



**A TRANSFORMADA DA INCERTEZA COMO MÉTODO PARA
AVALIAR A VARIABILIDADE DE CIRCUITOS EM TECNOLOGIAS
EMERGENTES**

REINALDO GUTIERREZ PIMENTA

**DISSERTAÇÃO DE MESTRADO EM ENGENHARIA DE SISTEMA
ELETRÔNICOS E AUTOMAÇÃO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**FACULDADE DE TECNOLOGIA
UNIVERSIDADE DE BRASÍLIA**

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

**THE UNSCENTED TRANSFORM AS A METHOD TO ASSESS
CIRCUIT VARIABILITY FOR EMERGENT TECHNOLOGIES**

**A TRANSFORMADA DA INCERTEZA COMO MÉTODO PARA
AVALIAR A VARIABILIDADE DE CIRCUITOS EM TECNOLOGIAS
EMERGENTES**

REINALDO GUTIERREZ PIMENTA

**ORIENTADOR: STEFAN MICHAEL BLAWID, DR. RER. NAT.
COORIENTADOR: JOSÉ EDIL GUIMARÃES DE MEDEIROS, DR.**

DISSERTAÇÃO DE MESTRADO EM
ENGENHARIA DE SISTEMAS ELETRÔNICOS E
AUTOMAÇÃO

PUBLICAÇÃO: PPGEA.TD-720/2019

BRASÍLIA/DF: JUNHO - 2019

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

A TRANSFORMADA DA INCERTEZA COMO UM MÉTODO PARA
AVALIAR A VARIABILIDADE DE CIRCUITOS EM TECNOLOGIAS
EMERGENTES

REINALDO GUTIERREZ PIMENTA

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA
ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO
PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:



STEFAN MICHAEL BLAWID, Dr., ENE/UNB
(ORIENTADOR)



LEONARDO RODRIGUES ARAÚJO XAVIER DE MENEZES, Dr., ENE/UNB
(EXAMINADOR INTERNO)



WELLINGTON ÁVELINO DO AMARAL, Dr., FGA/UNB
(EXAMINADOR INTERNO)



JOSÉ EDIL GUIMARÃES DE MEDEIROS, Dr., ENE/UNB
(CO-ORIENTADOR)

Brasília, 14 de junho de 2019.

FICHA CATALOGRÁFICA

REINALDO GUTIERREZ

A transformada da incerteza como método para avaliar a variabilidade de circuitos em tecnologias emergentes

[Distrito Federal] 2019.

xi, 100p., 210 x 297 mm (ENE/FT/UnB, Mestre, Engenharia de Sistema Eletrônicos e Automação, 2019).

Dissertação de mestrado – Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Simulação de circuitos

2. CNTFET

3. Método de Monte Carlo

4. Transformada da Incerteza

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

PIMENTA (2019). A transformada da incerteza como método para avaliar a variabilidade de circuitos em tecnologias emergentes , Dissertação de mestrado, Publicação PPGA.TD-720/2019, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 100p.

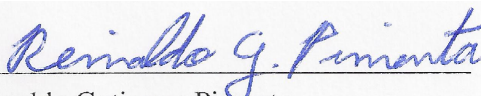
CESSÃO DE DIREITOS

AUTOR: Reinaldo Gutierrez Pimenta

TÍTULO: A transformada da incerteza como método para avaliar a variabilidade de circuitos em tecnologias emergentes .

GRAU: Mestre ANO: 2019

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.



Reinaldo Gutierrez Pimenta

Departamento de Engenharia Elétrica (ENE) - FT

Universidade de Brasília (UnB)

Campus Darcy Ribeiro

CEP 70919-970 - Brasília - DF - Brasil

Dedico este trabalho à minha família e a todos que fizeram possível a realização deste, em especial à minha mãe dona Julia Gutierrez Saiz e ao meu irmão Braulio Gutierrez Pimenta.

AGRADECIMENTOS

Agradeço ao professor José Camargo, por ter me dado a oportunidade de realizar este trabalho e ser o meu mentor durante todos estes anos na UnB, ao professor orientador Stefan Blawid por toda a paciência, dedicação e compreensão durante todo esse tempo, ao professor José Edil por me coorientar e dar a confiança de seu trabalho. Agradeço também a todos os meus amigos, responsáveis por fazer da minha vida uma grande jornada. Agradeço, por fim, ao Goku, por ter salvo a Terra mais uma vez.

“Se eu vi mais longe, foi por estar sobre ombros de gigantes.”

Sir Isaac Newton, 1675.

RESUMO

Título: A transformada da incerteza como método para avaliar a variabilidade de circuitos em tecnologias emergentes

Autor: Reinaldo Gutierrez Pimenta

Orientador: Stefan Michael Blawid, Dr. rer. nat.

Coorientador: José Edil Guimarães de Medeiros, Dr.

O presente trabalho propõe a transformada da incerteza (UT) como uma alternativa ao método de Monte Carlo (MC) para avaliar o funcionamento sob variabilidade de circuitos elétricos baseados em tecnologias emergentes. É mostrado que a aproximação discreta pela UT de funções de distribuição de probabilidade de variáveis aleatórias pode ser realizada a partir da quadratura Gaussiana. A tecnologia de nanotubos de carbono (CNT) possibilita a criação de circuitos de rádio frequência de comprimento moderado do canal e densidade de nanotubos. Como exemplo de aplicação, são comparadas as previsões de rendimento utilizando os métodos MC e UT de osciladores em anel construídos a partir de *buffers* de lógica em modo corrente. Apesar do escalonamento exponencial, este trabalho mostra evidências de sua aplicabilidade para a análise de circuitos menores de *benchmark*. Para circuitos maiores, a natureza determinística da UT permite a exploração de redundâncias inerentes ao circuito. Por fim, são apontados parâmetros da tecnologia com distribuição não-normais, que requerem novos algoritmos para computar a discretização utilizando a UT.

ABSTRACT

Title: The Unscented Transform as a method to assess circuit variability for emergent technologies

Author: Reinaldo Gutierrez Pimenta

Supervisor: Stefan Michael Blawid, Dr. rer. nat.

Co-Supervisor: José Edil Guimarães de Medeiros, Dr.

The unscented transform (UT) is proposed as an alternative to the Monte Carlo (MC) method for assessing performance variability of electronic circuits based on emergent technologies. We show that the discrete UT approximation of a continuous probability distribution of random variables can be solved by Gaussian quadrature. Carbon nanotube (CNT) technology enables radio frequency circuits at moderate channel length and semiconducting tube density. As an application example, we compare UT and MC yield predictions for CNTFET current-mode-logic ring oscillators. Despite of the exponential scaling of the computational effort with system size, the UT proves to be a powerful tool for the analysis of small benchmark circuits. For larger circuits the deterministic nature of the UT allows to explore redundancies. Finally, we point out that technology parameters are not necessarily normal distributed, which calls for new algorithms to compute the UT discretization.

CONTEÚDO

1	INTRODUÇÃO	1
1.1	RENDIMENTO DE FABRICAÇÃO E DESIGN VISANDO A FABRICAÇÃO	2
1.2	CNTFET	4
1.3	MÉTODO DE MONTE CARLO E A TRANSFORMADA DA INCERTEZA	6
1.4	OBJETIVOS	7
1.5	ORGANIZAÇÃO DA DISSERTAÇÃO	8
2	CNTFET	9
2.1	FABRICAÇÃO DO CNTFET	13
2.2	MODELAGEM DO CNTFET	17
3	MÉTODO DE MONTE CARLO E A TRANSFORMADA DA INCERTEZA	19
3.1	TRANSFORMADA DA INCERTEZA	20
3.2	CÁLCULO DOS PONTOS SIGMA E DE SEUS PESOS	22
4	ASPECTOS TÉCNICOS	25
4.1	MODELO COMPACTO DE TRANSISTOR DE NANOTUBO DE CARBONO - CCAM	26
4.2	CNTFET-WORKFLOW	30
5	RESULTADOS	38
5.1	IMPLEMENTAÇÃO DO CCAM NO NGSPICE	38
5.2	CIRCUITOS UTILIZADOS NA ANÁLISE	40
5.2.1	<i>Buffer</i> /INVERSOR DE LÓGICA EM MODO CORRENTE	40
5.2.2	OSCILADORES EM ANEL	41
5.3	CONVERGÊNCIA DO MÉTODO DE MONTE CARLO	42
5.4	EFICIÊNCIA COMPUTACIONAL DA UT	44
5.5	ESTUDO DE CASO: OSCILADORES EM ANEL	47
5.6	UTILIZAÇÃO DE PARÂMETROS COM DISTRIBUIÇÃO NÃO-NORMAL NA UT ..	51
6	CONCLUSÃO	54
6.1	PERSPECTIVAS FUTURAS	55
	REFERÊNCIAS BIBLIOGRÁFICAS	56

A APÊNDICE	63
A.1 NETLISTS CRIADAS PARA A DISSERTAÇÃO.....	65
A.1.1 CNTFET_CURVA_DE_TRANSFERENCIA.CIR.....	65
A.1.2 CNTFET_CURVA_DE_SAIDA.CIR	66
A.1.3 DC.CIR	68
A.1.4 FREQ_3_NOCAP.CIR	70
A.1.5 FREQ_5_NOCAP.CIR	75

Lista de Figuras

1.1	Domínio dos parâmetros (a) e domínio de performance (b). Adaptado de [9]. .	3
1.2	Modelo de CNTFET onde os nanotubos são cobertos por um óxido isolante (<i>top gating</i>), sob a porta (<i>gate</i>), onde o nanotubo conecta fonte (<i>source</i>) e dreno (<i>drain</i>) do transistor [15].	6
1.3	Aplicação da transformada da incerteza em uma função, o princípio da UT. Adaptado de [24].	7
2.1	Nanotubos de carbono em suas formas <i>armchair</i> (metálico) e <i>zig-zag</i> (semicondutor) [29].	9
2.2	O vetor quiral pode ser definido por um índice quiral (n, m) , utilizando os vetores unitários a_1 e a_2 em uma folha de grafeno [31].	10
2.3	Imagens de CNTFETs obtidas através de microscopia eletrônica de varredura.	11
2.4	Curva $I_d \times V_{ds}$ do CNTFET para diferentes valores de $V_{gs} - V_{th}$ [35].	13
2.5	Curva $I_d \times V_{gs}$ de um CNTFET para valores de $V_{ds} = 0,4$ V, com eletrodos de metais cujas funções trabalho são iguais a 4,2 eV, e com eletrodos de metais com três funções trabalho para a fonte e dreno: 3, 9 (linha contínua), 4, 2 (linha pontilhada) e 4,5 eV (linha tracejada) (adaptado de [36]).	14
2.6	Diagramas de bandas de energia de um CNTFET de barreira Schottky, com nanotubo de diâmetro $D_{cnt} = 1,26$ nm, comprimento $L_{cnt} = 20$ nm; funções trabalho do nanotubo e de todos os metais dos eletrodos iguais a 4,5 eV; $V_{ds} = 0,4$ V. Injeção de lacunas com $V_{gs} = 0,05$ V (linhas e setas pontilhadas); injeção de lacunas e elétrons com $V_{gs} = 0,2$ V (linhas e setas contínuas); injeção de elétrons com $V_{gs} = 0,35$ V (linhas e setas tracejadas) (adaptado de [36]).	15
2.7	(a) Diagrama do CNTFET ilustrando como o canal do nanotubo (corte mostrado à direita) é suspenso através da trincheira de silício e conectado em ambos os lados de fonte e dreno. (b) e (c) mostra imagens TEM de nanotubos na geometria <i>wrap-around</i> [41].	16
3.1	Transformação em três pontos sigma utilizando a UT de uma função distribuição normal com $\mu = 5$ e $\sigma^2 = 2$	21
4.1	Fluxo de trabalho da simulação.	25

4.2	Circuito equivalente do CCAM. A parte hachurada vermelha é responsável pela modelagem dos nanotubos semicondutores, enquanto que a parte azul modela os nanotubos metálicos (adaptado de [15]).	28
4.3	Medidas (símbolos) e modelagem (linhas contínuas) de curvas de transferência de um CNTFET de um único nanotubo com comprimento do canal $L_{ch} = 9$ nm. Gráfico da esquerda: escala logaritmica; gráfico da direita: escala linear. $V_{ds} = 0,01$ V (símbolo de adição), $V_{ds} = 0,4$ V (círculo) [15].	29
4.4	Medidas (símbolos) e modelagem (linhas contínuas) de curvas de saída de um CNTFET de um único nanotubo com comprimento do canal $L_{ch} = 9$ nm. $V_{gs} = 0, 1$ a $1,1$ V, com incrementos de $0,2$ V [15].	30
4.5	Curvas de transferência para vários valores de V_d	31
4.6	Configuração do CNTFET-Workflow.	32
4.7	Simulação utilizando do CNTFET-Workflow.	34
4.8	Tela principal do CNTFET-Workflow.	36
4.9	Tela de configuração do CNTFET-Workflow.	36
4.10	Tela de seleção dos parâmetros a serem variados.	37
5.1	Curva de transferência do CNTFET, mostrando a densidade de corrente para diferentes valores de densidade de nanotubos (n_{ta}).	39
5.2	Curva de saída do CNTFET, mostrando a densidade de corrente para diferentes valores de densidade de nanotubos (n_{ta}).	40
5.3	Esquemático do circuito <i>Buffer</i> /inversor de lógica em modo corrente.	41
5.4	Curva de transferência característica DC, $V_{out,m}$ (linha tracejada) e $V_{out,p}$ (linha contínua).	42
5.5	Representação de um oscilador em anel de “n” estágios. As saídas de um estágio são ligadas às entradas do estágio seguinte.	42
5.6	Sinal de saída $V_{out,diff}$ do oscilador de três estágios.	43
5.7	Média de cada conjunto de simulações de Monte Carlo	44
5.8	Gráficos das variáveis de entrada das simulações utilizando o método de Monte Carlo (a) e a UT (b). Os gráficos (c) e (d) mostram os passos das simulações de Monte Carlo e UT, respectivamente. O gráfico (e) mostra a função densidade de probabilidade das saídas das simulações de Monte Carlo (linha tracejada) e UT (linha contínua).	45
5.9	Simulações utilizando o método de Monte Carlo, com nove passos cada, com os mesmos parâmetros da simulação MC da Figura 5.8. A média e o desvio padrão de cada simulação estão escritos nos respectivos gráficos.	46

- 5.10 Gráficos das variáveis de entrada (n_{ta}) das simulações utilizando o método de Monte Carlo (a) e a UT (b). As frequências das simulações de Monte Carlo (linha contínua) e utilizando a UT (linha tracejada) do oscilador de três estágios são mostradas no gráfico (c), enquanto que as frequências da simulação do oscilador de cinco estágios são mostradas no gráfico (d). 48
- 5.11 Histogramas das simulações MC e UT dos osciladores de três e cinco estágios. 50
- 5.12 Variáveis de entrada, (a) para MC e (b) para UT e os resultados das simulações utilizando estas variáveis, (c) para a variável modelada com distribuição normal linha contínua, com $\mu_{p_{mt}} = 0,6\%$ e $\sigma_{p_{mt}} = 0,1\%$ (linha contínua), (d) para a variável com $\mu_{p_{mt}} = 0,6\%$ e desvio padrão $\sigma_{p_{mt}} = 0,4\%$ (linha tracejada) e (e) com $\mu_{p_{mt}} = 0,6\%$ e desvio padrão $\sigma_{p_{mt}} = 0,3\%$ (linha pontilhada). 52

Lista de Tabelas

4.1	Parâmetros físicos do CNTFET utilizados no cálculo dos parâmetros elétricos.	35
5.1	Valores listados de média e desvio padrão de cada simulação realizada para esta seção.	49
5.2	Parâmetros das distribuições normais das variáveis aleatórias de entrada e da figura de mérito $V_{out,diff}$	53
A.1	Parâmetros elétricos do CCAM	63
A.2	Parâmetros físicos base e parâmetros elétricos equivalentes	64

Lista de símbolos

Símbolos Latinos

L_{ch}	Comprimento do canal
R_c	Resistência de contato
I_d	Corrente de dreno
V_{ds}	Diferença de tensão entre dreno e fonte
V_{gs}	Diferença de tensão entre <i>gate</i> e fonte
V_{th}	Tensão de limiar

Símbolos gregos

μ	Média
σ^2	Variância

Siglas

Siglas

CMOS	Semicondutor de metal-óxido complementar (<i>Complementary Metal-Oxide-Semiconductor</i>)
CNTFET	Transistor de efeito de campo de nanotubos de carbono (<i>Carbon Nanotube Field-Effect Transistor</i>)
CNT	Nanotubos de Carbono (<i>Carbon Nanotube</i>)
CI	Circuito integrado
UT	Transformada da Incerteza (<i>Unscented Transform</i>)
SWCNT	Nanotubos de parede única (<i>single-wall carbon nanotubes</i>)
MWCNT	Nanotubos de paredes múltiplas (<i>multi-wall carbon nanotubes</i>)
MOSFET	Transistor de efeito de campo de óxido de metal semicondutor (<i>Metal-Oxide-Semiconductor Field-Effect Transistor</i>)
PDF	Função de densidade de probabilidade (<i>Probability Density Function</i>)
SRAM	Memória estática de acesso aleatório (<i>Static Random Access Memory</i>)
DFM	Projeto visando a fabricação (<i>Design For Manufacturability</i>)
DFY	Projeto visando o rendimento (<i>Design For Yield</i>)
CPU	Unidade central de processamento (<i>Central Processing Unit</i>)
CVD	Deposição Química por Vapor (<i>Chemical Vapor Deposition</i>)
CCAM	Modelo Compacto de Transistor de Efeito de Campo de Nanotubos de Carbono (<i>Compact Carbon Nanotube Field-Effect Transistor Model</i>)

1

INTRODUÇÃO

Para atingir uma taxa de desenvolvimento compatível com a demanda mundial, a indústria de semicondutores tem a necessidade de criar e utilizar um amplo conjunto de ferramentas para auxiliar no projeto e fabricação automáticos de circuitos, assim como a microeletrônica (e já presente nanoeletrônica) é uma parte crucial para o desenvolvimento tecnológico atual.

Um passo essencial para tal processo é a simulação de circuitos eletrônicos, para que assim os seus comportamentos físico e, conseqüentemente, elétrico sejam previstos, levando em conta uma margem de erro aceitável. O alto rendimento da fabricação destes circuitos, isto é, a proporção dos dispositivos fabricados que funcionam dentro das especificações, é essencial para a indústria de dispositivos semicondutores ser economicamente viável [1, 2]. O projeto e design de circuitos eletrônicos deve ser capaz de prever uma série de fatores e variações no ato de sua fabricação, como por exemplo a variabilidade das dimensões dos transistores. Assim, a previsão da variação de parâmetros físicos e elétricos durante a fase de simulação destes circuitos se faz necessária.

A simulação de circuitos, dependendo do seu tamanho e complexidade, necessita um esforço computacional cada vez maior, especialmente em se tratando de circuitos analógicos e de RF, que dependem mais de suas propriedades físicas e elétricas que circuitos digitais. O custo para projetar e fabricar circuitos integrados tende a aumentar através dos anos, sendo a complexa tecnologia necessária para fabricar dispositivos cada vez menores, até mesmo em escala nanométrica, além do aumento da variabilidade dos dispositivos, a razão para os altos custos. Para economizar tempo e dinheiro, é necessário que métodos de simulação sejam mais rápidos e mais confiáveis para que a qualidade de circuitos integrados seja maior a um custo menor.

A utilização da transformada da incerteza (UT) como alternativa ao método de Monte Carlo em simulações de circuitos elétricos baseados na tecnologia de transistores de efeito de campo de nanotubos de carbono (CNTFET) visa a redução do esforço computacional em determinados simulações de circuitos e, conseqüentemente, do tempo necessário para realizá-las. A escolha da tecnologia do CNTFET para este trabalho é justificada pelo potencial de ampliar e complementar a tecnologia CMOS em várias áreas de aplicação da microeletrônica. É com base neste contexto que a proposta desta dissertação mostra-se válida e necessária. O presente trabalho faz uma análise da variabilidade dos parâmetros físicos de circuitos de tecnologia CNTFET, apresentando algumas análises sobre o rendimento do processo utilizando a UT, baseados em Cheralathan, Claus e Blawid [3], que analisam os

extremos da variabilidade do processo de fabricação.

1.1 RENDIMENTO DE FABRICAÇÃO E DESIGN VISANDO A FABRICAÇÃO

Devido a limitações no controle de processos, as tolerâncias de fabricação na tecnologia não progredem no mesmo ritmo que a redução do comprimento do canal. Além disso, desvios causados pelos limites físicos fundamentais aumentam significativamente com a redução das dimensões da tecnologia. Devido a tais fatos, variações estatísticas de diversos parâmetros se tornam piores em sucessivas gerações de tecnologia, e a variabilidade é atualmente um dos mais desafios que a indústria de semicondutores enfrenta. Esta variabilidade tem afetado o projeto de circuitos analógicos por um bom tempo, e agora exerce um impacto dramático no projeto de vários tipos de circuitos integrados, tanto analógicos e RF como digitais, como por exemplo no de memória estática de acesso aleatório (SRAM, ou *Static Random Access Memory*) em tecnologias de nós em escala nanométrica [4].

É essencial fabricar um produto com rendimento suficiente para cobrir os custos de fabricação e ter um lucro mínimo. Existem duas maneiras de se aumentar tal rendimento: (i) aumentar o controle sobre o processo de fabricação e (ii) projetar circuitos de maneira a minimizar os efeitos de flutuação inerentes à fabricação no funcionamento do circuito [5]. O segundo item pode ser descrito também como “*Design* visando a fabricação”. O projeto visando a fabricação (DFM, ou *Design for manufacturability*) [6] é, em sua ampla definição, a metodologia que assegura que um circuito possa ser fabricado repetidamente, de forma consistente, de modo confiável e com um custo efetivo, tomando as medidas necessárias para esse fim durante os processos de projeto, fabricação, e montagem.

Apesar do conceito de DFM possuir certa maturidade, somente recentemente se separou do conceito de projeto visando o rendimento (DFY, ou *Design For Yield*) [7], estes dedicados principalmente para o cálculo de rendimento como uma função de densidades de defeitos. O conceito de DFY, por outro lado, é um conceito recente, baseado no fato de que na era dos dispositivos nanoeletrônicos não basta se ater somente a regras de projeto, ou seja, a regras de DFM, pois o rendimento resultante pode ser proibitivamente baixo. Com isso, um novo conjunto de procedimentos e regras de projeto é necessário para serem aplicados a um projeto além das regras fabricação para se assegurar um rendimento aceitável [8]. Atualmente, o uso dos conceitos de DFM e DFY tem-se considerados como conceitos iguais. No entanto, é mais correto dizer que o conceito de DFY é derivado do conceito de DFM.

