comparison of variations in MOSFET versus CNFET in gigascale integrated systems," Quality Electronic Design (ISQED), 2012 13th International Symposium on , vol., no., pp.378,383, 19-21 March 2012

[55] Schroder,D. K.,Semiconductor Material and Device Characterization. NJ: Wiley, 2006, pp. 223–224.

[56] Nitta, T. Shibuya, and K. Homma,“Statistical static timing analysis technology,” FUJITSU Sci. Tech J., vol. 43, pp. 516–523, Oct 2007.

[57] Shahi, A.A.M.; Zarkesh-Ha, P., "Prediction of gate delay variation for CNFET under CNT density variation," Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2012 IEEE International Symposium on , vol., no., pp.140,145, 3-5 Oct. 2012

[58] Weste, N., and D. Harris, CMOS VLSI Design, 3rd edition, Addison Wesley, 2004.

[59] Jan M. Rabaey, Anantha Chandrakasan, Borivoje Nikolic,Digital Integrated Circuits,

[60] Sheldon M. Ross, Introduction to Probability Models, Tenth edition, Academic Press,2010

[61] Bowman, K.A.; Duvall, S.G.; Meindl, J.D."Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration,"Solid-State Circuits, IEEE Journal of , vol.37, no.2, pp.183-190, Feb 2002

[62] Shahidipour, H.; Ahmadi, A.; Maharatna, K."Effect of variability in SWCNT-based logic gates," Integrated Circuits, ISIC '09. Proceedings of the 2009 12th International Symposium on , vol., no., pp.252-255, 14-16 Dec. 2009

[63] http://www.itrs.net/Links/2010ITRS/Home2010.htm.

[64] Tegze P. Haraszti, CMOS MEMORY CIRCUITS, Spinger, 2000.

[65] Shahidipour, H.; Yue Zhong; Ahmadi, A.; Maharatna, K., "Effects of CNT diameter variability on a CNFET-based SRAM," Circuits and Systems (APCCAS), 2010 IEEE Asia Pacific Conference on , vol., no., pp.971,974, 6-9 Dec. 2010

[66] A. Sil, S. Ghosh, N. Gogineni, M. Bayoumi, “A novel high write speed, low power, read-SNM-free 6T SRAM cell,” Circuits and Systems, MWSCAS 2008. 2008.

[67] Y. Kim, Y. Kim, F. Lombardi, Y. Lee “A Low Power 8T SRAM Cell Design technique for CNFET”, IEEE International SOC Design Conference, 2008.

[68] Aghababa, H.; Zangeneh, M.; Afzali-Kusha, A.; Forouzandeh, B., "Statistical delay modeling of read operation of SRAMs due to channel length variation," Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on , vol., no., pp.2502,2505, May 30 2010-June 2 2010

[69] A. Bhavanagarwala, X. Tang, and J. Meindl, “The impact of intrinsic device fluctuations on CMOS SRAM cell stability,” IEEE J. Solid-State Circuits, vol. 36, no. 4, pp. 658–665, Apr. 2001.

[70] J. Lohstroh, E. Seevinck, and J. D. Groot, “Worst-case static noise margin criteria for logic circuits and their mathematical equivalence,” IEEE J.Solid-State Circuits, vol. 18, no. 6, pp. 803–806, Dec. 1983.

[71] J. Bhasker,Rakesh Chadha,"Static Timing Analysis for Nanometer Designs: A Practical Approach",Springer,2009

[72] Blaauw, D.; Chopra, K.; Srivastava, A.; Scheffer, L., "Statistical Timing Analysis: From Basic Principles to State of the Art," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , vol.27, no.4, pp.589,607, April 2008

[73] T.A DeMassa, Z. Ciccone, Digital Integrated Circuits, John Wiley & Sons,New York,1996

[74] Sakurai, T.; Newton, A.R., "Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas," Solid-State Circuits, IEEE Journal of , vol.25, no.2, pp.584,594, Apr 1990

[75] T. Yamada, “Modeling of electronic transport in scanning tunneling microscope tip-carbon nanotube systems,” Appl. Phys. Lett., vol. 78, no. 12, March. 2001.

[76] T. Yamada, “Modeling of kink-shaped carbon-nanotube Schottky diode with gate bias modulation,” Appl. Phys. Lett., vol. 8, no. 21, May 2002.

[77] T. Yamada, “Analysis of submicron carbon nanotube field-effect transistors,” Appl. Phys. Lett., vol. 76, no. 5, Jan. 2000.

[78] W. Rao, A. Orailoglu, R. Karri, “Fault Tolerance Arithmetic with Applications in Nanotechnology based Systems,” International Test Conference, pp. 472-478, OCT. 2004.

[79] European Commission, Technology Roadmap for Nanoelectronics, 2001.

[80] K. Nikolic, A. Sadek, and M. Forshaw, “Architectures for Reliable Computing with Unreliable Nanodevices,” IEEE-NANO, pp. 254-259, 2001.

[81] P. Beckett and A. Jennings, “Toward Nanocomputer Architecture,” Asia-Pacific Computer System Architecture Conference, pp. 141-150, October 2002.

[82] S. Mitra, “Globally Optimized Robust Systems to Overcome Scaled CMOS Reliability Challenges”, Design Automation and Test in Europe (DATE), pp. 941-946, March 2008.

[83] Zarkesh-Ha, P.; Shahi, A.A.M., "Logic Gate Failure Characterization for Nanoelectronic EDA Tools," Defect and Fault Tolerance in VLSI Systems (DFT), 2010 IEEE 25th International Symposium on , vol., no., pp.16,23, 6-8 Oct. 2010

[84] J. Von Neumann, “Probabilistic logics and the synthesis of reliable organisms from unreliable components,” Automata Studies, pp. 43-98, 1956.

[85] K. El Shabrawy, K. Maharatna, D. Bagnall, and B. Al-Hashimi, “Modeling SWCNT bandgap and effective mass variation using a Monte Carlo approach,” IEEE Trans. Nanotechnol., vol. 9, no. 2, pp. 184–193, Mar. 2010.

[86] N. Patil, A. Lin, J. Zhang, N. Patil, H.-S. P. Wong, and S. Mitra, “Digital VLSI logic technology using carbon nanotube FETs: Frequently asked questions,” in Proc. ACM IEEE Des. Autom. Conf., Jul. 2009, pp. 304–309.

[87] P. G. Collins, M. S. Arnold, and P. Avouris, “Engineering carbon nanotubes and nanotube circuits using electrical breakdown,” Science, vol. 292, no. 5517, pp. 706–709, Apr. 2001.

[88] Appenzeller, J., "Carbon Nanotubes for High-Performance Electronics—Progress and Prospect," Proceedings of the IEEE , vol.96, no.2, pp.201,211, Feb. 2008

[89] A. Keshavarzi, A. Raychowdhury, J. Kurtin, K. Roy, and V. De,“Carbon nanotube field-effect transistors for high-performance digital circuits—Transient analysis, parasitics, and scalability,” IEEE Trans.Electron Devices, vol. 53, no. 11, pp. 2718–2726, Nov. 2006.

[90] Wolfram Mathematica, http://www.wolframalpha.com

[91] Nangate, http://www.nangate.com