O rendimento paramétrico, definido como a fração de circuitos fabricados que funcionam

dentro das especificações, tornou-se um critério importante para avaliar a robustez do projeto sob variações do processo de fabricação. No geral, o rendimento paramétrico pode ser estimado tanto no domínio de performance quanto no de parâmetro, como mostrado na Figura 1.1. O domínio da performance é definido como o espaço de todas as figuras de mérito (*i.e.* tensão, ganho, largura de banda, etc.), enquanto que o domínio dos parâmetros é o espaço constituído por todos os parâmetros do circuito (*i.e.* largura de canal, tensão de limiar, etc.), cujas fronteiras são seus valores mínimos e máximos. Em ambos os domínios, os conjuntos de parâmetros dentro das especificações formam a área de rendimento. Assim, a fração do conjunto de amostras aceitáveis em ambos os domínios pode ser usada para estimar o rendimento.

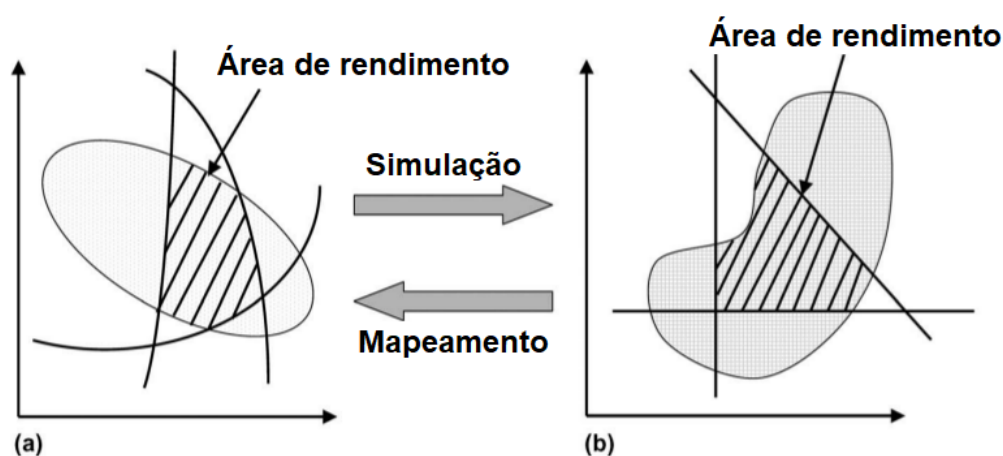


Figura 1.1: Domínio dos parâmetros (a) e domínio de performance (b). Adaptado de [9].

Um exemplo comumente associado a problemas de rendimento e como um projeto que não visa o rendimento em uma etapa crítica na fabricação de circuitos integrados é referente ao Pentium Pro. Lançado em novembro de 1995 com conexão no formato *socket 8*, este processador continha 5,5 milhões de transistores, além de uma memória cache L2 no mesmo encapsulamento da Unidade Central de Processamento (CPU, ou *Central Processing Unit*). Esta memória cache L2 era conectada à CPU nas fases iniciais do processo de produção, não existindo qualquer tipo de teste e validação de um dos dois componentes. Como consequência, qualquer problema presente em um deles tornava todo o processador inutilizável, visto que era impossível detectar qualquer falha antes da inserção de ambos no encapsulamento. Logo, o rendimento total do processador é tido com o produto do rendimento de ambos os componentes, fazendo com que o processo de fabricação utilizado possuísse um rendimento baixo, mantendo o preço do processador alto em relação a seus equivalentes de mercado [10]. Este é um exemplo de DFM a nível de sistema e pode ser relacionado de maneira mais específica ao projeto visando a testabilidade (ou *Design For Test*, DFT).

Raramente tantos fatores e variáveis de um projeto particular e de um processo de fabricação passaram por mudanças fundamentais simultâneas como os da era atual de escala nanométrica da indústria de semicondutores. O aumento do número de novas variáveis introduzidas em qualquer processo de fabricação aumenta a probabilidade de erro causado por um defeito em qualquer parâmetro físico, ainda mais com a necessidade de uma consideração meticulosa do impacto de cada variável no processo de fabricação como um todo. As tecnologias emergentes, em seus estágios iniciais de desenvolvimento, são fortemente afetadas por variáveis poucos controláveis em seus processos de fabricação.

1.2 CNTFET

Com o dimensionamento da tecnologia CMOS se aproximando de seus limites fundamentais, vários dispositivos de processamento de informação e microarquiteturas para as novas e existentes aplicações são explorados para estender os aspectos históricos do dimensionamento de circuitos integrados. Segundo International Roadmap for Devices and Systems [11], a indústria de semicondutores enfrenta alguns desafios relacionados para ampliar a tecnologia de circuitos integrados para novas aplicações e para além do fim do dimensionamento da tecnologia CMOS, podendo classificá-los em cinco aspectos. Como primeiro aspecto, executa-se o ato de se avançar nesse tipo de tecnologia em sua máxima densidade e funcionalidade, integrando-se novas tecnologias de alta velocidade, alta densidade e baixo consumo. Segundo, estender o dimensionamento da tecnologia CMOS com materiais alternativos para compor o canal do transistor. Terceiro, avançar no processamento de informação substancialmente além do atingível pela tecnologia CMOS usando uma combinação inovadora de novos dispositivos, interconexões e mudanças na arquitetura para estender a tecnologia CMOS e eventualmente criar novas plataformas de tecnologia em processamento de informação. Quarto, estender ao máximo a tecnologia CMOS em seu dimensionamento máximo como uma plataforma de tecnologia nos novos domínios de aplicação e funcionalidade. O quinto e último aspecto é composto de desafios é o fato de se ocupar o espaço entre dispositivos novos e arquiteturas não convencionais e paradigmas computacionais.

O segundo aspecto se torna uma área de trabalho essencial, possuindo o desafio mais difícil de se pesquisar materiais com propriedades controladas que permitam a utilização de componentes emergentes com alta densidade em escala nanométrica. Alguns componentes promissores que possuem tais propriedades são os transistores de efeito de campo de nanotubos de carbono (CNTFET), os transistores de efeito de campo de nanofio (FETs de nanofio), os canais de FETs de material 2D, os dispositivos de substituição do canal tipo P com elementos III-V, os dispositivos de substituição do canal tipo N de Germânio e os FETs de efeito

túnel. A tecnologia de CNTFET, escolhida para as análises realizadas no presente trabalho, é descrita a seguir.

Com o esgotamento da lei de Moore [12], novas tecnologias são necessárias para atender às demandas e necessidades da sociedade moderna. Novos materiais, arquiteturas de computador e paradigmas computacionais são desenvolvidos para suplementar a atualmente dominante tecnologia CMOS. Para muitos pesquisadores, a procura por um semiconductor ideal para ser usado nos transistores de efeito de campo foi realizada com sucesso quando nanotubos de carbono de parede única foram inicialmente apresentados como dispositivos promissores, mostrando-os como uma alternativa que se mostra consideravelmente viável em determinadas aplicações.

O CNTFET, por se tratar de uma tecnologia relativamente nova, possui diversos desafios a serem superados perante sua fabricação. São especialmente importantes o controle de quiralidade, que determina as propriedades semicondutoras do nanotubo, o alinhamento dos nanotubos através do transistor, dopagem, aprimoramento dos contatos, e os desafios mais importantes, o aumento da densidade de nanotubos (ultrapassar a densidade linear de 200 CNTs μm^{-1}), e a alta porcentagem de nanotubos semicondutores, essencial para um funcionamento satisfatório do transistor [13]. Por ser uma tecnologia com grandes desafios no que tange à sua fabricação, a utilização do CNTFET na indústria de semicondutores ainda reside em perspectivas futuras.

Principal componente do CNTFET, o nanotubo de carbono, compondo o transistor cujo modelo é mostrado na Figura 1.2, é um material excepcional de muitas maneiras diferentes. São extremamente fortes mecanicamente, a superfície é perfeitamente regular e normalmente isenta de imperfeições, o transporte dos elétrons é feito por movimento balístico [14]. Como consequência, o CNTFET possui algumas características altamente desejadas em transistores, tais como a alta capacidade de miniaturização, o grande controle sobre a formação do canal, a baixa tensão de limiar (V_{th}), a alta mobilidade de elétrons e a alta densidade de corrente.

O CNTFET possui vantagens que podem ser exploradas em várias aplicações, tanto em analógicas e baseadas em RF como em aplicações digitais. Pode-se citar sensoriamento com aplicações em biomedicina [16], *OP-AMPs low power* [17], aplicações em RF [18], portas inversoras para a síntese de circuitos digitais [19].

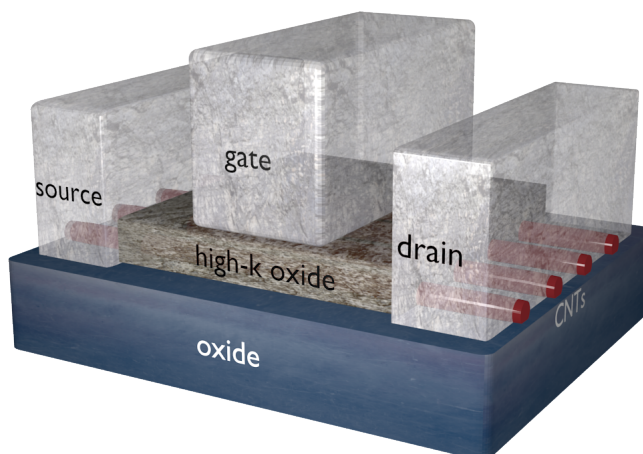


Figura 1.2: Modelo de CNTFET onde os nanotubos são cobertos por um óxido isolante (*top gating*), sob a porta (*gate*), onde o nanotubo conecta fonte (*source*) e dreno (*drain*) do transistor [15].

1.3 MÉTODO DE MONTE CARLO E A TRANSFORMADA DA INCERTEZA

Um dos métodos computacionais mais utilizados na análise de variação de processos de fabricação é o método de Monte Carlo [20]. Criada após a Segunda Guerra Mundial no projeto Manhattan, foi inicialmente uma ferramenta desenvolvida para estudar a distância que nêutrons estatisticamente passariam através de diferentes materiais. Inicialmente, tal método fora adotado por determinadas áreas de pesquisa de ciências puras, tais como física, físico-química e pesquisas em operações militares. O método consiste em sortear amostras aleatórias massivas e aplicá-las ao modelo em questão, mapeando-o. O conceito principal é utilizar a aleatoriedade intrínseca ao modelo para se resolver problemas a princípio determinísticos.

Embora muito popular, requer um esforço computacional considerável, dependendo de sua aplicação. Pesquisas são feitas para propor e criar alternativas a este método, embora seja amplamente utilizado nos dias de hoje em diversas áreas, como por exemplo ciências físicas [21], economia e finanças [22] e até mesmo na pesquisa de jogos eletrônicos [23].

A transformada da incerteza (em inglês *Unscented Transform*, ou simplesmente UT) é uma transformada criada por Simon Julier e Jeffrey Uhlmann [24], inicialmente para fins de rastreamento de objetos. Ela é baseada na premissa de que “é mais fácil aproximar uma função de distribuição de probabilidade do que uma função arbitrária não-linear ou transformação” [25]. Esta transformada pode ser aplicada, por exemplo, em funções de densidade de probabilidade (PDF), gerando um conjunto de quantidade arbitrária de pontos, nomeados de

pontos sigma, e seus respectivos pesos $\{s_i, w_i\}$. Estes pontos têm a característica de possuir a média e a variância (e outros momentos) com aproximação em relação à função original proporcional ao número de pontos sigma escolhidos. Por possuir estas características, a UT é amplamente utilizada nesta dissertação e analisada como alternativa ao método de Monte Carlo em determinadas aplicações. A Figura 1.3 mostra um esquema representando a aplicação da UT em uma função.

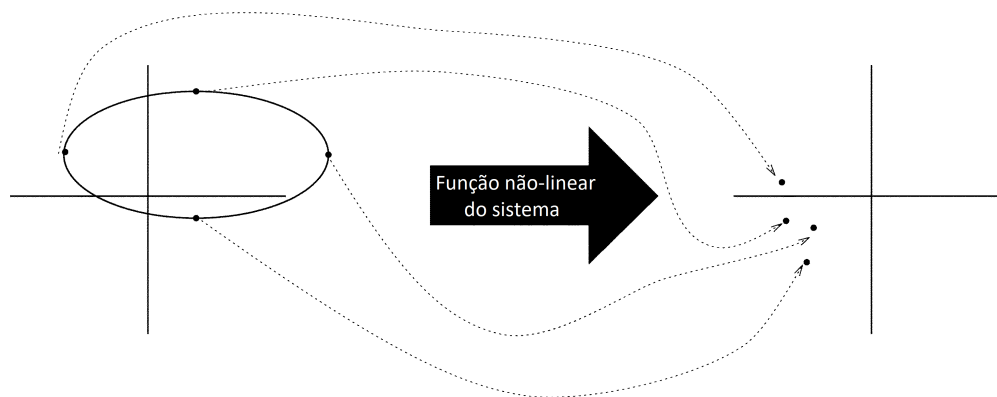


Figura 1.3: Aplicação da transformada da incerteza em uma função, o princípio da UT. Adaptado de [24].

1.4 OBJETIVOS

Esta dissertação tem como foco a união de duas frentes de pesquisa: a modelagem compacta do transistor de efeito de campo de nanotubos de carbono e o uso da transformada da incerteza como uma alternativa para simulações utilizando o método de Monte Carlo. Assim, estuda-se o impacto das incertezas de fabricação na variabilidade dos circuitos integrados baseados em CNTFETs utilizando um modelo compacto, desenvolvido por Schröter et al. [26] e a transformada da incerteza. Tendo um conhecimento maior sobre o comportamento eletrônico do CNTFET implica em uma maior capacidade de entendimento e de projeto de circuitos baseados nesta tecnologia, estendendo um pouco mais a lei de Moore.

De uma maneira concisa, pode-se dividir os objetivos desta dissertação nos seguintes itens:

1. Apresentar o CNTFET, seus desafios como tecnologia emergente, utilizada nas análises da dissertação;

2. Criar um programa de *workflow* no ambiente de programação GNU Octave [27] para a automatização do processo de simulação, utilizando o simulador de circuitos elétricos Ngspice [28] e o modelo de CNTFET CCAM [15];
3. Prever o rendimento paramétrico limitado por uma dada incerteza na densidade de nanotubos e na porcentagem de nanotubos metálicos, empregando ambos os métodos MT e UT.

1.5 ORGANIZAÇÃO DA DISSERTAÇÃO

O corpo principal da dissertação está organizado em seis capítulos, incluindo este capítulo introdutório. O segundo capítulo faz uma fundamentação teórica sobre uma das principais frentes de pesquisa associadas a este trabalho: tecnologias emergentes na fabricação de circuitos integrados, bem como a tecnologia CNTFET e o rendimento na fabricação de circuitos integrados. O terceiro capítulo descreve o método de Monte Carlo e a transformada da incerteza. O quarto capítulo apresenta o Ngspice e do CCAM, modelo semicompato de CNTFET, utilizado nas simulações, além do programa CNTFET-Workflow, desenvolvido para realizar as simulações presentes na dissertação. Este capítulo apresenta também a estratégia de execução das simulações, para suas posteriores análises. No capítulo 5 são apresentadas as simulações realizadas, assim como é feita uma análise das mesmas. Por fim, o sexto capítulo conclui o trabalho apresentado e traz um breve ensaio sobre a perspectiva de trabalhos futuros.

2

CNTFET

Nanotubos de carbono são uma forma alotrópica do carbono que podem ser consideradas o resultado do enrolamento de uma folha de grafeno. Possuem características mecânicas e elétricas únicas, podendo ser divididos em basicamente duas categorias: nanotubos de parede simples (*single-wall carbon nanotubes*, SWCNT) e nanotubos de paredes múltiplas (*multi-wall carbon nanotubes*, MWCNT). Dependendo do ângulo de enrolamento e do diâmetro do nanotubo gerado, ele pode possuir características de condução metálicas ou semicondutoras, ilustradas na Figura 2.1.

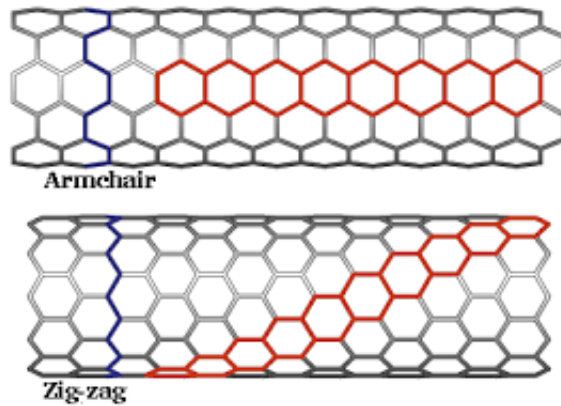


Figura 2.1: Nanotubos de carbono em suas formas *armchair* (metálico) e *zig-zag* (semicondutor) [29].

A Figura 2.2 ilustra os vetores do índice quiral, que caracteriza o comportamento elétrico do nanotubo [30]. Para possuir características de condução metálica, os índices n e m de enrolamento do nanotubo devem ser iguais ($n = m$). Se $n - m$ é múltiplo de 3 além de $n \neq m$, $n \neq 0$ e $m \neq 0$, o nanotubo terá características quasi-metálicas. Porém, se $n \neq m$, o nanotubo terá características semicondutoras. A condução elétrica no CNT é descrita detalhadamente em [14].

O diâmetro do nanotubo pode ser calculado a partir da Eq. (2.1)

$$D = \frac{a\sqrt{n^2 + m^2 + nm}}{\pi} \quad (2.1)$$

em que a é a constante da rede cristalina do grafeno, igual a $\sqrt{3} \cdot a_{cc} = 0,246$ nm e $a_{cc} = 0,142$ nm (distância entre átomos na ligação entre carbonos).

O ângulo quiral é definido pela Eq. (2.2)

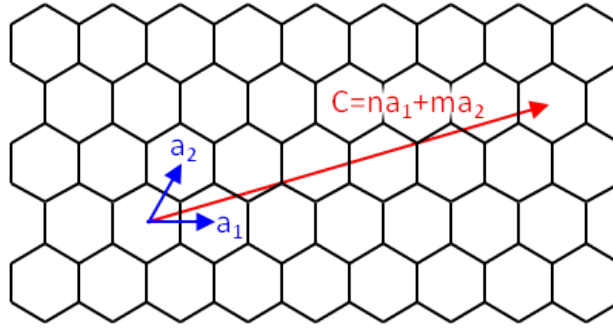


Figura 2.2: O vetor quiral pode ser definido por um índice quiral (n, m) , utilizando os vetores unitários a_1 e a_2 em uma folha de grafeno [31].

$$\theta = \tan^{-1} \left(\frac{\sqrt{3}n}{2m + n} \right) \quad (2.2)$$

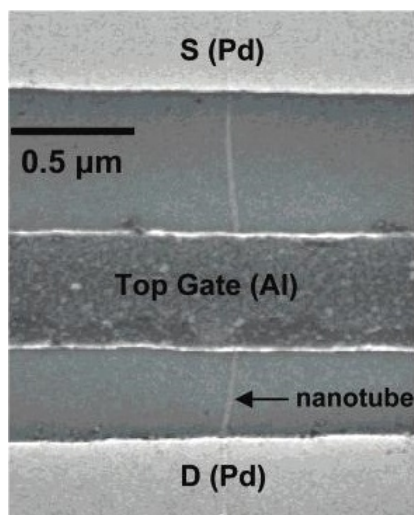
Este ângulo é definido como o ângulo entre o vetor quiral C , mostrado na Figura 2.2 e o vetor de sentido do enrolamento zigue zague, mesma direção do vetor a_1 da mesma figura citada anteriormente.

O *bandgap* do nanotubo de carbono está diretamente ligado ao seu diâmetro. Guardada a característica de condutividade do nanotubo de acordo com sua quiralidade, quanto maior o diâmetro do nanotubo, menor o seu *bandgap*. O trabalho de Matsuda, Tahir-Kheli e III [32] faz uma comparação dos *bandgaps* teóricos e calculados de nanotubos de carbono de parede simples.

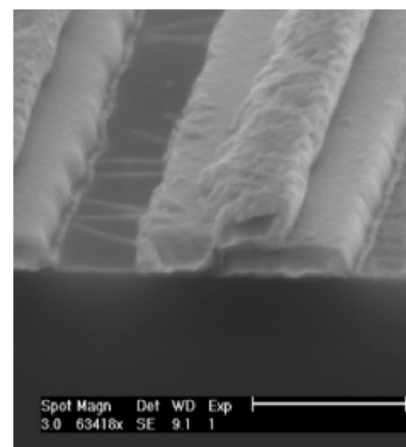
Tendo o CNT como principal componente do CNTFET, é natural que algumas de suas características alterem substancialmente o funcionamento do transistor. As principais características são a densidade de nanotubos e a porcentagem de nanotubos metálicos. A densidade de nanotubos está diretamente ligada à densidade de corrente do dispositivo, assim quanto maior a densidade de nanotubos maior a corrente que passa pelo dispositivo. Embora uma densidade de nanotubos de $10 \mu\text{m}^{-1}$ seja realista para experimentos atuais [18], esta dissertação trabalha com valores razoavelmente acima deste valor, para que assim permaneça aplicável mesmo com futuros avanços na tecnologia. A porcentagem de nanotubos metálicos altera drasticamente o funcionamento do transistor. Aumentando este parâmetro, aumenta-se o número de conexões de nanotubos metálicos entre dreno e fonte, diminuindo assim o controle do transistor. Tal fato é justificado por não apresentarem variação da condução com a variação da tensão no *gate* V_g . Portanto, é altamente desejável que a porcentagem de nanotubos metálicos seja sempre a menor possível.

O CNTFET é caracterizado principalmente por ter um ou mais nanotubos de carbono conectando fonte e dreno, como mostrado na Figura 2.3. Quando há um conjunto de tensões

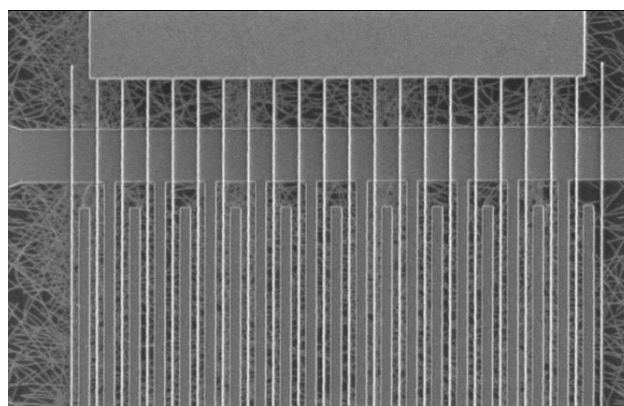
aplicadas à fonte, dreno e porta do transistor, uma corrente é conduzida entre fonte e dreno. A corrente pode ser gerada a partir de duas maneiras, a depender do comprimento do nanotubo. Em transistores com nanotubos maiores que 100 nm, o transporte de carga é feito pelos mecanismos de deriva e difusão. Para nanotubos com comprimento menor, o mecanismo de transporte balístico de elétrons e lacunas se torna predominante [14].



(a) Nanotubos de carbono de parede única vistos por microscopia eletrônica [33].



(b) Região do canal de um CNTFET com nanotubos visíveis [18].



(c) CNTFET para aplicações de alta frequência, composto por vários “dedos” e múltiplos nanotubos, visíveis na camada inferior [34].

Figura 2.3: Imagens de CNTFETs obtidas através de microscopia eletrônica de varredura.

Um arquitetura bastante utilizada em pesquisas consiste nas conexões entre fonte e dreno e nanotubos serem feitas por deposição dos nanotubos no substrato e posterior metalização de suas extremidades, sendo a distância entre fonte e dreno o valor de L_{ch} . As conexões entre metal e semiconductor, por terem diferentes valores da função trabalho, são caracterizadas por formarem uma barreira Schottky, resultando em uma resistência de contato. Na ausência

de contatos ôhmicos, o controle da corrente provém do controle eletrostático da espessura da barreira Schottky, fazendo com que a resistência de contato tenha um valor mínimo de aproximadamente $R_c = 6,6 \text{ k}\Omega$ por nanotubo [35], próximo do limite quântico previsto. Este valor pode variar dependendo do diâmetro e do número de nanotubos presentes no transistor.

Variando-se a tensão aplicada ao *gate* obtém-se a modulação da corrente do transistor. Enquanto que no MOSFET convencional a porta controla a barreira de potencial dentro do canal, no CNTFET o controle da corrente é feito pela modulação da espessura da barreira nos contatos. Mais detalhes e informações sobre a modelagem do CNTFET serão apresentadas na seção 4.1.

O transistor CNTFET é alvo de várias pesquisas pelo seu potencial de complementar e ampliar a tecnologia MOSFET (*Metal-Oxide-Semiconductor Field-Effect Transistor*, ou transistor de efeito de campo de óxido de metal semiconductor). Tal fato possui duas estratégias distintas:

1. Atingir um comprimento de canal L_{ch} menor que o menor comprimento possível na tecnologia CMOS [35];
2. Alcançar um melhor desempenho possuindo o mesmo L_{ch} que a tecnologia CMOS em comparação.

Vale salientar que as simulações realizadas na presente dissertação seguem a segunda estratégia supracitada.

Transistores MOSFET com comprimento menor começam a ter um comportamento alterado, diminuindo o controle da corrente que passa pelo transistor. O transistor CNTFET é um razoável candidato a substituir o transistor MOSFET pela sua capacidade de controle de corrente com L_{ch} abaixo de 10 nm.

A curva $I_d \times V_{ds}$ característica do CNTFET é mostrada na Figura 2.4, para diferentes valores de $V_{gs} - V_{th}$. Pode-se notar um comportamento semelhante ao MOSFET funcionando na região de triodo. As lacunas podem ser bastante importantes na composição da corrente dos CNTFET. Pela estrutura das bandas de condução e de valência em nanotubos de carbono serem simétricas, lacunas podem ser injetadas em nanotubos assim como elétrons, considerando que funções trabalho em ambos os contatos do nanotubo sejam favoráveis. Como um exemplo, a Figura 2.5 mostra uma situação onde um CNTFET de barreira de Schottky possui eletrodos de metais na fonte, dreno e *gate* cujas funções trabalho são iguais. A Figura 2.6 mostra o diagrama de bandas de energia do exemplo em questão.

Com um valor de V_{ds} fixo, os perfis das primeiras bandas de condução e valência para três valores de V_{gs} são mostradas. Para um valor baixo de V_{gs} há injeção de lacunas pelo dreno

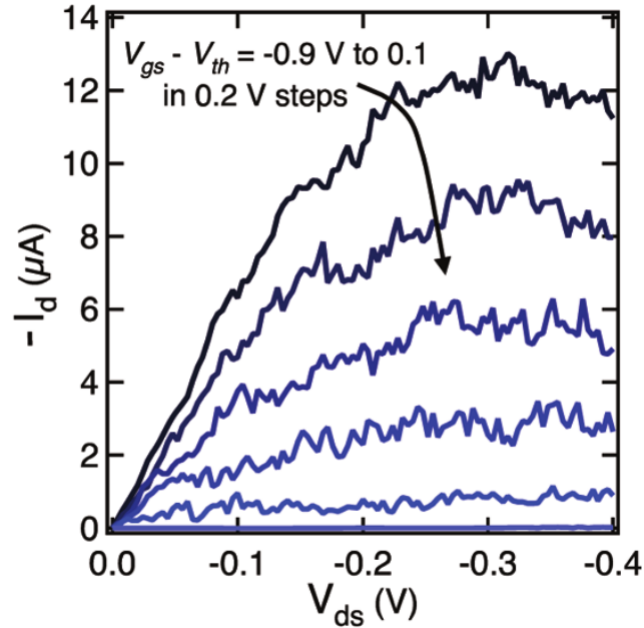


Figura 2.4: Curva $I_d \times V_{ds}$ do CNTFET para diferentes valores de $V_{gs} - V_{th}$ [35].

por tunelamento. Assim, o dispositivo opera como um transistor tipo P. Essa corrente diminui com a afinação da barreira no dreno e o aumento do valor de V_{gs} . Entretanto, a corrente total não continua a diminuir como num MOSFET de silício por causa da barreira na fonte se tornar mais fina, possibilitando o fluxo de elétrons através da fonte por tunelamento.

Por causa da presença de uma corrente de corte notável, o CNTFET deste exemplo não seria ideal para aplicações de lógica digital. Entretanto, a ambipolaridade do dispositivo, isto é, a habilidade de injetar elétrons e lacunas, em dispositivos suficientemente longos, pode fazer com que haja recombinação de elétrons e lacunas dentro do nanotubo. Como o *bandgap* nestes nanotubos é direto, pode haver então emissão de luz. A simetria do transporte de elétrons e lacunas é explorada no modelo utilizado nas simulações deste trabalho.

2.1 FABRICAÇÃO DO CNTFET

A fabricação de circuitos integrados baseados em CNTFET enfrenta atualmente uma série de desafios a serem superados para ser industrial e economicamente viável. A deposição dos nanotubos de carbono durante a fabricação do transistor, essencial para o correto funcionamento do mesmo, é uma delas. Um método comumente empregado no crescimento dos nanotubos é a deposição química por vapor (CVD, ou *Chemical Vapor Deposition*), normalmente por gás metano, metanol ou etileno na presença de um catalisador adequado [37]

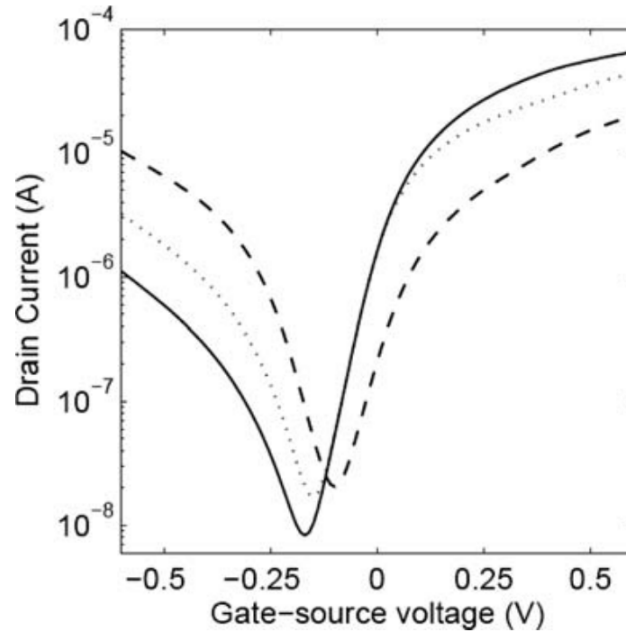


Figura 2.5: Curva $I_d \times V_{gs}$ de um CNTFET para valores de $V_{ds} = 0,4 \text{ V}$, com eletrodos de metais cujas funções trabalho são iguais a $4,2 \text{ eV}$, e com eletrodos de metais com três funções trabalho para a fonte e dreno: $3,9$ (linha contínua), $4,2$ (linha pontilhada) e $4,5 \text{ eV}$ (linha tracejada) (adaptado de [36]).

para o crescimento de nanotubos de determinado tipo e dimensões físicas, alinhados em um substrato de quartzo. Assim, são transferidos para o substrato do transistor, utilizando uma técnica de transferência de camada em baixas temperaturas. Dependendo da arquitetura do transistor, os eletrodos de fonte e dreno são então desenhados utilizando fotolitografia óptica tradicional e depositados utilizando um evaporador de feixe de elétrons [38]. Também são desenvolvidas aplicações em escala de *wafers*, como o computador implementado a partir da tecnologia CNTFET, por Shulaker et al. [39].

A deposição ou crescimento pode ser realizada com o *gate* sobre ou sob o nanotubo, aspectos descritos a seguir [14]:

1. *Gate* sobre o nanotubo (*top gating*): Este é o método mais simples para a fabricação do CNTFET. Além do *gate*, os eletrodos do dreno e da fonte também podem ser implementados sobre o nanotubo. Esta arquitetura permite a fabricação de dispositivos complexos com vários tipos diferentes de materiais de contato, incluindo metais normais, ferromagnéticos e supercondutores. Depois do crescimento e deposição dos nanotubos, os eletrodos podem ser desenhados por litografia de feixe de elétrons e *lift-off*. Embora a maior presença de impurezas e o desordenamento dos nanotubos introduzidos na fabricação sejam motivos de preocupação, transistores fabricados com esta

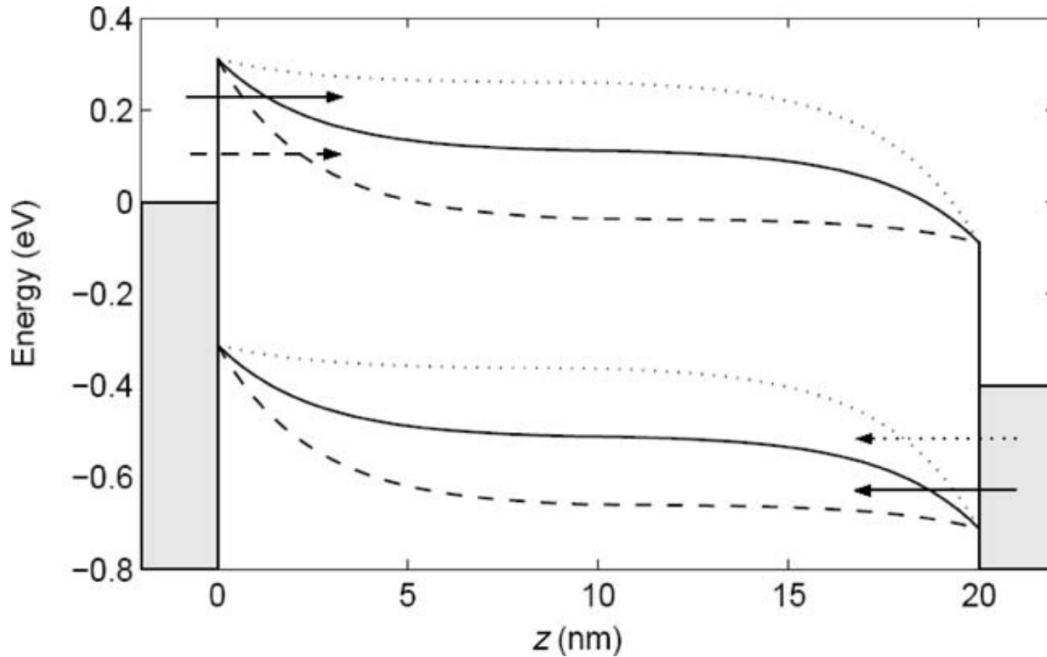


Figura 2.6: Diagramas de bandas de energia de um CNTFET de barreira Schottky, com nanotubo de diâmetro $D_{cnt} = 1,26$ nm, comprimento $L_{cnt} = 20$ nm; funções trabalho do nanotubo e de todos os metais dos eletrodos iguais a $4,5$ eV; $V_{ds} = 0,4$ V. Injeção de lacunas com $V_{gs} = 0,05$ V (linhas e setas pontilhadas); injeção de lacunas e elétrons com $V_{gs} = 0,2$ V (linhas e setas contínuas); injeção de elétrons com $V_{gs} = 0,35$ V (linhas e setas tracejadas) (adaptado de [36]).

arquitetura possuem uma interface metal-semicondutor (ou metal-nanotubo) suficientemente limpa para o transistor demonstrar operação ambipolar. Um das vantagens de se utilizar esta arquitetura é a de minimizar variações induzidas por moléculas de água ou de oxigênio nos nanotubos, fatores conhecidos por causar histerese na corrente do transistor [40].

2. *Gate* sob o nanotubo (*bottom gating*): Uma desvantagem da arquitetura de *top gating* é o fato de que o próprio processo de fabricação pode induzir uma certa desordem nos nanotubos. Uma alternativa é o crescimento e depósito dos mesmos sobre eletrodos predefinidos, resultando em dispositivos com menos impurezas e controle melhorado. Contudo, há a desvantagem do baixo rendimento de fabricação, pois o crescimento dos nanotubos entre dreno e fonte e sobre o *gate* depende de uma determinada probabilidade.
3. Nanotubo envolvido pelo *Gate* (*wrap-around gating*): fabricado pela primeira vez por Franklin et al. [35] (figura 2.7), este tipo de arquitetura permite um maior controle sobre o transistor, visto que o eletrodo do *gate* circunda completamente o nanotubo.

Além disso, o isolamento dos nanotubos os protege contra elementos externos, como por exemplo cargas que possam causar instabilidade no funcionamento do transistor.

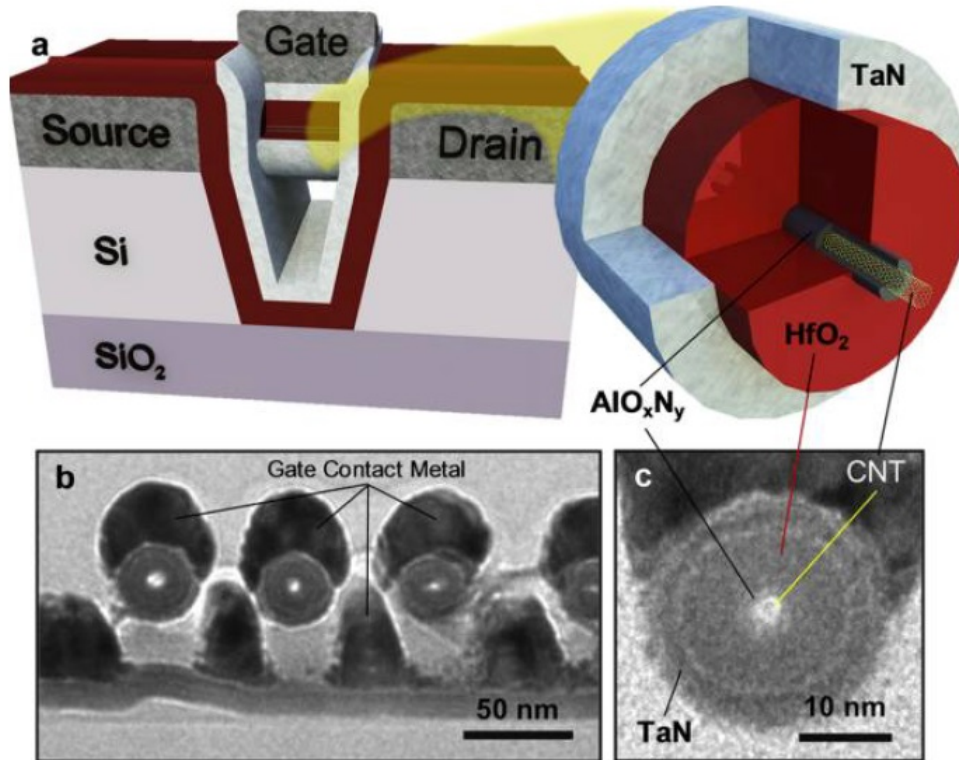


Figura 2.7: (a) Diagrama do CNTFET ilustrando como o canal do nanotubo (corte mostrado à direita) é suspenso através da trincheira de silício e conectado em ambos os lados de fonte e dreno. (b) e (c) mostra imagens TEM de nanotubos na geometria *wrap-around* [41].

A complexidade da estrutura de nanotubos de paredes múltiplas desencoraja o seu uso e estudo detalhado. Como consequência, há poucos CNTFETs produzidos e caracterizados com paredes múltiplas. A princípio, cada nanotubo pode ter comportamento metálico ou semicondutor, com diferentes quiralidades. Além disso, os mesmos podem interagir, podendo possuir propriedades não desejadas em determinadas aplicações. Em nanotubos de paredes múltiplas conectados a eletrodos metálicos, somente os nanotubos externos contribuem para o transporte de carga. Entretanto, pelo *bandgap* dos nanotubos ser inversamente proporcional ao seu diâmetro, somente nanotubos de paredes múltiplas de pequeno diâmetro possuem características que possam ser utilizadas em transistores, em temperatura ambiente [30]. A título de simplificação, o modelo de CNTFET utilizado no presente trabalho possui somente nanotubos de parede única.

Para fins de análise e testes em laboratório, há ainda um método de transferência mecânica de nanotubos de carbono, onde os mesmos ficam suspensos sobre o *gate*. Este método

obtem vantagens de ambos os métodos citados anteriormente, obtendo alto nível de sensibilidade às tensões incididas no eletrodo do *gate* sem processamento após a transferência. O substrato do transistor (sem nanotubos) e o substrato de crescimento (com os nanotubos) são fabricados separadamente. Antes do ensaio de medida propriamente dito, um único nanotubo é transferido do substrato de crescimento para o transistor usando um processo de estampagem alinhada. Utilizando manipuladores controlados por sondas piezo, a transferência é possível em vácuo a temperaturas criogênicas, permitindo que o nanotubo seja testado livre de impurezas e *in situ*. Aplicando uma força ao nanotubo, a condução do mesmo é alterada [42], possuindo assim o potencial de construção, e.g., de um dispositivo acelerômetro.

Outro desafio é a purificação dos nanotubos semicondutores, ou seja, a seleção de nanotubos a serem inseridos nos transistores, eliminando os nanotubos metálicos. Alguns métodos, como por exemplo os métodos físicos de dielectroforese e gradiente de densidade por centrifugação e os métodos químicos de adsorção em bromo e diazotização são utilizados. Porém, não possuem um rendimento alto, além de exigirem consideráveis recursos no seu emprego. Voggu et al. [43] desenvolveram um método relativamente simples para separar nanotubos de carbono semicondutores de parede simples dos metálicos através de transferência de carga molecular, utilizando uma solução de sal de potássio de ácido tetracarboxílico coroneno. Esta técnica possui a vantagem de resultar em altas concentrações de nanotubos semicondutores, em oposição às técnicas citadas anteriormente, onde a concentração de nanotubos metálicos ainda é relevante.

Uma técnica que não chega a envolver a purificação mas a “desconexão” seletiva de nanotubos metálicos, chamada *VMR* [44], permite “desconectar” nanotubos metálicos aplicando brevemente uma alta tensão, destruindo-os. Esta técnica é compatível com transistores utilizados em circuitos lógicos digitais, de larga integração.

2.2 MODELAGEM DO CNTFET

Uma revisão crítica dos modelos compactos é dado em [18] e [45]. Os modelos presentes em [46, 47, 48, 49, 50] focam somente em aplicações digitais e de baixa tensão e, além disso, a corrente de dreno estática e a modelagem de carga nos nanotubos não-linear é negligenciada ou muito simplificada para simulações RF. Os modelos apresentados em [46], [48] e [51] descrevem a corrente de dreno estática como uma solução explícita da equação de Landauer. Como mostrado em [45], até mesmo derivações de primeira ordem correspondentes e mais o comportamento em baixa frequência de pequenos sinais de dispositivos fabricados pode vir a ser pouco fiéis. As descrições de carga de [47] e [48] são baseados na aproxima-

ção de pseudo-massa [45], em que não há um valor compatível com os dados de capacitância obtidos a partir de ambos os resultados experimentais, além da simulação do dispositivo.

A tentativa de incluir o efeito da barreira de Schottky na fonte e dreno através da aproximação de Wentzel–Kramers–Brillouin resulta em uma expressão implícita [52] que envolve considerável esforço computacional para ser resolvida, fato não recomendável em modelos compactos. O modelo descrito em [53] descreve a dependência não-linear de correntes e cargas em relação à polarização com polinômios de terceira ordem, cujos coeficientes são determinados a partir de simulações do dispositivo.

Uma extensão para ordens superiores, onde são necessárias para aplicações práticas, seria onerosa em termos da determinação dos parâmetros. Todos os modelos mencionados anteriormente tem desvantagens que fazem com que não seja recomendado o seu uso no projeto de circuitos analógicos RF. Uma das razões em que modelos puramente baseados em parâmetros físicos falham em descrever características obtidas experimentalmente é o conhecimento limitado de dispositivos fabricados, especialmente no que tange os efeitos de transporte e armazenamento de carga [26]. Logo, nesta dissertação foi utilizado um modelo semi-físico, que atinge as necessidades de aplicações práticas em termos de precisão e extração de parâmetros para dispositivos experimentais, assim como para eficiência computacional.

3

MÉTODO DE MONTE CARLO E A TRANSFORMADA DA INCERTEZA

O método de Monte Carlo pode ser definido como um conjunto de algoritmos computacionais que realiza repetidos sorteio aleatórios para se obter resultados numéricos. A ideia central consiste em usar a aleatoriedade para se resolver problemas que, a princípio, são determinísticos. Ele é utilizado basicamente em problemas de estimação (como por exemplo integração numérica), otimização e para se gerar valores a partir de um distribuição (ou mais) de probabilidade, sendo o último o grupo de problemas em que o método é aplicado neste trabalho. Por ser um método onde se utiliza amostras aleatórias em grande número, o esforço computacional necessário para este cálculo é notadamente dispendioso.

A previsão das funções de distribuição de probabilidade utilizando o método de Monte Carlo é amplamente utilizado na indústria de semicondutores, como por exemplo no uso em programas de projeto e simulação de CI's [54]. No caso de rendimento de produção de semicondutores, onde aspectos físicos possam sofrer variações, tais como largura e comprimento de transistores e densidade de dopantes, as simulações de circuitos eletrônicos utilizando o método de Monte Carlo podem prever os limites no funcionamento e conseqüentemente o rendimento da fabricação destes circuitos. Citando alguns exemplos de pesquisa acerca deste assunto, pode-se citar o estudo do melhoramento do rendimento por redundância de nanotubos por Ashraf et al. [55], o estudo do rendimento na manufatura de equipamentos eletrônicos por Huang [56], o rendimento de CI's de SRAM por Doorn et al. [57], dentre muitos outros mais.

Embora tenha uma ampla gama de aplicações, este método possui passos relativamente definidos para sua execução. Estes passos normalmente sofrem adaptações, dependendo da aplicação. Eles podem ser resumidos em:

1. Caracterizar a(s) entrada(s) do sistema (ou a variável de entrada em questão) como uma função de densidade de probabilidade (PDF);
2. Sortear, de acordo com a(s) PDF's da(s) entrada(s), valores que serão usados como entrada no sistema. O número de valores sorteados varia de acordo com o grau de convergência desejado;
3. Computar e analisar a saída do sistema.

A formulação do problema e da análise a ser feita é essencial para a realização da simulação utilizando o método e conseqüentemente a formulação correta do modelo em questão. Os pontos-chave para a definição da função não-linear do sistema utilizada na simulação são as saídas desejadas, a utilização das saídas, a acurácia/precisão das saídas, a exatidão da definição das entradas, a exatidão do modelo e a modelagem dos processos inerentes.

As saídas desejadas definem basicamente o objetivo da simulação, visto que são a partir delas que se respondem as questões relacionadas ao sistema. A precisão das saídas normalmente é diretamente ligada ao número de valores sorteados usados como entrada do sistema. Entretanto, faz-se necessário sempre analisar o *trade-off* entre a acurácia/precisão e o esforço computacional necessário. A exatidão da definição das entradas confere uma maior fidelidade à realidade, embora esta fidelidade seja também influenciada por outros elementos. Já a exatidão do modelo é definida pelas funções matemáticas inerentes ao sistema. Quanto mais próximas à realidade, mais fiel o modelo será do sistema real.

A geração das entradas da simulação de Monte Carlo é o aspecto principal neste tipo de simulação. As entradas são geradas a partir de Funções de Densidade de Probabilidade (PDF), que define a faixa de valores possíveis para determinado passo da simulação, bem como sua probabilidade relativa. A faixa dos valores possíveis deve possuir valores reais não-negativos e a integral desta faixa deve ser necessariamente igual a um.

No que tange às simulações realizadas no presente trabalho, os processos inerentes são tidos como transformações não-lineares numéricas, dadas na forma de simulações de circuitos elétricos.

3.1 TRANSFORMADA DA INCERTEZA

A ideia básica da UT é baseada na premissa de que “*é mais fácil aproximar uma distribuição de probabilidade do que aproximar uma função arbitrária não linear ou uma transformação*” [58]. Um conjunto de pontos, chamados de pontos sigma, é calculado a partir de determinada entrada do sistema em questão. Eles não são randômicos e sim deterministicamente escolhidos para que tenham determinadas propriedades, como por exemplo, mesma média e variância. Como consequência os momentos da distribuição de determinada entrada podem ser traduzidos por um pequeno número fixo de pontos. Em seguida, a função, linear ou não, é aplicada no cálculo de cada ponto sigma, do sistema em questão, tendo como saída um conjunto de pontos sigma e seus respectivos pesos, que possuem os mesmos momentos (considerando os erros associados ao número de pontos sigma e à operação) da saída da função quando aplicada a um conjunto contínuo de dados na entrada.

A UT faz, inicialmente, uma aproximação discreta de, por exemplo, uma função densidade de probabilidade. Tomando como exemplo, a Figura 3.1 mostra a aproximação de uma PDF de uma distribuição normal com média $\mu = 5$ e variância $\sigma^2 = 2$.

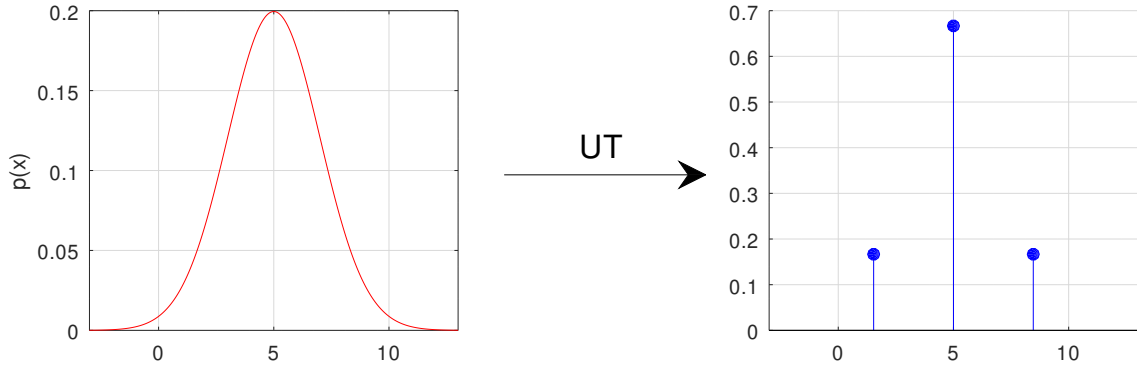


Figura 3.1: Transformação em três pontos sigma utilizando a UT de uma função distribuição normal com $\mu = 5$ e $\sigma^2 = 2$.

Um conjunto de pontos sigma S é constituído de $p + 1$ vetores e de seus pesos associados $S = \{i = 0, 1, \dots, p : \mathbf{s}^{(i)}, w^{(i)}\}$. Os pesos $w^{(i)}$, assim como os pontos sigma, podem ser tanto positivos quanto negativos. Porém, para fins de estimação não enviesados, estes pares devem obedecer à condição

$$\sum_{i=0}^p w^{(i)} = 1. \quad (3.1)$$

Em seguida, aplica-se os pontos sigma $s^{(i)}$ da entrada à função do sistema, obtendo assim os pontos sigma $z^{(i)}$ da saída do mesmo. Assim,

$$\mathbf{z}^{(i)} = h [s^{(i)}], \quad (3.2)$$

onde h é a função do sistema. A média e a variância do conjunto dos pontos sigma da saída são calculadas a partir da equações listadas abaixo.

- A média (μ_z) é calculada pela média ponderada dos pontos da transformação:

$$\mu_z = \sum_{i=0}^p w^{(i)} \mathbf{z}^{(i)} \quad (3.3)$$

- A variância (σ_z^2) é calculada a partir do produto ponderado dos pontos da transforma-

ção:

$$\sigma_z^2 = \sum_{i=0}^p w^{(i)} \{ \mathbf{z}^{(i)} - \mu_z \}^2 \quad (3.4)$$

O valores de outros momentos podem ser calculados de maneira similar.

A UT tem uma série de vantagens [24], considerando a sua simplicidade, a saber:

1. Pelo algoritmo funcionar com um número finito de pontos sigma, pode ser utilizado como se fosse uma “caixa preta” em forma de biblioteca. Dado um modelo, uma rotina padrão pode ser utilizada para se calcular os valores necessários para uma dada função.
2. O custo computacional do algoritmo é da mesma ordem de magnitude que do Filtro de Kalman Estendido (EKF, *Extended Kalman Filter*). As operações que mais demandam esforço computacional são calcular a raiz quadrada da matriz e de outros produtos necessários para calcular a variância dos pontos sigma projetados. Isso contrasta com métodos como por exemplo a quadratura de Gauss-Hermite, pelo número de pontos necessários aumentar geometricamente com o número de dimensões.
3. O algoritmo pode ser utilizado em funções com descontinuidades. Os pontos sigma podem abranger uma descontinuidade e ainda aproximar o efeito da descontinuidade na estimativa transformada.
4. A técnica pode, em determinadas aplicações, explorar redundâncias no sistema analisado, reduzindo assim o esforço computacional exigido.

3.2 CÁLCULO DOS PONTOS SIGMA E DE SEUS PESOS

A transformada da incerteza envolve uma extensa teoria. No presente trabalho, foi utilizada a implementação dos cálculos dos pontos sigma e de seus pesos feita por Medeiros [59] em Matlab, utilizando como base suas definições. A UT é baseada na quadratura mecânica, um processo arbitrário que faz o mapeamento de somas de amostras com pesos associados de uma função $h(x)$ em números. O caso de interesse na formulação da UT é tal cujos pesos w_i são definidos como

$$\int_a^b h(x)p(x)dx = \sum_{i=1}^n w^{(i)} h(x^{(i)}) \quad (3.5)$$

Os passos para o cálculo dos pontos sigma e de seus pesos podem ser organizados no seguinte algoritmo:

1. Defina uma função de densidade de probabilidade $p(x)$ das variáveis de o número de pontos sigma n . Calcule os termos α_i e β_i , $i = 1, \dots, n$, pertencentes à relação de recorrência de três termos, utilizando as equações:

$$\rho_{-1}(x) = 0, \quad (3.6)$$

$$\rho_0(x) = 1 \quad (3.7)$$

e, para ordens superiores,

$$\rho_{i+1}(x) = (x - \alpha_{i+1})\rho_i(x) - \beta_{i+1}\rho_{i-1}(x). \quad (3.8)$$

Sendo α e β calculados a partir das equações

$$\alpha_{i+1} = \frac{\langle x\rho_i(x), \rho_i(x) \rangle}{\langle \rho_i(x), \rho_i(x) \rangle}, \quad \text{para } i \geq 0,$$

$$\beta_{i+1}^2 = \begin{cases} 1, & \text{para } i = 0, \\ \frac{\langle \rho_i(x), \rho_i(x) \rangle}{\langle \rho_{i-1}(x), \rho_{i-1}(x) \rangle}, & \text{para } i \leq 0. \end{cases}$$

O produto interno $\langle f, g \rangle$ é definido como

$$\langle f, g \rangle = \int_a^b f(x)g(x)p(x)dx,$$

onde as funções f e g são definidas no intervalo $[a, b]$. O objetivo é construir polinômios ortogonais, ou seja, $\langle \rho_i, \rho_k \rangle = 0$.

2. Construa a matriz jacobiana J_n :

$$J_n = \begin{bmatrix} \alpha_1 & \sqrt{\beta_2} & 0 & 0 & 0 & \dots & 0 \\ \sqrt{\beta_2} & \alpha_2 & \sqrt{\beta_3} & 0 & 0 & \dots & 0 \\ 0 & \sqrt{\beta_3} & \alpha_3 & \sqrt{\beta_4} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sqrt{\beta_{n-2}} & \alpha_{n-2} & \sqrt{\beta_{n-1}} & 0 \\ 0 & 0 & \dots & 0 & \sqrt{\beta_{n-1}} & \alpha_{n-1} & \sqrt{\beta_n} \\ 0 & 0 & \dots & 0 & 0 & \sqrt{\beta_n} & \alpha_n \end{bmatrix} \quad (3.9)$$

onde α_i e β_i são os coeficientes da relação de recorrência de três termos, definidos

acima;

3. Calcule os autovalores de J_n utilizando qualquer método conhecido;
4. Calcule os autovetores de J_n utilizando qualquer método conhecido e dimensione-os utilizando o teorema abaixo. Utilize as componentes do primeiro autovetor como os pesos λ_i ;

Teorema: Seja $v^{(i)} = (v_1^{(i)}, \dots, v_n^{(i)})^T$ um autovetor de J_n , definido pela equação (3.9), para o autovalor X_i , i.e., $J_n v^{(i)} = x_i v^{(i)}$. Suponha que $v^{(i)}$ seja adotado de tal maneira na qual

$$v^{(i)T} v^{(i)} = \langle \rho_0, \rho_0 \rangle = \int_a^b 1 \cdot 1 \cdot p(x) dx = 1.$$

Logo, os pesos w_i serão dados por

$$w_i = (v_1^{(i)})^2, \quad i = 1, \dots, n.$$

5. Dimensione os autovalores de J_n e pesos λ_i de acordo com a função de densidade de probabilidade desejada e utilize-os como pontos sigma s_i e pesos w_i da UT.

4

ASPECTOS TÉCNICOS

O cerne deste trabalho é a utilização da transformada da incerteza para se avaliar a variabilidade em tecnologias emergentes, mais especificamente as tecnologias baseadas em CNT-FET. A utilização da UT tem a principal vantagem de se exigir uma quantidade menor de tempo, mantidas certas condições. Para isso, foram utilizados o Modelo Compacto de Transistor de Nanotubo de Carbono (CCAM), o simulador de circuitos elétricos Ngspice [28], necessário para a realização das simulações dos circuitos eletrônicos baseados na tecnologia CNTFET e um programa desenvolvido no *software* GNU Octave [27], o CNTFET-Workflow.

Por não ser utilizada normalmente no projeto e simulação de circuitos integrados, a UT e a posterior comparação com o método de Monte Carlo requereu a criação de um programa, o CNTFET-Workflow, que realizasse tanto a definição dos parâmetros de entrada da simulação quanto os passos da simulação em si. A Figura 4.1 mostra o fluxograma da realização de uma simulação utilizando o programa dito anteriormente.

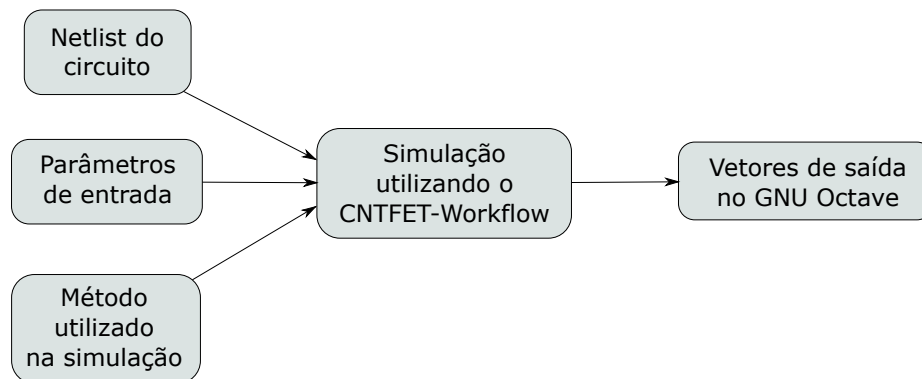


Figura 4.1: Fluxo de trabalho da simulação.

Primeiramente define-se o circuito a ser simulado, descrevendo-o na *netlist* a ser utilizada como arquivo de entrada para o Ngspice. Presta-se atenção para o fato de se descrever os parâmetros elétricos dos CNTFETs do circuito como variáveis, e não como valores numéricos, pois serão inseridos no final do arquivo da *netlist* pelo CNTFET-Workflow em cada passo da simulação. Para se descrever um transistor, por exemplo, de nome M1, basta inserir na *netlist* o seguinte código:

```
1 .MODEL mod1 nmos level=99
2 +idstn=M1_idstn vthin=M1_vthin vth0in=M1_vth0in athin=M1_athin
3 +sthin=M1_sthin idstp=M1_idstp vthip=M1_vthip vth0ip=M1_vth0ip ...
```

```

    athip=M1_athip
4 +sthip=M1_sthip vdcr=M1_vdcr betdcr=M1_betdcr smss=M1_smss ...
    facss=M1_facss
5 +rscs=M1_rscs rdcs=M1_rdcs ctn0=M1_ctn0 ctp0=M1_ctp0
6 +vthqs=M1_vthqs vth0qs=M1_vth0qs athqs=M1_athqs vthqd=M1_vthqd ...
    vth0qd=M1_vth0qd
7 +athqd=M1_athqd pqsd=M1_pqsd rmta=M1_rmta amto=M1_amto
8 +cmt=M1_cmt pqmt=M1_pqmt rscm=M1_rscm rdcm=M1_rdcn
9 +fanof=M1_fanof fanofmt=M1_fanofmt hoogef=M1_hoogef ...
    beta_fn=M1_beta_fn rsf=M1_rsf rdf=M1_rdf
10 +rg=M1_rg cgsparl=M1_cgsparl cgsparl2=M1_cgsparl2 cgdparl=M1_cgdparl
11 +cgdpar2=M1_cgdpar2 cdspar=M1_cdspar tnom=M1_tnom rths=M1_rths ...
    rthm=M1_rthm
12 +zetrmta=M1_zetrmta zetamto=M1_zetamto alvthin=M1_alvthin ...
    alvthin0in=M1_alvthin0in alvthip=M1_alvthip alvthin0ip=M1_alvthin0ip
13 +alidst=M1_alidst alct0=M1_alct0 alvthqs=M1_alvthqs ...
    alvthin0qs=M1_alvthin0qs alvthqd=M1_alvthqd alvthin0qd=M1_alvthin0qd
14 +alrscon=M1_alrscon alrmcon=M1_alrmcon atrap=M1_atrap ...
    btrap=M1_btrap ctrap=M1_ctrp w0trap=M1_w0trap strap=M1_strap

```

Para inserir outros transistores na *netlist*, basta substituir no código acima o nome M1 pelo nome do transistor desejado. Os parâmetros elétricos declarados serão explicados posteriormente na seção 4.1.

A seguir, define-se os parâmetros físicos que sofrerão variação durante a simulação. Estes parâmetros podem ser definidos como funções de distribuições de probabilidade uniforme e normal. Os parâmetros físicos serão apresentados e descritos na seção 4.2. Por fim, seleciona-se o método utilizado na simulação entre o método de Monte Carlo e a transformada da incerteza.

Depois de realizada a simulação, os resultados ou vetores de saída da simulação são salvos na área de trabalho do GNU Octave. Mais detalhes sobre o funcionamento do CNTFET-Workflow serão apresentados na seção 4.2.

4.1 MODELO COMPACTO DE TRANSISTOR DE NANOTUBO DE CARBONO - CCAM

Para simular circuitos elétricos baseados na tecnologia CNTFET, o uso de um modelo do transistor de nanotubos de carbono torna-se necessário. Como modelos que descrevem

os fenômenos físicos inerentes ao dispositivo em questão não são rápidos o bastante para se utilizar em ferramentas de alto nível de abstração, como por exemplo ferramentas SPICE, o emprego de modelos empíricos (chamados comumente de modelos compactos) faz-se necessário.

Para auxiliar no projeto de circuitos baseados em tecnologia CNTFET e prover um melhor entendimento das tecnologias emergentes, um modelo circuito equivalente é necessário. O CCAM [26] é um modelo semi-físico do transistor de efeito de campo de nanotubos de carbono, aplicável a sistemas digitais, analógicos e de alta frequência. Sendo escrito em Verilog-A, ele descreve o comportamento dos sinais DC e de pequenos sinais de CNTFETs fabricados, modelando-os como o circuito equivalente mostrado na Figura 4.2. Para ser utilizado pelo Ngspice, o modelo é traduzido para linguagem C pelo programa ADMS. Este programa converte modelos escritos em Verilog-AMS em código escrito em C para ser utilizado na API de simuladores SPICE. O código gerado é compilado no arquivo executável do simulador e o novo modelo de dispositivo pode ser utilizado em simulações [60].

Este modelo descreve com acurácia a forma das curvas características DC e de pequenos sinais de CNTFETs fabricados. Ele permite, para um dado comprimento de *gate*, o escalonamento da geometria de um transistor com um único dedo do *gate* e um único nanotubo até um transistor com múltiplos dedos do *gate* e múltiplos nanotubos. Outras características incluem capacitâncias parasitas, transporte ambipolar, comportamento dinâmico e modelagem de armadilhas de carga. Este modelo mostra grandes semelhanças entre os dados tanto da equação de transporte de Boltzmann quanto de medidas em CNTFETs de barreira Schottky. O modelo foi implementado em Verilog-A, fazendo com que seja utilizável em vários simuladores de circuitos elétricos. O modelo de CNTFET apresentado e utilizado nas simulações é o transistor apresentado e caracterizado em [26].

O estado-da-arte atual do CNTFET permite uma performance superior sob operações em alta frequência, tendo a maior transcondutância com uma capacitância de canal relativamente baixa na tecnologia FET [61]. Como consequência, tem-se um grande potencial para aplicações como, por exemplo, sistemas RF. Os principais desafios [61] da tecnologia CNTFET são basicamente obter-se uma maior taxa semiconductor-metal (s/m) e uma maior densidade de nanotubos de carbono através do CNTFET. Estas características provêm uma melhor aplicação em sistemas de alta frequência, com seu requerimento típico de impedância de 50Ω .

A Figura 4.2 mostra o circuito equivalente do CCAM. A porção externa, composta pelos componentes R_{df} , R_{sf} , R_g , $C_{gd,par}$, $C_{gs,par}$ e $C_{ds,par}$ modela a resistências e capacitâncias dos “dedos” do *gate*, dreno e fonte, representando a metalização das conexões. Como os nanotubos semicondutores e metálicos têm diferentes comportamentos eletrônicos, possuem

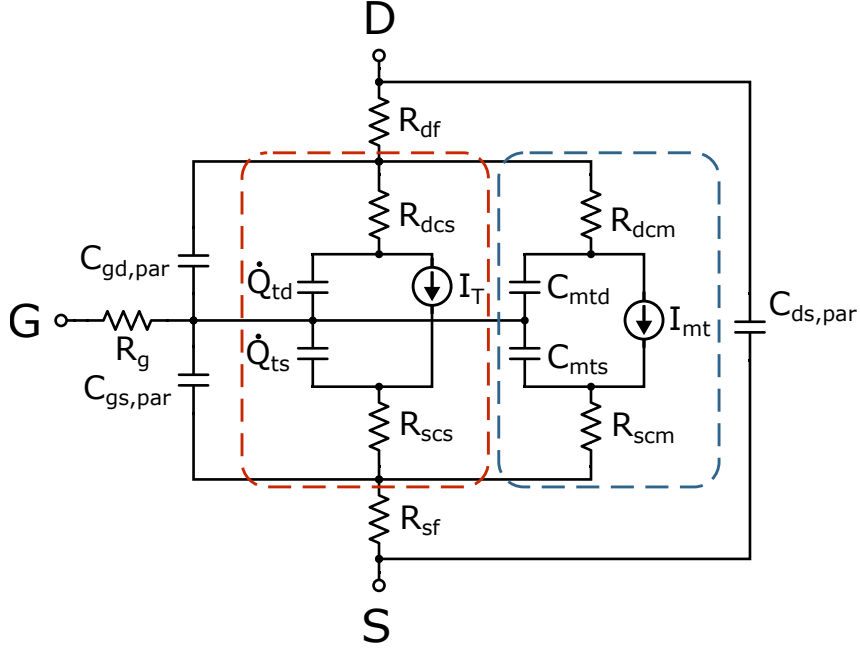


Figura 4.2: Circuito equivalente do CCAM. A parte hachurada vermelha é responsável pela modelagem dos nanotubos semicondutores, enquanto que a parte azul modela os nanotubos metálicos (adaptado de [15]).

diferentes representações no circuito. A parte hachurada vermelha do circuito modela a porção de nanotubos semicondutores, enquanto que a parte hachurada azul modela os nanotubos metálicos. Uma rede de aprisionamento de carga modela o atraso referente às cargas do transistor em simulações transientes [62], enquanto que uma rede térmica modela a influência da temperatura no transistor [15]. Ambas as redes citadas não são utilizadas na modelagem utilizada nesta dissertação.

As propriedades intrínsecas da parte responsável pelo canal semicondutor, isto é, a corrente de carregamento $\dot{Q}_{td/s}$, a corrente de transferência I_T e as resistências $R_{d/s,cs}$ escalonam com n_{sem} (densidade de nanotubos semicondutores), contidos na área hachurada em vermelho, mostrada na Figura 4.2, enquanto que as componentes equivalentes para o canal metálico C_{mt} , I_{mt} , e $R_{d/s,cm}$ escalonam com n_{mt} (densidade de nanotubos metálicos), contidos na área hachurada em azul (*idem*). O escalonamento das capacitâncias extrínsecas, fora das áreas hachuradas na Figura 4.2, depende somente da geometria do transistor, cujas relações são dadas pelas Eqs. (4.1), (4.2) e (4.3) [3]

$$\frac{C_{gs}}{C_{gs,ref}} = \frac{n_{gf}}{n_{gf,ref}} \cdot \frac{1 + W_g \cdot a_{ref}}{1 + W_{g,ref} \cdot a_{ref}} \quad (4.1)$$

$$\frac{C_{gd}}{C_{gd,ref}} = \frac{n_{gf}}{n_{gf,ref}} \cdot \frac{W_g}{W_{g,ref}} \quad (4.2)$$

$$\frac{C_{ds}}{C_{ds,ref}} = \frac{3/2 + n_{gf}}{3/2 + n_{gf,ref}} \quad (4.3)$$

As equações de modelagem das fontes I_T e I_{mt} , assim como a modelagem de cargas, são apresentadas e discutidas em [15].

Para se utilizar o modelo, faz-se necessária a sua calibração com resultados de medidas obtidas a partir de dispositivos fabricados, medidos e caracterizados. Para tal, atribui-se valores para os parâmetros elétricos do modelo. A ideia básica é que parâmetros de calibração do modelo pode ser extrapolados para novos valores, escalonando as equações descritas através de parâmetros físicos do CNTFET.

A Tabela A.1 lista os parâmetros que compõem o CCAM. Por se tratar de um modelo semi-físico, estes parâmetros ditam o comportamento do transistor, embora nem sempre traduzam diretamente os parâmetros físicos do mesmo. A Figura 4.3 mostra os gráficos de curvas de transferência e de saída obtidas a partir de [35]. Nota-se uma ótima aproximação dos resultados medidos (símbolos) com a modelagem do CCAM (linhas contínuas). A Figura 4.4 mostra as medidas de saídas obtidas para diferentes valores de V_{ds} . Com a calibração correta do modelo, o mesmo pode ser utilizado em diversas simulações de aplicações, possibilitando assim o desenvolvendo de projetos baseados em CNTFET antes mesmo de sua fabricação em larga escala.

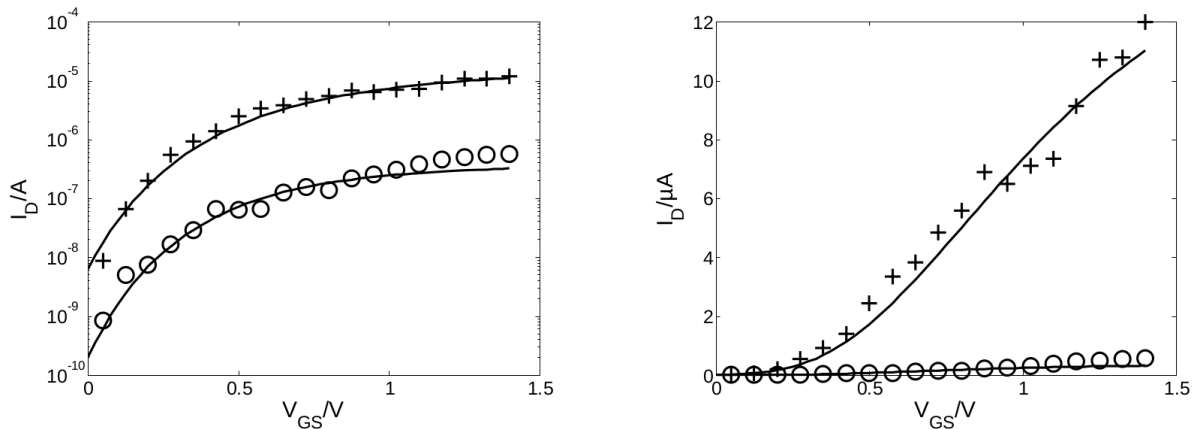


Figura 4.3: Medidas (símbolos) e modelagem (linhas contínuas) de curvas de transferência de um CNTFET de um único nanotubo com comprimento do canal $L_{ch} = 9$ nm. Gráfico da esquerda: escala logarítmica; gráfico da direita: escala linear. $V_{ds} = 0,01$ V (símbolo de adição), $V_{ds} = 0,4$ V (círculo) [15].

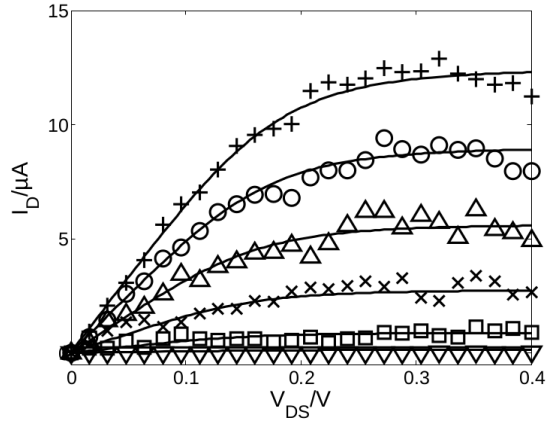


Figura 4.4: Medidas (símbolos) e modelagem (linhas contínuas) de curvas de saída de um CNTFET de um único nanotubo com comprimento do canal $L_{ch} = 9$ nm. $V_{gs} = 0, 1$ a $1,1$ V, com incrementos de $0,2$ V [15].

No Capítulo 2 foi relatada uma característica útil para determinados tipos de aplicações do CNTFET, a ambipolaridade. Por se tratar de um modelo semi-físico, é possível emular esta característica bastando apenas permutar determinados parâmetros do modelo, sendo eles:

- $id_{stn} \leftrightarrow id_{stp}$;
- $v_{thin} \leftrightarrow v_{thip}$;
- $v_{th0in} \leftrightarrow v_{th0ip}$;
- $a_{thin} \leftrightarrow a_{thip}$;
- $s_{thin} \leftrightarrow s_{thip}$;

Com isso, obtém-se uma certa simetria na curva de transferência do modelo, ilustrada na Figura 4.5. O modelo, quando configurado para se comportar como tipo P, apresenta as curvas mostradas aplicando-se valores negativos de V_{gs} , enquanto que se comporta como um transistor tipo N se configurado para tal, para valores positivos de V_{gs} .

4.2 CNTFET-WORKFLOW

Para que as simulações propostas no trabalho presente fossem realizadas, seria necessário um grande trabalho de manejo tanto dos dados de entrada quanto de saída, devido ao grande número de passos de cada simulação. Um método de automatização deste processo

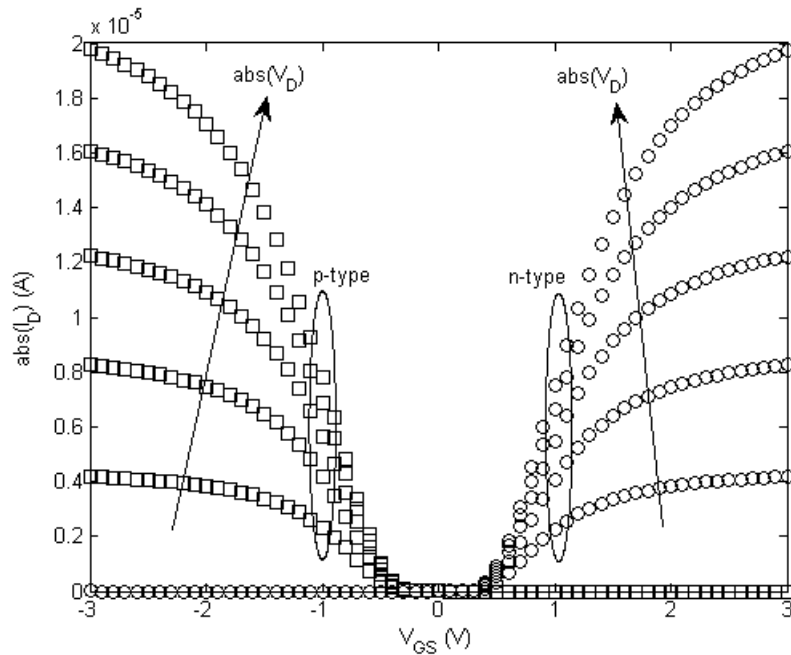


Figura 4.5: Curvas de transferência para vários valores de V_d .

fez-se necessário, tanto para a elaboração dos dados de entrada quanto do processo de alteração da *netlist* do circuito a ser simulado, além do processamento dos resultados. Para tanto, foi criado o programa CNTFET-Workflow, desenvolvido no GNU Octave, com o intuito de automatizar e facilitar a elaboração deste tipo de simulação de circuitos elétricos. O programa permite a automatização da configuração da simulação, da simulação em si utilizando o Ngspice e a importação dos dados de saída da simulação para a área de trabalho do GNU Octave. A Figura 4.6 mostra as etapas da configuração de cada tipo simulação.

Primeiro, descreve-se a topologia do circuito a ser simulado em um arquivo de *netlist*. A sintaxe utilizada na descrição do circuito deve necessariamente ser compatível com a sintaxe do Ngspice. A *netlist* deve conter o código de descrição de cada CNTFET do circuito. Elaborado o arquivo de *netlist*, faz-se então a configuração do programa. Embora seja realizado na linha de comando do GNU Octave, o uso e configuração do programa é relativamente simples. Nesta etapa, são definidos os arquivos e pastas de arquivos que o programa irá utilizar nas simulações.

Em seguida, é definido o método de simulação, sendo as opções possíveis a distribuição uniforme, o método de Monte Carlo e a transformada da incerteza. Vale ressaltar que múltiplos parâmetros físicos de múltiplos CNTFETs podem ser variados, praticamente não havendo limite para este número. Dependendo do método de simulação, entretanto, pode-se acarretar um número excessivo de passos a serem realizados durante a simulação. Como o número de passos varia de acordo com cada método, o seu número de passos de cada simula-

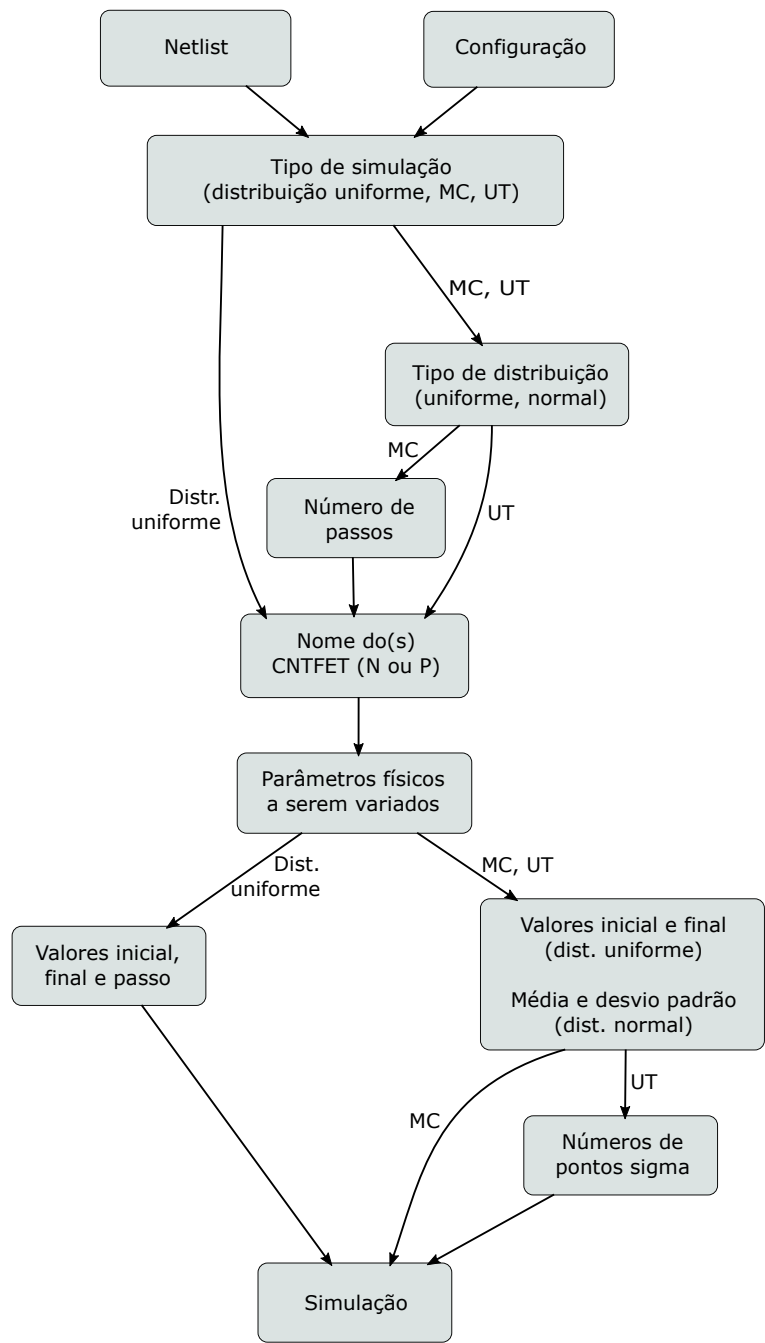


Figura 4.6: Configuração do CNTFET-Workflow.

ção deve ser calculado e analisado anteriormente. Os métodos de simulação estão descritos abaixo:

- **Distribuição uniforme:** como o próprio nome diz, este tipo de simulação faz uma variação uniforme dos parâmetros a serem variados. Escolhe-se o nome do CNTFET a ter algum parâmetro físico variado, escolhendo-se em seguida seus valores inicial, final e o tamanho do passo entre os valores anteriores. Percebe-se que neste método não é introduzido o número de passos, e sim o tamanho entre os valores dos passos entre os valores inicial e final. O número total de passos da simulação é dado pela multiplicação dos passos de cada parâmetro físico de cada transistor. Em seguida, é realizada a simulação.
- **Monte Carlo:** utilizando o método de Monte Carlo escolhe-se inicialmente o tipo de distribuição a ser utilizada no sorteio dos parâmetros físicos entre as distribuições uniforme e normal. Em seguida, define-se o número de passos que a simulação terá. Define-se os nomes dos CNTFETs, além dos parâmetros físicos de cada transistor que sofrerão variação. Dependendo do tipo de simulação, define-se os valores inicial e final, se escolhida a distribuição uniforme, ou média e desvio padrão se escolhida a distribuição normal. Assim, são sorteados valores de acordo com a probabilidade da distribuição escolhida. Logo, inicia-se a simulação.
- **Transformada da incerteza:** este último método, embora partilhando certas similaridades com o método de Monte Carlo, possui certas características chave. A principal delas é a ausência da definição do número de passos que a simulação terá, embora a definição dos parâmetros físicos e as opções de funções de distribuição sejam as mesmas. Tal fato é justificado pela definição do número de pontos sigma de cada parâmetro físico a sofrer variação, definido junto dos valores da distribuição do mesmo. Como consequência, o número de passos total da simulação será dado pela multiplicação do número de passos (ou pontos sigma) de cada parâmetro físico de cada transistor. Por ser um método determinístico, os valores calculados sempre serão os mesmos, sendo os valores dos parâmetros de cada passo da simulação a combinação de cada passo de cada parâmetro físico de cada transistor.

A realização da simulação se dá por um conjunto de processos, realizados em ordem sistemática, ilustrada na Figura 4.7.

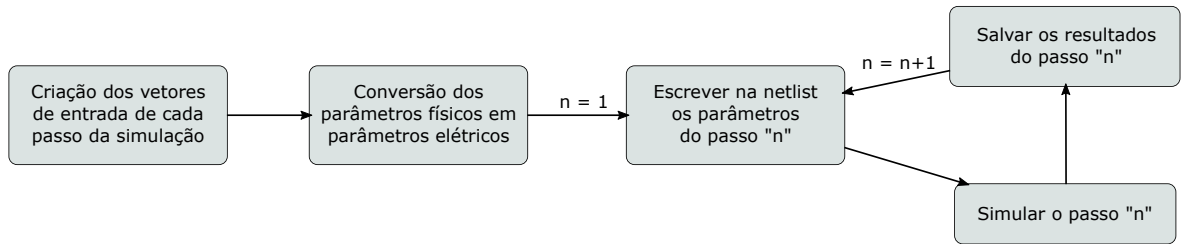


Figura 4.7: Simulação utilizando do CNTFET-Workflow.

Inicialmente os vetores de entrada de cada passo da simulação são criados, definidos pela configuração do programa, feita anteriormente. Esta etapa é responsável por fazer a combinação dos vetores de cada parâmetro físico de cada transistor em cada passo da simulação, de acordo com a configuração feita. Em seguida, realiza-se a conversão dos parâmetros físicos de cada transistor em parâmetros elétricos. Tais parâmetros são essenciais para a realização da simulação, visto que serão utilizados pelo Ngspice no circuito equivalente do CCAM.

Os parâmetros físicos utilizados no cálculo dos parâmetros elétricos são mostrados na Tabela 4.1. Todos eles possuem um valor base, obtidos a partir de um gerador de *modelcard*, conjunto de dados que possui as informações e medidas de um CNTFET, tais como suas dimensões físicas e características elétricas. A partir das medidas deste transistor, faz-se uma aproximação por *curve fitting*, alterando-se os parâmetros desejados. Os que não possuem os valores alterados são utilizados os valores base. Em seguida, escreve-se no final do arquivo da *netlist* os valores dos parâmetros elétricos do primeiro passo da simulação, utilizando o comando *.PARAM*, sintaxe própria do Ngspice, e executa-se o primeiro passo da simulação propriamente dita do circuito. O passo da simulação gera um arquivo de saída, contendo os resultados desejados dos parâmetros de saída. Estes dados são então lidos e salvos em uma variável do ambiente de trabalho do GNU Octave, para que sejam posteriormente analisados. Logo, repete-se as etapas ditas anteriormente para cada passo da simulação, até que todos os passos da mesma tenham sido executados. O programa mostra na linha de comando do GNU Octave os dados do Ngspice em modo *batch* durante a simulação. Assim, não é necessário nenhuma execução de comandos durante a simulação.

A tela inicial do CNTFET-Workflow é mostrada na Figura 4.8. Estão disponíveis cinco opções, sendo a primeira de configuração dos arquivos e pastas da simulação (*Setup simulation files and folders*), as três opções seguintes para cada tipo de simulação (*Linear scaling simulation*, *Monte Carlo simulation* e *Unscented transform simulation*) e a quinta e última opção sendo o menu de realizar a simulação (*Simulate*).

Na primeira opção, de configuração do CNTFET-Workflow, é possível fazer a seleção dos arquivos e pastas relativos à simulação. Pode-se mudar a pasta do arquivo de *netlist*, o nome

Parâmetro	Valor base	Significado físico
n_gf	10	Número de dedos do <i>gate</i>
w_g	4,1 μm	Largura do <i>gate</i>
l_g	0,18 μm	Comprimento do <i>gate</i>
l_gsd	0,225 μm	Gap entre <i>gate</i> e fonte/dreno
l_sd	6 μm	Comprimento do dedo de fonte/dreno
d_cnt	1,5 μm	Diâmetro do nanotubo
n_ta	40 μm^{-1}	Densidade de nanotubos
p_mt	0.0001	Fração de nanotubos metálicos
h_mg	0,19 μm	Altura da metalização do <i>gate</i>
h_msd	0,15 μm	Altura da metalização de fonte/dreno
h_m1	0,07 μm	Altura da camada de metal 1
t_ox	25 nm	Espessura do óxido
w_ms	6 μm	Largura do contato da fonte
w_mg	5 μm	Largura do contato do <i>gate</i>
w_do	3,1 μm	Sobreposição do contato entre fonte e dreno
eps_is	4	Constante do óxido acima dos nanotubos entre os contatos
eps_ox	10	Constante do óxido do <i>gate</i>
eps_ins	4	Constante do óxido sob os nanotubos e contatos

Tabela 4.1: Parâmetros físicos do CNTFET utilizados no cálculo dos parâmetros elétricos.

do arquivo de *netlist*, a pasta do arquivo do gerador de *modelcard*, a pasta onde serão realizadas as simulações, onde será salvo o arquivo dos dados de saída e o comando de execução do Ngspice, mostrado na Figura 4.9. É importante frisar que o arquivo de *netlist* possui também as instruções para salvar os dados de saída da simulação em um arquivo, normalmente nomeado *rawspice.raw*. As últimas mudanças possíveis são a alteração do nome arquivo do *modelcard*, assim como o nome da variável onde estão inseridos os parâmetros elétricos a serem utilizados no *curve-fitting*, equivalentes aos parâmetros físicos base do CNTFET.

Da segunda à quarta opção é possível fazer a escolha e configuração do tipo de simulação desejado, escolhendo-se também os parâmetros físicos a serem variados, o nome e o tipo dos transistores. A Figura 4.10 mostra um exemplo do ambiente de configuração dos parâmetros. Pode-se notar que foi escolhida a utilização da transformada da incerteza, com parâmetros modelados por distribuição normal. Foi escolhido o parâmetro *n_ta* (densidade de nanotubos) com uma média de $80 \mu\text{m}^{-1}$, desvio padrão de $10 \mu\text{m}^{-1}$ e três pontos sigma do transistor de nome *M1*, do tipo *N*.

```
CNTFET-Workflow
-> Menu select:
1. Setup simulation files and folders
2. Linear scaling simulation
3. Monte Carlo simulation
4. Unscented transform simulation
5. Simulate

  (By selecting 2,3 or 4 any previous simulation parameters will be erased.)
Select the item entering its index number:
```

Figura 4.8: Tela principal do CNTFET-Workflow.

```
-> Setup program folders

1. Circuit Netlist folder:
   /home/debian/Documents/Git/CNTFET-Workflow-simulations/ngspice_simulations/novo

2. Circuit Netlist file name:
   freq_3_nocap

3. Modelcard generator folder:
   /home/debian/Documents/Git/CNTFET-Workflow/src/modelGen

4. Simulation folder:
   /home/debian/Documents/Git/CNTFET-Workflow-simulations/ngspice_simulations/novo

5. Simulator command line:
   /home/debian/.local/ngspice/bin/ngspice freq_3_nocap -b

6. Modelcard file:
   referenceModelcards

7. Modelcard variable:
   n_gfl0w_g4_1n_ta40p_mt0_0001

Choose the parameter you want to change (enter "0" to return to main menu):
```

Figura 4.9: Tela de configuração do CNTFET-Workflow.

```
Distribution type: UT normal
CNTFET name: M1
CNTFET type: N

Parameters selected to be scaled:
n_ta: Mean: 80   Standard deviation: 10   Sigma points: 3

-> Select parameters to be scaled:
1. n_gf   2. w_g   3. l_g   4. l_gsd   5. l_sd   6. d_cnt
7. n_ta   8. p_mt   9. h_mg  10. h_msd  11. h_m1  12. t_ox
13. w_ms  14. w_mg  15. w_do  16. eps_is  17. eps_ox  18. eps_ins

19. Add/edit CNTfet
Enter the parameter index to be scaled (enter "0" to return to main menu):
```

Figura 4.10: Tela de seleção dos parâmetros a serem variados.

A quinta e última opção realiza a simulação propriamente dita. Qualquer outra opção de comando possível está descrita na tela no momento da execução do programa.

Até a data do fim da elaboração deste trabalho o programa CNTFET-Workflow foi utilizado em dois trabalhos de conclusão de curso, por Azevedo [63], cujo trabalho foi o projeto de um misturador de frequências *downconverter* utilizando CNTFETs para um transceptor *ZigBee* e Barbosa [64], que tratou do projeto e estimativa de desempenho e qualidade de um *mixer upconverter* para um transceptor *ZigBee* utilizando CNTFETs.

5

RESULTADOS

Realizar a comparação entre o método de Monte Carlo e a transformada da incerteza em simulações de circuitos elétricos pode ser uma tarefa extremamente ampla devido ao grande número de possibilidades de aplicações destes métodos. Contudo, neste trabalho a comparação será feita utilizando um circuito de oscilador de três e cinco estágios, além de seu subcircuito *buffer/inversor*. Os parâmetros pertinentes ao escopo deste trabalho são a média e o desvio padrão. Possíveis alterações na simetria das distribuições das variáveis aleatórias passíveis de observação são notadas quando conveniente.

Os circuitos, cujas *netlists* foram complementadas com os parâmetros elétricos gerados pelo programa CNTFET-Workflow, foram simulados no Ngspice, um simulador de circuitos elétricos, utilizado principalmente pela sua compatibilidade com o programa ADMS, um gerador de códigos para a linguagem Verilog-AMS, que o converte em código escrito em C, linguagem interpretada pelo Ngspice. Para ser possível realizar uma análise detalhada do considerável volume de dados gerados pelas simulações, os dados de saída gerados foram salvos como variáveis na área de trabalho do GNU Octave, para serem posteriormente analisados.

Basicamente o objeto de análise do trabalho presente é constituído de um oscilador em anel, constituído ora por três ora por cinco estágios. Seu subcircuito, o *buffer/inversor* de lógica em modo corrente (ou simplesmente *buffer* CML) também é objeto de análise. A variação na densidade de nanotubos de carbono, assim como a variação na porcentagem de nanotubos metálicos são as principais variáveis de entrada dos métodos a serem analisados.

5.1 IMPLEMENTAÇÃO DO CCAM NO NGSPICE

A validação da implementação do CCAM no Ngspice faz-se necessária para a sua correta aplicação em posteriores simulações de circuitos elétricos mais complexos. Tal validação pode ser feita a partir da elaboração de curvas de transferência e de saída do modelo do transistor. A curva de transferência do CNTFET é mostrada na Figura 5.1, enquanto que na Figura 5.2 é mostrada a curva de saída. Ambas são do modelo de CNTFET tipo N. As *netlists* utilizadas nestas simulações são descritas nas seções A.1.1 e A.1.2.

Pode-se notar um aumento linear da corrente com um aumento da densidade de nanotubos. No entanto, a saturação do CNTFET não é bem definida, sendo um dos desafios da

tecnologia projetar circuitos com esta característica do transistor. Para fins de comparação, as curvas de transferência e de saída são comparadas com as do trabalho de Cheralathan, Claus e Blawid [3], obtendo-se resultados semelhantes, validando assim a implementação do CCAM tal como sua utilização no simulador Ngspice. Pode-se notar a pequena diferença da corrente I_d , evidenciada na Figura 5.1, da ordem de 10^2 (contra um valor aproximado de 10^6 em um transistor CMOS típico), entre os estados *on/off* ($V_{gs} = 0,4\text{ V}$, $V_{gs} = 0\text{ V}$, respectivamente). Tal fato é justificado pelo valor relativamente pequeno da tensão de limiar V_{th} do transistor.

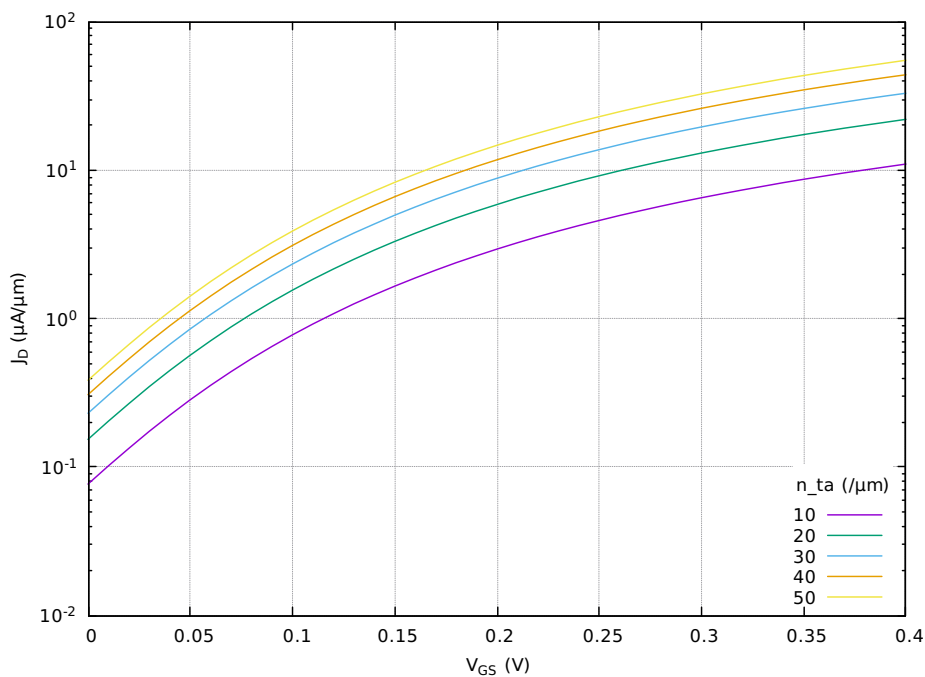


Figura 5.1: Curva de transferência do CNTFET, mostrando a densidade de corrente para diferentes valores de densidade de nanotubos (n_{ta}).

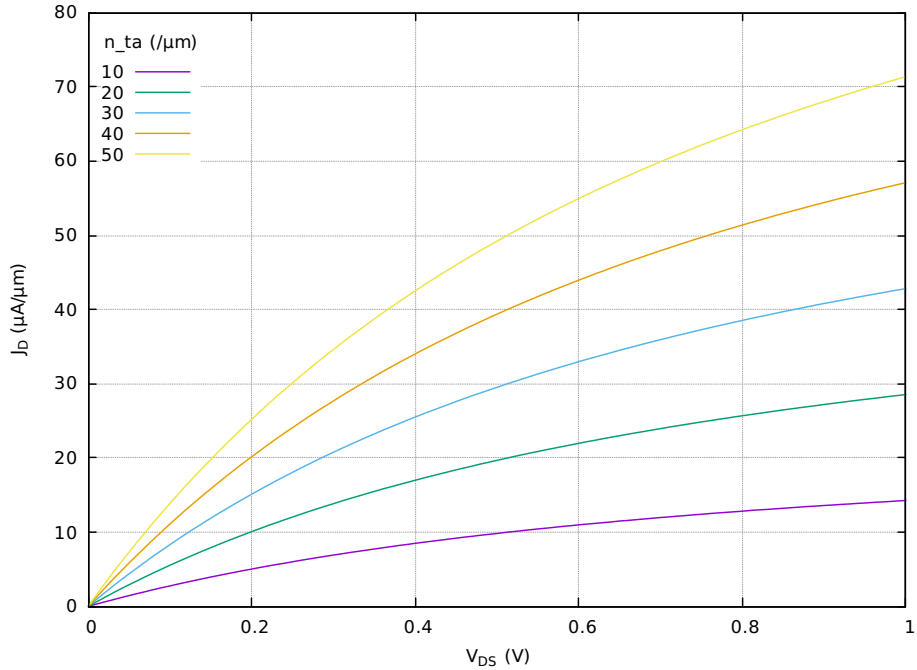


Figura 5.2: Curva de saída do CNTFET, mostrando a densidade de corrente para diferentes valores de densidade de nanotubos (n_{ta}).

5.2 CIRCUITOS UTILIZADOS NA ANÁLISE

5.2.1 Buffer/inversor de lógica em modo corrente

O primeiro circuito objeto de análise é um *buffer* CML, mostrado na Figura 5.3. Mais detalhes sobre este circuito podem ser encontrados em [3]. Os valores utilizados para R_L e C_L são $333\ \Omega$ e $58\ \text{fF}$, respectivamente, e os valores das fontes do circuito são $V_{DD} = 1,8\ \text{V}$ e $I_T = 1,8\ \text{mA}$. Os parâmetros físicos modificados nas análises são a densidade de nanotubos de carbono (n_{ta}) e a porcentagem de nanotubos metálicos (p_{mt}). Mais detalhes sobre estas alterações são mostradas posteriormente. Os outros parâmetros, se não mencionada a mudança de seus valores base, são descritos na Tabela A.2, com seus parâmetros elétricos equivalentes.

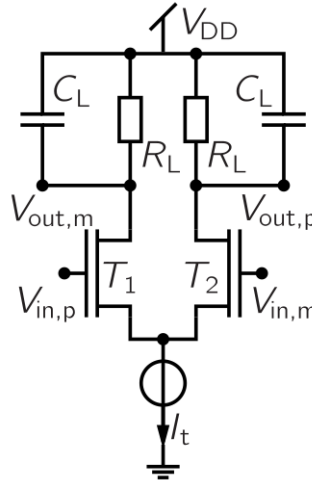


Figura 5.3: Esquemático do circuito *Buffer/inversor* de lógica em modo corrente.

A operação do *buffer* CML é baseada no par diferencial. Havendo um valor constante para a soma das correntes dos transistores T_1 e T_2 , provida pela fonte de corrente I_t , os transistores são alternados entre as regiões de saturação e de corte. A tensão total V_{DD} é distribuída entre os três componentes (R_L , T_x e I_t), cuja excursão da tensão de saída é igual a $\Delta V_{out,i} = I_t \times R_L$.

A principal figura de mérito analisada são as tensões de saída do *buffer* CML ($V_{out,p}$ e $V_{out,m}$), tensões cujos pontos são indicados na Figura 5.3. Com a mudança de tensão nas entradas, há uma mudança de tensão nas saídas do circuito. Relaciona-se então essas mudanças de tensões à média da diferença entre as entradas $V_{in,m}$ e $V_{in,p}$, ou seja, $V_{in,diff} = (V_{in,p} - V_{in,m})/2$. A curva de transferência característica DC é mostrada na Figura 5.4. Os parâmetros físicos modificados são n_{ta} , para $40 \mu\text{m}^{-1}$ nos transistores T_1 e T_2 do circuito.

5.2.2 Osciladores em anel

Os osciladores em anel utilizados nas simulações deste trabalho foram projetados utilizando o *buffer* CML, descrito anteriormente. A Figura 5.5 mostra a topologia básica do oscilador, sendo n o número de inversores no circuito, número necessariamente ímpar.

A partir desta topologia, foram elaborados dois osciladores diferentes: de três e cinco estágios. O objetivo principal da utilização de dois osciladores diferentes é em razão da análise do número de variáveis e a influência das mesmas no funcionamento dos osciladores. O sinal de saída é dado pela diferença de tensões entre as saídas de um determinado estágio. A Figura 5.6 mostra a saída do oscilador em anel de três estágios com todos os parâmetros físicos inalterados.

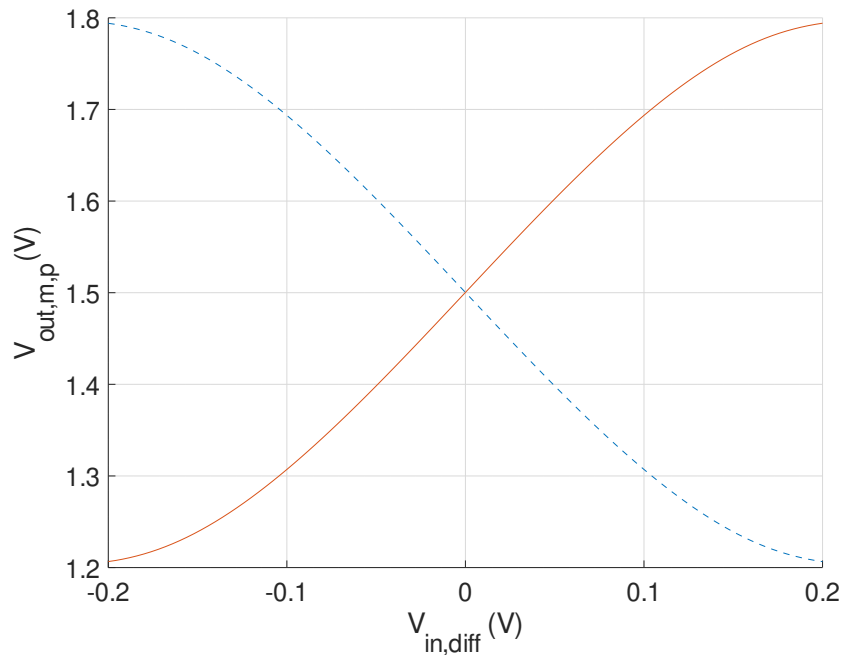


Figura 5.4: Curva de transferência característica DC, $V_{out,m}$ (linha tracejada) e $V_{out,p}$ (linha contínua).

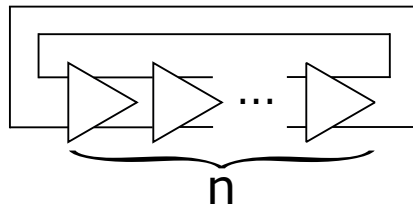


Figura 5.5: Representação de um oscilador em anel de “n” estágios. As saídas de um estágio são ligadas às entradas do estágio seguinte.

5.3 CONVERGÊNCIA DO MÉTODO DE MONTE CARLO

Variações simétricas e não correlatas de qualquer parâmetro do modelo de ambos os CNTFETs têm o potencial de serem uma figura de mérito com valor médio em 0 V, o que pode ser explorado como uma maneira de se verificar a convergência de um método que prevê variabilidade em um circuito elétrico. Assim, para analisar a convergência do método de Monte Carlo, várias séries de simulações dos parâmetros DC foram feitas variando-se a densidade de nanotubos (n_{ta}) de cada CNTFET do *buffer* CML independentemente, tendo assim um total de dois parâmetros físicos usados como variáveis de entrada. Estes parâmetros físicos foram tratados como variáveis aleatórias com distribuição normal, com $\mu_{n_{ta}} = 40 \mu\text{m}^{-1}$ (média) e $\sigma_{n_{ta}} = 6,7 \mu\text{m}^{-1}$ (desvio padrão). A *netlist* utilizada nas simulações desta seção está descrita na seção A.1.3.

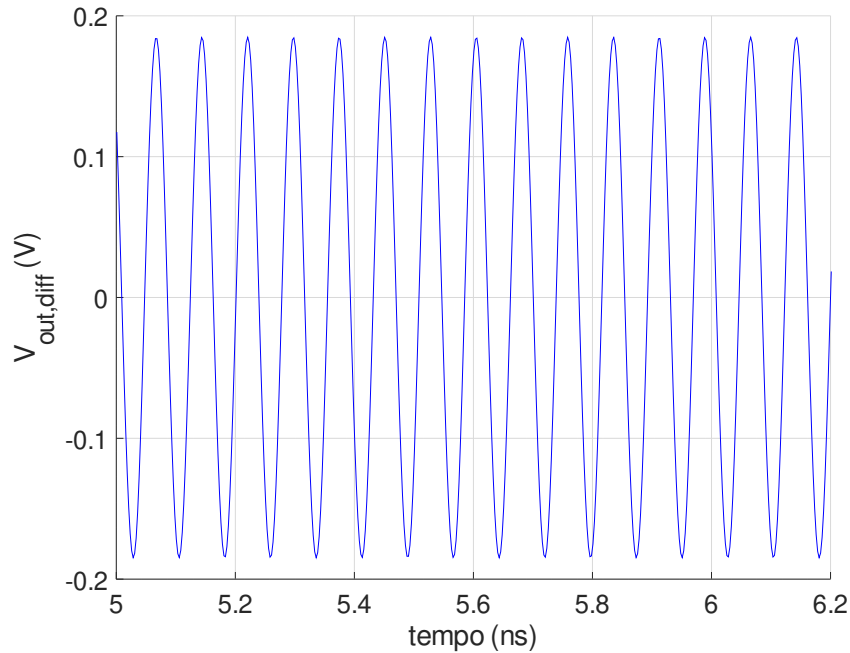


Figura 5.6: Sinal de saída $V_{out,diff}$ do oscilador de três estágios.

A figura de mérito consiste na tensão $V_{out,m} - V_{out,p}$, quando $V_{in,m} = V_{in,p}$, pela sua simplicidade de análise. As séries de simulações consistem em conjuntos de simulações utilizando 10, 50, 100, 500, 1000, 5000 e 10000 passos, com cada conjunto contendo 10 simulações. A média de cada conjunto de simulação é mostrado na Figura 5.7. Além disso, a mesma figura mostra o intervalo de confiança (linha contínua) de três desvios padrões (equivalente a 99,7%).

Comparando-se os valores de cada conjunto de diferentes números de passos, pode-se notar que o conjunto de simulações de 10000 passos têm uma diferença entre os valores máximo e mínimo de $4,7 \times 10^{-4}V$, sendo um número de passos válido para se usar como comparação com os resultados de simulações utilizando a UT.

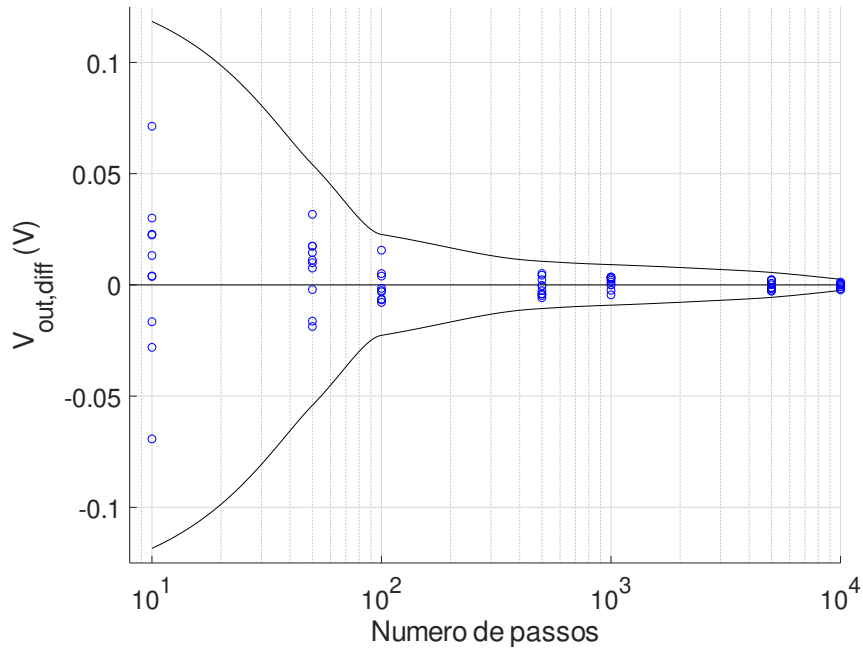


Figura 5.7: Média de cada conjunto de simulações de Monte Carlo

5.4 EFICIÊNCIA COMPUTACIONAL DA UT

Para demonstrar a eficiência computacional da UT foram feitas simulações dos parâmetros DC do circuito CML, cuja figura de mérito escolhida é $V_{out,diff}$ quando $V_{in,m} = V_{in,p}$. A *netlist* utilizada nas simulações desta seção está descrita na seção A.1.3. Para um circuito CML com ambos os transistores com as mesmas características físicas e elétricas, $V_{out,diff} = 0$ V. O número de passos utilizados nas simulações é o número necessário de passos para a simulação utilizando a UT, 9 passos (3^2 , três para cada variável de entrada). Tal escolha para o número de passos na simulação utilizando o método de Monte Carlo é justificada pela ideia de se analisar a suposta vantagem na eficiência da UT em relação ao método citado anteriormente. A densidade de nanotubos de carbono de cada um dos dois transistores do CML foi modelada de forma independente, ou seja, como duas variáveis diferentes, de acordo com uma função de densidade de probabilidade normal de $\mu_{n_{ta}} = 40 \mu\text{m}^{-1}$ e $\sigma_{n_{ta}} = 10 \mu\text{m}^{-1}$.

A Figura 5.8 mostra a modelagem das variáveis de entrada da simulação de Monte Carlo (a) e utilizando a UT (b). Com a combinação de parâmetros gerados a partir da modelagem descrita anteriormente da densidade de nanotubos em cada transistor, são realizadas os passos de cada tipo de simulação, resultando nos valores mostrados nos gráficos (c) (Monte

Carlo) e (d) (UT) da mesma figura. Pode-se notar uma simetria nos passos da simulação UT, além da repetição de um padrão nos resultados. A partir dos resultados obtidos calculou-se a média e desvio padrão de cada simulação, obtendo funções de densidade de probabilidade ilustradas no gráfico (e) da mesma figura supracitada. Com valores de média e desvio padrão iguais a $\mu_{mc} = 0,0416 \text{ V}$ e $\sigma_{mc} = 0,0911 \text{ V}$ para a simulação utilizando o método de Monte Carlo e média e desvio padrão iguais a $\mu_{ut} = 1,7 \times 10^{-18} \text{ V}$ e $\sigma_{ut} = 0,0976 \text{ V}$ para a simulação utilizando a UT, nota-se valores mais perto do esperado utilizando a seleção determinística dos pontos sigma e o cálculo de seus momentos. Vale lembrar que a simulação utilizando o método da UT, neste caso onde a simulação utilizando o método de Monte Carlo possui poucos passos para ter uma razoável ordem de convergência, é utilizada como referência para os valores dos momentos.

A amostragem aleatória das variáveis de entrada no método de Monte Carlo leva a combinações redundantes de variáveis, e assim a previsões imprecisas de variabilidade. A UT requer um mínimo de três pontos sigma para cada variável de entrada e todas as suas combinações possíveis (3^2 , como dito anteriormente) para prever os dois primeiros momentos (média e desvio padrão) da distribuição da figura de mérito em questão com uma precisão maior que o método de Monte Carlo, reduzindo o esforço computacional exigido.

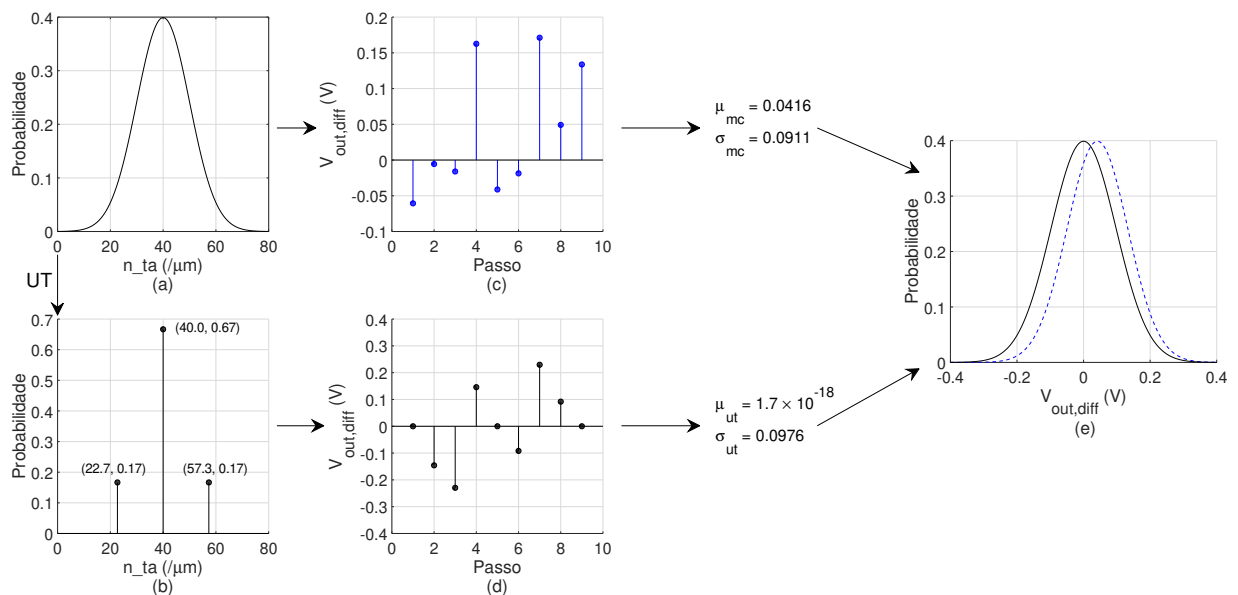


Figura 5.8: Gráficos das variáveis de entrada das simulações utilizando o método de Monte Carlo (a) e a UT (b). Os gráficos (c) e (d) mostram os passos das simulações de Monte Carlo e UT, respectivamente. O gráfico (e) mostra a função densidade de probabilidade das saídas das simulações de Monte Carlo (linha tracejada) e UT (linha contínua).

É importante destacar que os valores resultantes nas simulações de Monte Carlo geram sempre valores aleatórios, como mostrada no gráfico (c) da Figura 5.8. Contudo, os valo-

res gerados pela UT são notadamente determinísticos (gráfico (d) da mesma figura), fato justificado pela natureza da transformada em questão. Nota-se ainda a simetria dos valores resultantes utilizando a transformada, mostrados no mesmo gráfico citado anteriormente, possuindo um total de quatro valores com módulos iguais para diferentes passos. Como consequência, é possível realizar a simulação executando apenas três passos, fazendo um arranjo com os valores (considerando a simetria entre os dois transistores do circuito: passos 2 e 4, 3 e 7, 6 e 8) de modo a atingir o resultado em todos os nove passos. O valor restante é igual a zero, para densidades de nanotubos iguais entre os transistores do circuito, fato presente nos passos 1, 5 e 9.

A Figura 5.9 mostra os resultados de oito simulações utilizando o método de Monte Carlo com nove passos cada, com os mesmos parâmetros da simulação MC ilustrada na Figura 5.8. Em cada gráfico estão descritos os valores da média e do desvio padrão de cada simulação. Pode-se notar a relativa variação destes momentos entre cada simulação, considerando a simulação da UT como referência. Considerando que o esforço computacional entre os dois métodos tenha uma diferença irrisória, em se tratando de simulações com poucas variáveis, há consequentemente uma nítida vantagem da UT sobre o método de Monte Carlo sob estas condições.

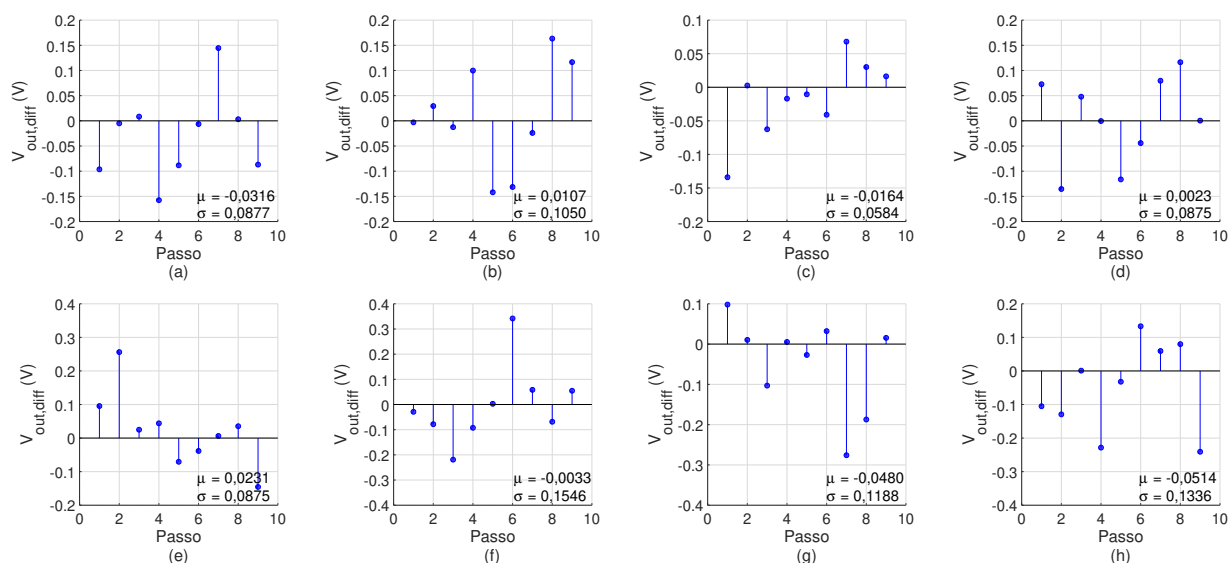


Figura 5.9: Simulações utilizando o método de Monte Carlo, com nove passos cada, com os mesmos parâmetros da simulação MC da Figura 5.8. A média e o desvio padrão de cada simulação estão escritos nos respectivos gráficos.

5.5 ESTUDO DE CASO: OSCILADORES EM ANEL

Osciladores em anel são amplamente utilizados em projetos de circuitos integrados, especialmente em se tratando de circuitos RF. Como a variação de parâmetros físicos dos transistores utilizados na sua fabricação podem levar a variações na frequência do sinal de saída e até mesmo ao não funcionamento do circuito, faz-se necessária a análise da variabilidade neste tipo de circuito. Os circuitos simulados nesta seção foram construídos utilizando o circuito *buffer* CML, descrito em seções anteriores. Mais informações sobre o projeto destes circuitos pode ser encontrado em [3].

Como elemento de análise foi escolhida a densidade dos nanotubos (n_{ta}) dos transistores, consideradas variáveis aleatórias não correlatas (densidade individual de cada transistor) com distribuição normal de $\mu_{n_{ta}} = 80 \mu\text{m}^{-1}$ e $\sigma_{n_{ta}} = 10 \mu\text{m}^{-1}$. A UT aumenta a previsão do erro de maneira controlada de acordo com o número de pontos sigma utilizados, servindo assim como referência. A figura de mérito dos osciladores é a frequência de oscilação f_{osc} . As *netlists* utilizadas nas simulações desta seção estão descritas na seção A.1.4 (oscilador de três estágios) e A.1.5 (oscilador de cinco estágios).

A Figura 5.10 ilustra os parâmetros físicos modelados como variáveis aleatórias, utilizados nas simulações dos osciladores. A Tabela 5.1 mostra as médias e desvios-padrões das simulações MC e UT, mostradas na Figura 5.10, dos osciladores de três e cinco estágios.

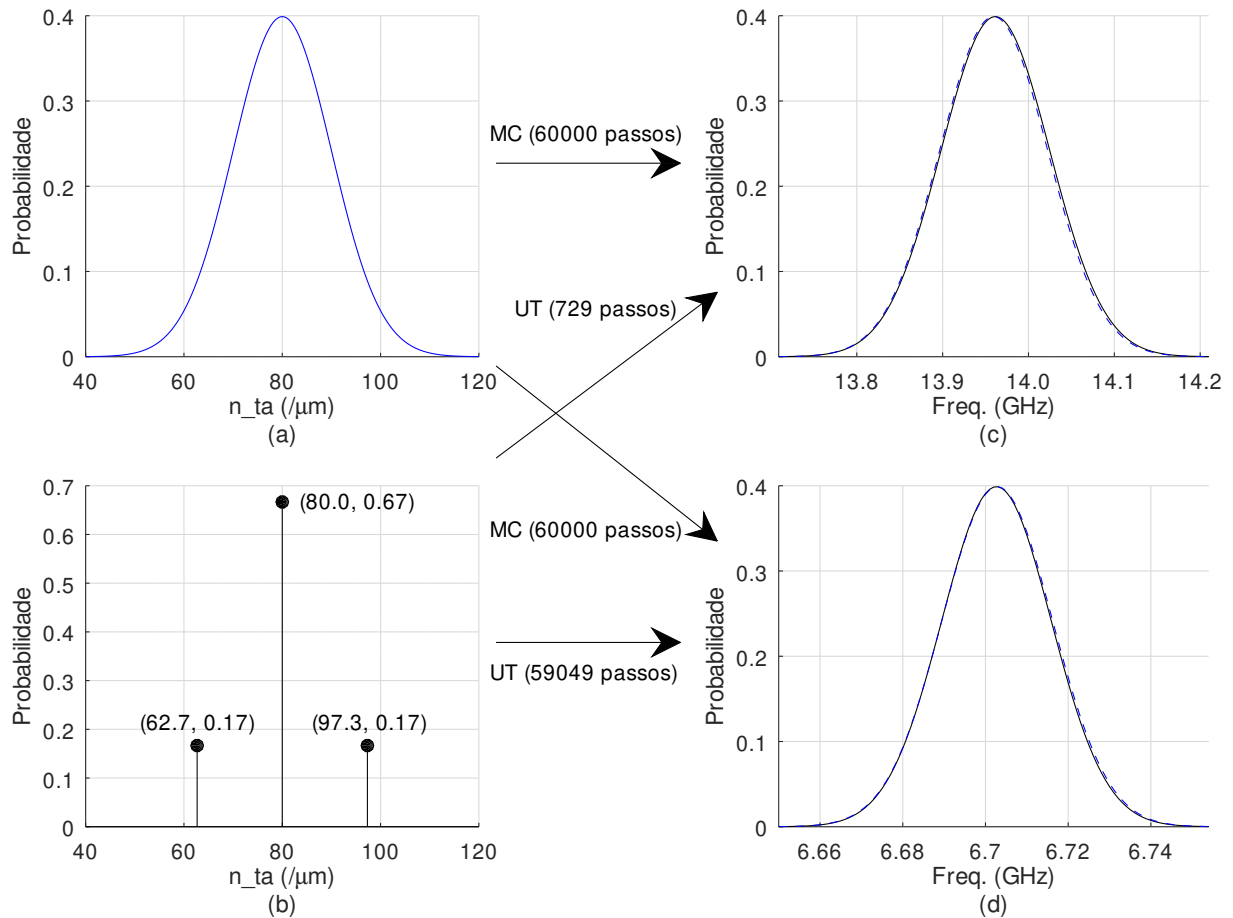


Figura 5.10: Gráficos das variáveis de entrada (n_{ta}) das simulações utilizando o método de Monte Carlo (a) e a UT (b). As frequências das simulações de Monte Carlo (linha contínua) e utilizando a UT (linha tracejada) do oscilador de três estágios são mostradas no gráfico (c), enquanto que as frequências da simulação do oscilador de cinco estágios são mostradas no gráfico (d).

Por um lado, ambos os métodos computacionais geram previsões de rendimento equivalentes, porém com diferentes níveis de esforço computacional. Os osciladores em anel analisados são compostos de três e cinco *buffers* CML, e conseqüentemente de seis e dez CNTFETs, respectivamente. Assim, seis ou dez variáveis podem ser variadas de forma independente para o caso presente. Prever a variabilidade dos dois osciladores através do método da UT requer simulações com 729 (3^6) e 59049 (3^{10}) passos (com diferentes combinações de valores para as variáveis de cada transistor) respectivamente. Por outro lado, a análise do *buffer* CML mostrou que simulações da ordem de dezenas de milhares são necessárias para se prever a variabilidade do circuito corretamente, como mostrado na Figura 5.7. Como conseqüência, foi escolhido um total de 60000 passos para a compor as simulações utilizando o método de Monte Carlo, sendo notadamente similar ao número de passos derivados da combinação de pontos sigma necessários para a realização da simulação do oscilador de

Tipo de simulação	Estágios	μ_f/GHz	σ_f/GHz	Número de passos	Valores diferentes
MC	3	13,7594	0,0629	60000	123
UT	3	13,7614	0,0633	729	67
MC	5	6,6428	0,0134	60000	57
UT	5	6,6426	0,0132	59049	58
MC	3	13,7603	0,0569	100	47
MC	3	13,7578	0,0651	1000	90
MC	3	13,7599	0,0628	10000	107
MC	5	6,6444	0,0125	100	26
MC	5	6,6428	0,0130	1000	37
MC	5	6,6428	0,0133	10000	53

Tabela 5.1: Valores listados de média e desvio padrão de cada simulação realizada para esta seção.

cinco estágios utilizando a UT.

O número de passos requeridos em cada tipo de simulação mostra um aspecto diferente entre os métodos MC e UT. Por um lado, o método de Monte Carlo permite realizar o *tradeoff* entre esforço computacional e precisão, escolhendo o melhor benefício levando em consideração as condições de análise. Em contrapartida, o método da UT é uma aproximação sistemática que requer um número definido de passos de acordo com o número de variáveis e pontos sigma de cada uma das mesmas, tendo uma precisão constante, definida pelo número de pontos sigma de cada variável. Tal fato aumenta de forma drástica o esforço computacional para prever os momentos de variáveis de um determinado sistema.

Levando em conta a utilização de uma resolução de 4 MHz para o oscilador de três estágios e 2 MHz para o de cinco estágios, resultantes da utilização de um período de amostragem de 250 ns e 500 ns, respectivamente, um total de 67 valores diferentes para o oscilador de três estágios e 58 valores para o de cinco foram encontrados realizando uma transformada rápida de Fourier nos dados das simulações UT de cada oscilador, com uma frequência de amostragem de 5 ps. A Tabela 5.1 mostra ainda os valores para simulações MC de ambos os osciladores com cem, mil e dez mil passos. Pode-se notar o aumento do número de valores diferentes para os resultados, levando-se em consideração as resoluções supracitadas. Considerando as frequências resultantes da simulação de cada oscilador, seria aplicável utilizar uma resolução ainda menor para o oscilador de cinco estágios. Porém, pelo custo computacional, utilizou-se o tempo descrito. Como esperado, a redundância decresce à medida em que a frequência de amostragem aumenta.

É possível reduzir o número de passos simulados para algumas centenas para o oscilador de três estágios e cinco estágios, analisando-se a configuração da combinação dos pontos sigma (e conseqüentemente dos parâmetros físicos) adotados nos transistores. A exploração

da simetria do oscilador em anel leva a uma redução do número de passos executados para se completar a simulação. A Figura 5.11 mostra os histogramas de cada simulação da presente seção. Pode-se notar uma evidente assimetria no resultado da simulação MC do oscilador de cinco estágios. Este é um fato importante para o DFM em níveis maiores de abstração, pois as figuras de mérito são geralmente não normalmente distribuídas. No entanto, a análise de tal fato não pertence ao escopo deste trabalho, feita a partir de momentos de ordens superiores da distribuição em questão. A forma dos histogramas das simulações UT não possuem fatos relevantes, embora guardem informações acerca dos momentos das variáveis aleatórias.

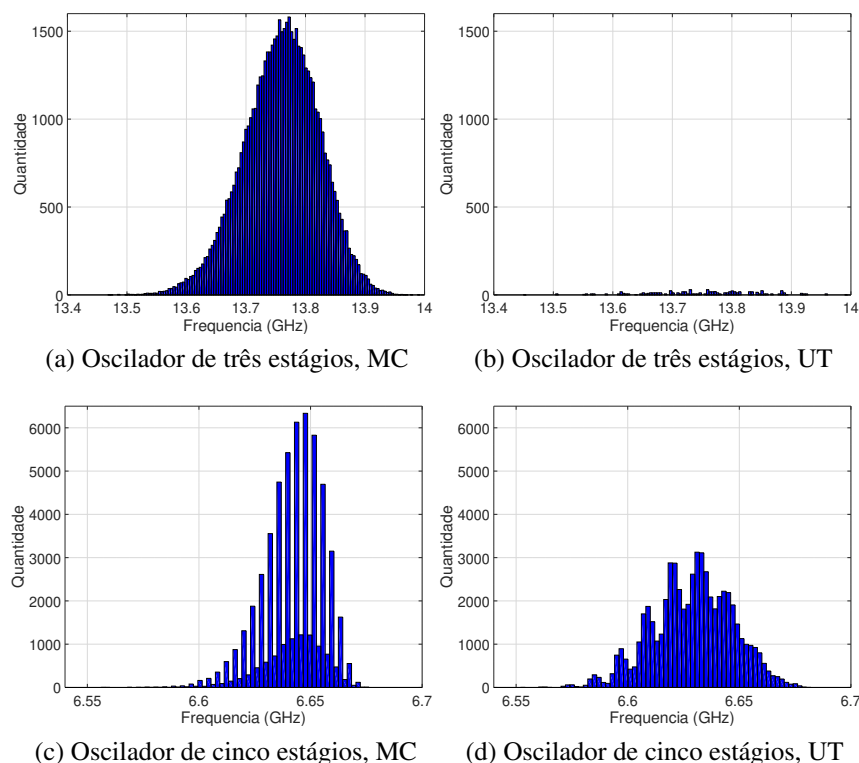


Figura 5.11: Histogramas das simulações MC e UT dos osciladores de três e cinco estágios.

Utilizando os resultados obtidos na simulação MC pode-se exemplificar um rendimento paramétrico dos osciladores a partir da variação da densidade de nanotubos. Considerando uma variação de 78 MHz (faixa de operação do *bluetooth*) faz-se necessário determinar o rendimento de ambos os osciladores para uma faixa de 13,720 GHz a 13,798 GHz para o oscilador de três estágios e 6,604 GHz a 6,682 GHz para o de cinco estágios. De acordo com os momentos calculados a partir das simulações supracitadas, o rendimento dos osciladores é de 46,48% e 99,64%, respectivamente. Pode-se notar uma influência maior da variação da densidade de nanotubos dos transistores na frequência de operação do oscilador de três estágios. Tal fato é justificado pelo oscilador possuir menos variáveis, neste caso a densidade de nanotubos de cada um dos transistores. Assim, cada variável possui uma influência maior

na frequência de funcionamento do oscilador.

Pode-se citar outro exemplo de simulação utilizando o oscilador de 5 estágios como circuito a ser analisado, onde a área de rendimento é composta por uma variação de 20 MHz a partir de determinada frequência central. Partindo da variação da densidade de nanotubos em cada transistor, de forma independente, com média de $40 \mu\text{m}^{-1}$ e desvio padrão de $1 \mu\text{m}^{-1}$, chegou-se à figura de mérito da frequência de saída do oscilador com uma média de 6,164 GHz e desvio padrão de 8,770 MHz. Por fim, obteve-se um rendimento paramétrico de aproximadamente 74,58%.

Como esperado, o método utilizando a UT reduz o esforço computacional consideravelmente em comparação com MC para circuitos menores, como o oscilador de três estágios. Para o oscilador de cinco estágios, o método de Monte Carlo se mostra tão eficiente quanto a UT, pois a mesma necessita um número de passos que aumenta exponencialmente com o número de variáveis. Contudo, a topologia do circuito de osciladores em anel é altamente simétrica. Assim, a simulação de todos os passos oriundos das possíveis combinações de pontos sigma contém uma quantidade notável de redundância. Há a possibilidade de se simplificar o método de Monte Carlo, sorteando valores para uma variável aleatória de acordo com uma determinada distribuição normal, porém com valores pré-calculados de acordo com a UT, tal ideia se assemelhando ao método de quasi-Monte Carlo.

5.6 UTILIZAÇÃO DE PARÂMETROS COM DISTRIBUIÇÃO NÃO-NORMAL NA UT

Até agora foram utilizadas variáveis de entrada modeladas com funções de densidade de probabilidade estritamente normais. Contudo, a distribuição normal nem sempre é a distribuição mais adequada para se modelar determinado parâmetro. Um exemplo cabível nessa situação é a distribuição da porcentagem de nanotubos metálicos (p_{mt}) no CNTFET. Como um parâmetro físico, tal valor não pode ser menor que zero. Como consequência, considerando a variável com distribuição normal com um valor suficientemente pequeno para a média e um valor suficientemente grande para o desvio padrão, tal variável terá um truncamento em sua distribuição. Para analisar este efeito, foram realizadas simulações com dez mil passos utilizando o circuito CML, elegendo a variável de saída $V_{out,diff}$ como figura de mérito. A justificativa do número de passos utilizados nas simulações MC é dada na seção 5.3. Como variável de entrada, foi definida a porcentagem de nanotubos metálicos de ambos os transistores como variáveis independentes, como três distribuições diferentes, modeladas como distribuições normais com média $\mu_{p_{mt}} = 0,6\%$ e desvio padrão $\sigma_{p_{mt}} = 0,1\%$,

$\sigma_{p_mt} = 0,2\%$ e $\sigma_{p_mt} = 0,3\%$, a última com truncamento evidenciado. A *netlist* utilizada nas simulações desta seção está descrita na seção A.1.3.

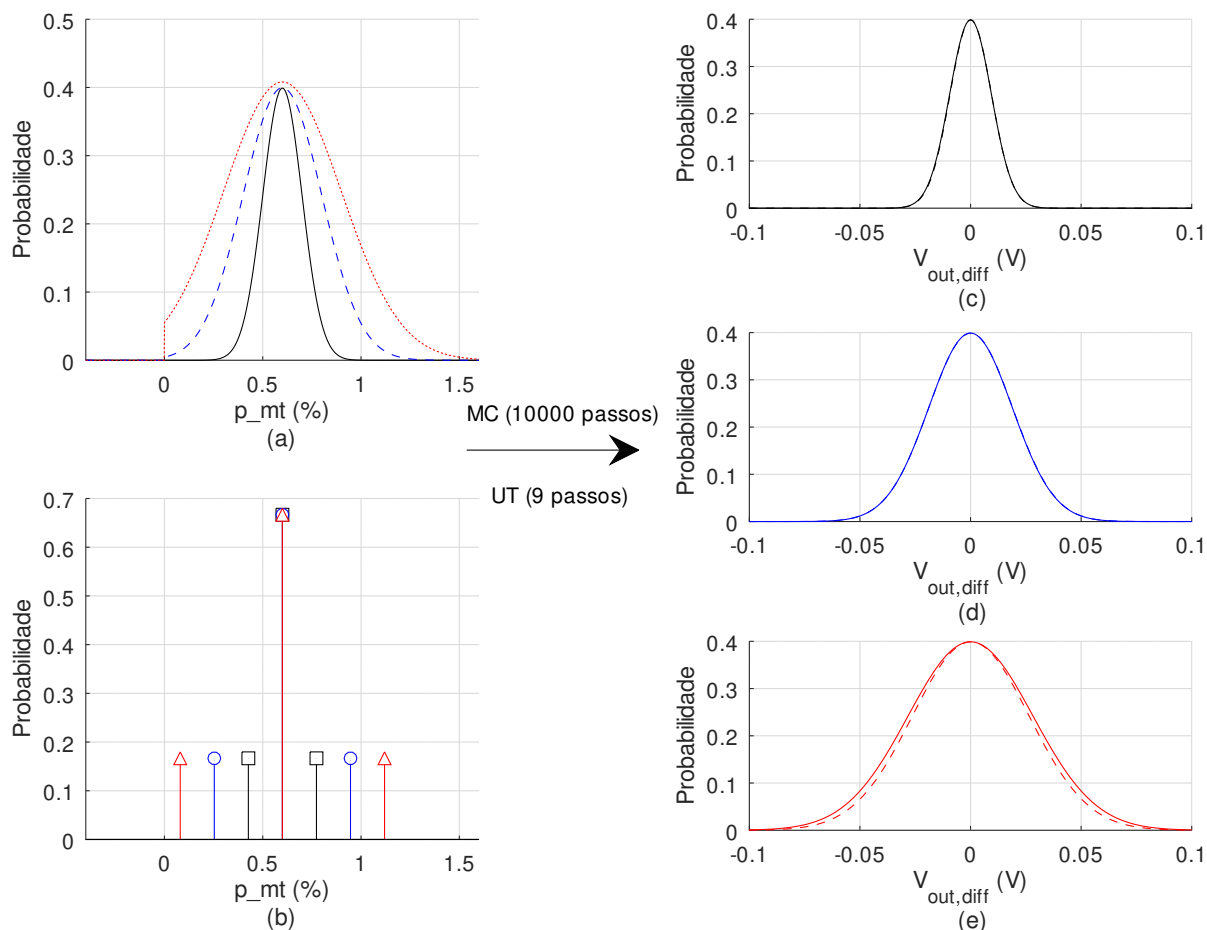


Figura 5.12: Variáveis de entrada, (a) para MC e (b) para UT e os resultados das simulações utilizando estas variáveis, (c) para a variável modelada com distribuição normal linha contínua, com $\mu_{p_mt} = 0,6\%$ e $\sigma_{p_mt} = 0,1\%$ (linha contínua), (d) para a variável com $\mu_{p_mt} = 0,6\%$ e desvio padrão $\sigma_{p_mt} = 0,4\%$ (linha tracejada) e (e) com $\mu_{p_mt} = 0,6\%$ e desvio padrão $\sigma_{p_mt} = 0,3\%$ (linha pontilhada).

A Figura 5.12 mostra as variáveis de entrada para o método de Monte Carlo (a) e para a UT (b). Nota-se que a variável com $\mu_{p_mt} = 0,6\%$ e $\sigma_{p_mt} = 0,3\%$ (linha pontilhada) possui um truncamento para valores menores que zero, ou seja, a função é definida como igual a zero para valores da porcentagem menores que zero. Como consequência, pode-se notar uma diferença nos valores da distribuição da figura de mérito $V_{out,diff}$, gráfico (e) da mesma figura, onde o desvio padrão possui um valor menor (linha tracejada), enquanto que para as simulações cujas variáveis de entrada que possuem desvios padrões iguais a $\sigma_{p_mt} = 0,1\%$ e $\sigma_{p_mt} = 0,2\%$ não possuem esta diferença, onde as linhas tracejadas, gráficos (c) e (d) da mesma figura, que indicam os resultados das simulações utilizando a UT, estão sobrepondo

as linhas contínuas, que representam as figuras de mérito das simulações de MC. A Tabela 5.2 contém os valores dos pontos sigma calculados para as simulações UT, tal como a média e o desvio padrão da figura de mérito (FoM) de cada simulação.

Densidade de nanotubos metálicos (p_{mt})	μ_{FoM}/V	σ_{FoM}/V
$\mu_{p_{mt}} = 0,6\%$, $\sigma_{p_{mt}} = 0,1\%$ (MC)	$1,103 \times 10^{-4}$	$9,347 \times 10^{-3}$
(0.43, 0.17), (0.6, 0.67), (0.78, 017) (UT)	0	$9,418 \times 10^{-3}$
$\mu_{p_{mt}} = 0,6\%$, $\sigma_{p_{mt}} = 0,2\%$ (MC)	$7,305 \times 10^{-5}$	$1,886 \times 10^{-2}$
(0.25, 0.17), (0.6, 0.67), (0.95, 017) (UT)	0	$1,886 \times 10^{-2}$
$\mu_{p_{mt}} = 0,6\%$, $\sigma_{p_{mt}} = 0,3\%$ (MC)	$1,062 \times 10^{-4}$	$2,640 \times 10^{-2}$
(0.08, 0.17), (0.6, 0.67), (1.12, 017) (UT)	0	$2,825 \times 10^{-2}$

Tabela 5.2: Parâmetros das distribuições normais das variáveis aleatórias de entrada e da figura de mérito $V_{out,diff}$.

Utilizar a UT como alternativa ao método de Monte Carlo possui certas restrições no que tange às distribuições das variáveis de entrada. Quando utilizada substituindo uma função de densidade de probabilidade truncada, os pontos sigma que se localizam nos extremos da distribuição normal podem não estar dentro do domínio da distribuição da variável aleatória em questão. No entanto, a transformada da incerteza pode ser utilizada em distribuições não-normais, desde que utilizadas técnicas específicas nos cálculos dos pontos sigma. A título de exemplo das simulações realizadas para esta seção, a variabilidade do circuito não pode ser prevista para uma distribuição normal truncada da porcentagem de nanotubos metálicos (p_{mt}) com $\mu_{p_{mt}} = 0,6\%$ e $\sigma_{p_{mt}} = 0,4\%$. Além disso, os pontos sigma são selecionados simetricamente. Como consequência, erros na variabilidade do circuito devem ser previstos quando a assimetria da distribuição da variável aleatória de entrada for notadamente relevante.

6

CONCLUSÃO

O presente trabalho uniu duas frentes de pesquisa que, de acordo com os estudos realizados, possuem características que essa união pode gerar benefícios ainda maiores. Por um lado, o transistor de efeito de campo de nanotubos de carbono, possuidor de certas características que permitem a sua aplicação similares à tecnologia CMOS, tendo as aplicações de RF um maior potencial de impactar a indústria de semicondutores. Por outro lado, a transformada da incerteza, possuidora de uma série de vantagens sobre o método de Monte Carlo em determinadas aplicações, como por exemplo em simulações de determinados circuitos elétricos, como visto neste trabalho. Sua maior vantagem é a de requerer um esforço computacional algumas ordens de grandeza menores que o método de Monte Carlo, sob determinadas condições, como por exemplo na utilização de poucas variáveis aleatórias como variáveis de entrada do sistema, de modo que o aumento exponencial do esforço computacional não seja desvantajoso. Contudo, há uma série de restrições em sua utilização, fato que pode ser estudado e possivelmente mitigado em trabalhos futuros. Unindo estas duas frentes de trabalho é possível analisar a variabilidade de vários aspectos físicos do transistor e de seu impacto nos circuitos elétricos cuja topologia o CNTFET faz parte, como por exemplo osciladores em anel, utilizados no *benchmarking* de atraso de transistores.

Como mostrado no capítulo 1, a tecnologia CNTFET é relativamente nova, necessitando de mais pesquisas no que tange à sua fabricação e caracterização, cujo modelo é utilizado no projeto de novos circuitos elétricos em variadas aplicações. Desafios em sua fabricação ainda precisam ser superados para se atingir um nível de qualidade e rendimento pertinentes à sua aplicação na indústria de semicondutores. Dentre estes desafios pode-se citar um controle maior da porcentagem de nanotubos metálicos conectados entre fonte e dreno durante a fabricação do transistor, nanotubos cuja característica de condução prejudica o controle da corrente do transistor, e conseqüentemente o funcionamento do mesmo. Além disso, outro desafio importante é o aumento da densidade de nanotubos no transistor. Aumentando-se a densidade, aumenta-se a corrente e a transcondutância, características importantes para garantir um maior ganho e aumentar a gama de potenciais aplicações do CNTFET.

Nos capítulos 2 e 3 foram introduzidos alguns aspectos técnicos relacionados ao trabalho, tais como o funcionamento básico do CNTFET, a modelagem do mesmo e algumas de suas características inerentes ao método de fabricação utilizado. Além disso, uma introdução ao método de Monte Carlo e a formulação utilizada para o cálculo dos pontos sigma na transformada da incerteza foram apresentadas.

Para realizar a aplicação, análise e comparação de ambos os métodos em simulações de circuitos elétricos baseados na tecnologia CNTFET, foi desenvolvido um programa em ambiente GNU Octave chamado CNTFET-Workflow, descrito no capítulo 4. Este programa permite aplicar os métodos de simulação tanto de Monte Carlo quanto da transformada da incerteza em parâmetros físicos do CNTFET, tratando-os como variáveis aleatórias. Assim, foram selecionados três circuitos para serem objetos de análise: um *buffer* CML, um oscilador de três e outro de cinco estágios. No *buffer* CML a figura de mérito analisada foi a tensão de saída, com a densidade de nanotubos de carbono sendo o parâmetro modelado como variável aleatória. Esta análise teve o objetivo de relatar a eficiência computacional da UT. Com o número de passos realizados definido pelo número necessário para se realizar a simulação utilizando a UT, a mesma mostrou uma precisão relativamente maior que o método de Monte Carlo, pois este necessita de um número maior de passos para atingir a convergência definida. A segunda análise mostrou o fato do número de passos da UT aumentar exponencialmente com o número de variável aleatórias, utilizando os circuitos dos osciladores de três (729 passos) e cinco estágios (59049 passos), tendo a frequência do oscilador como figura de mérito e modelando mais uma vez a densidade de nanotubos dos transistores como variáveis aleatórias para tal. Por fim, utilizando novamente o circuito do *buffer* CML e a porcentagem de nanotubos metálicos como variáveis aleatórias, foi mostrado o impacto nos resultados das simulações da utilização de distribuições normais truncadas, fato que altera o resultado das simulações em questão. Para contornar este fato, faz-se necessária outras técnicas para o cálculo dos pontos sigma.

6.1 PERSPECTIVAS FUTURAS

Há uma série de potenciais trabalhos futuros acerca dos estudos apresentados nesta dissertação. Pode-se citar como os mais importantes, mas não limitados a, os seguintes itens:

- Otimização do código do CNTFET-Workflow: o código, por ser feito especialmente para este trabalho, possui capacidade de otimização, além de expansão para outros métodos de simulação (*e.g.* método quasi-Monte Carlo), além de outros modelos de funções de densidade de probabilidade;
- Utilização de outros modelos do CNTFET além do CCAM, em especial modelos baseados em tabelas;
- Modelagem de outras tecnologias além do CNTFET;
- Execução das simulações a partir de arquivos contendo os parâmetros da simulação,

evitando assim erros durante a configuração da simulação. Como consequência, é gerada uma documentação do histórico das simulações realizadas;

- Implementação de métodos mais eficientes dos resultados, do ponto de vista de uso da memória;
- Conversão do código para uma versão *stand-alone*, sem a necessidade da utilização do GNU-Octave;
- Desenvolvimento de algoritmo aplicável à UT para a utilização de distribuições não-normais;
- Implementação de amostras utilizando o método de quasi-Monte Carlo preselecionadas pela UT;
- Implementação de técnicas que evitem o problema do escalonamento exponencial;
- Análise da redundância em circuitos de *benchmark*: Listagem de circuitos de *benchmark* importantes e suas simetrias, além da análise computacional da simetria dos circuitos;
- Inverter o problema da variabilidade com o auxílio da UT: da especificação de projeto (rendimento paramétrico) para a variabilidade desejada no processo de fabricação;
- Utilizar a UT em um espaço de parâmetro multidimensional.

Referências

- 1 HARTMANN, J. Towards a new nanoelectronic cosmology. In: IEEE. *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*. [S.l.], 2007. p. 31–37.
- 2 SCHROTER, M. H. M.; CLAUS, M. *CCAM Compact Carbon Nanotube Field-Effect Transistor Model*. 2015. Data de acesso: 21/03/2019. Disponível em: <<https://nanohub.org/publications/62/2>>.
- 3 JULIER, S. J.; UHLMANN, J. K. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, IEEE, v. 92, n. 3, p. 401–422, 2004.
- 4 STRUCTURE of Carbon Nanotubes. Disponível em: <<http://phycomp.technion.ac.il/~talimu/structure.html>>.
- 5 CHIRALITY of single-walled carbon nanotubes | Research | Kato Group at RIKEN. Disponível em: <<http://katogroup.riken.jp/en/nanotubechirality.html>>.
- 6 FRANKLIN, A. D. et al. Sub-10 nm carbon nanotube transistor. *Nano letters*, ACS Publications, v. 12, n. 2, p. 758–762, 2012.
- 7 PULFREY, D. L. *Understanding modern transistors and diodes*. [S.l.]: Cambridge University Press, 2010.
- 8 CNT wrap-gate transistors could extend transistor performance scaling. Data de acesso: 21/03/2019. Disponível em: <<https://phys.org/news/2013-05-cnt-wrap-gate-transistors-transistor-scaling.html>>.
- 9 CAO, Y. et al. Design sensitivities to variability: Extrapolations and assessments in nanometer VLSI. In: UNIVERSITY OF CALIFORNIA, BERKELEY, BERKELEY, UNITED STATES. *Proceedings of the Annual IEEE International ASIC Conference and Exhibit*. [S.l.]: IEEE, 2002. p. 411–415.
- 10 KOREN, I.; KOREN, Z. Defect tolerance in VLSI circuits: techniques and yield analysis. *Proc. IEEE*, v. 86, n. 9, p. 1819–1838, 1998.
- 11 CHERALATHAN, M.; CLAUS, M.; BLAWID, S. Projected tolerances of carbon nanotube current-mode logic to process variability. *IEEE Transactions on Circuits and Systems II: Express Briefs*, v. 65, n. 6, p. 704–708, June 2018. ISSN 1549-7747.
- 12 ABU-RAHMA, M. H.; ANIS, M. *Nanometer Variation-Tolerant SRAM: Circuits and Statistical Design for Yield*. [S.l.]: Springer, 2013.
- 13 DIRECTOR, S. W.; MALY, W.; STROJWAS, A. J. *VLSI design for manufacturing: yield enhancement*. [S.l.]: Springer Science & Business Media, 2012. v. 86.

- 14 RAINA, R. What is dfm & dfy and why should i care? In: IEEE. *2006 IEEE International Test Conference*. [S.l.], 2006. p. 1–9.
- 15 BÜHLER, M. et al. Dfm/dfy design for manufacturability and yield-influence of process variations in digital, analog and mixed-signal circuit design. In: EUROPEAN DESIGN AND AUTOMATION ASSOCIATION. *Proceedings of the conference on Design, automation and test in Europe: Proceedings*. [S.l.], 2006. p. 387–392.
- 16 CHIANG, C.; KAWA, J. *Design for manufacturability and yield for nano-scale CMOS*. [S.l.]: Springer Science & Business Media, 2007.
- 17 THOMPSON, R. B.; THOMPSON, B. F. *PC hardware in a nutshell: a desktop quick reference*. [S.l.]: " O'Reilly Media, Inc.", 2003.
- 18 INTERNATIONAL Roadmap for Devices and Systems (IRDS™) 2017 Edition: Beyond CMOS. Disponível em: <https://irds.ieee.org/images/files/pdf/2017/2017IRDS_BC.pdf>.
- 19 WILLIAMS, R. S. What's next? [the end of moore's law]. *Computing in Science Engineering*, v. 19, n. 2, p. 7–13, March 2017. ISSN 1521-9615.
- 20 INTERNATIONAL Roadmap for Devices and Systems (IRDS™) 2017 Edition: Emerging Research Materials. Disponível em: <https://irds.ieee.org/images/files/pdf/2017/2017IRDS_ERM.pdf>.
- 21 LAIRD, E. A. et al. Quantum transport in carbon nanotubes. *Reviews of Modern Physics*, APS, v. 87, n. 3, p. 703, 2015.
- 22 AHMAD, H. M. N. et al. Design and analysis of a cntfet-based potentiostat for bio medical sensing application. In: *2014 International Conference on Advances in Engineering Technology Research (ICAETR - 2014)*. [S.l.: s.n.], 2014. p. 1–4. ISSN 2347-9337.
- 23 PURI, A.; RANA, A. Performance analysis of cntfet based low power operational amplifier in analog circuits for biomedical applications. In: *2015 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. [S.l.: s.n.], 2015. p. 1–5.
- 24 SCHROTER, M. et al. Carbon nanotube fet technology for radio-frequency electronics: State-of-the-art overview. *IEEE Journal of the Electron Devices Society*, IEEE, v. 1, n. 1, p. 9–20, 2013.
- 25 FARHANA, S.; ALAM, A. Z.; KHAN, S. Cntfet inverter: A high voltage gain logic gate. In: *2014 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*. [S.l.: s.n.], 2014. p. 1–4.
- 26 BENOVA, D. M. The manhattan project, the first electronic computer and the monte carlo method. *Monte Carlo Methods and Applications*, De Gruyter, v. 22, n. 1, p. 73–79, 2016.

- 27 SINGHEE, A.; RUTENBAR, R. A. Why quasi-monte carlo is better than monte carlo or latin hypercube sampling for statistical circuit analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, v. 29, n. 11, p. 1763–1776, November 2010. ISSN 0278-0070.
- 28 MAASAR, M. A. et al. Monte carlo amp;amp; quasi-monte carlo approach in option pricing. In: *2012 IEEE Symposium on Humanities, Science and Engineering Research*. [S.l.: s.n.], 2012. p. 1401–1405. ISSN 2378-9808.
- 29 MAIA, L. F.; VIANA, W.; TRINTA, F. Using monte carlo tree search and google maps to improve game balancing in location-based games. In: *2017 IEEE Conference on Computational Intelligence and Games (CIG)*. [S.l.: s.n.], 2017. p. 215–222. ISSN 2325-4289.
- 30 UHLMANN SIMON J. JULIER, M. C. J. K. Nondivergent simultaneous map-building and localization using covariance intersection. In: . [s.n.], 1997. v. 3087, p. 3087 – 3087 – 10. Disponível em: <<https://doi.org/10.1117/12.277216>>.
- 31 SCHRÖTER, M. et al. A semiphysical large-signal compact carbon nanotube fet model for analog rf applications. *IEEE Transactions on Electron Devices*, v. 62, n. 1, p. 52–60, January 2015. ISSN 0018-9383.
- 32 EATON, J. W. et al. *GNU Octave version 4.4.0 manual: a high-level interactive language for numerical computations*. [S.l.], 2018. Disponível em: <<https://www.gnu.org/software/octave/doc/v4.4.1/>>.
- 33 NGSPICE - open source spice simulator. Data de acesso: 21/03/2019. Disponível em: <<http://ngspice.sourceforge.net/>>.
- 34 MARTEL, R. et al. Single-and multi-wall carbon nanotube field-effect transistors. *Applied physics letters*, AIP, v. 73, n. 17, p. 2447–2449, 1998.
- 35 MATSUDA, Y.; TAHIR-KHELI, J.; III, W. A. G. Definitive band gaps for single-wall carbon nanotubes. *The Journal of Physical Chemistry Letters*, ACS Publications, v. 1, n. 19, p. 2946–2950, 2010.
- 36 SIVAKUMAR, V. et al. Role of reaction and factors of carbon nanotubes growth in chemical vapour decomposition process using methane: a highlight. *Journal of Nanomaterials*, Hindawi Publishing Corp., v. 2010, p. 11, 2010.
- 37 SCHROTER, M. et al. A 4 wafer photostepper-based carbon nanotube fet technology for rf applications. In: *IEEE. 2011 IEEE MTT-S International Microwave Symposium*. [S.l.], 2011. p. 1–4.
- 38 KONG, J. et al. Synthesis, integration, and electrical properties of individual single-walled carbon nanotubes. *Applied Physics A*, Springer, v. 69, n. 3, p. 305–308, 1999.
- 39 PARK, R. S. et al. Hysteresis-free carbon nanotube field-effect transistors. *ACS nano*, ACS Publications, v. 11, n. 5, p. 4785–4791, 2017.

- 40 SHULAKER, M. M. et al. Carbon nanotube computer. *Nature*, Nature Publishing Group, v. 501, n. 7468, p. 526, 2013.
- 41 KIM, W. et al. Hysteresis caused by water molecules in carbon nanotube field-effect transistors. *Nano Letters*, ACS Publications, v. 3, n. 2, p. 193–198, 2003.
- 42 MINOT, E. et al. Tuning carbon nanotube band gaps with strain. *Physical review letters*, APS, v. 90, n. 15, p. 156401, 2003.
- 43 VOGGU, R. et al. A simple method of separating metallic and semiconducting single-walled carbon nanotubes based on molecular charge transfer. *Journal of the American Chemical Society*, ACS Publications, v. 132, n. 16, p. 5560–5561, 2010.
- 44 PATIL, N. et al. Vmr: Vlsi-compatible metallic carbon nanotube removal for imperfection-immune cascaded multi-stage digital logic circuits using carbon nanotube fets. In: IEEE. *2009 IEEE International Electron Devices Meeting (IEDM)*. [S.l.], 2009. p. 1–4.
- 45 CLAUS, M. et al. Critical review of cntfet compact models. In: *NSTI-Nanotech (Workshop on Compact Modeling)*. [S.l.: s.n.], 2012. v. 2, p. 770–775.
- 46 RAYCHOWDHURY, A.; MUKHOPADHYAY, S.; ROY, K. A circuit-compatible model of ballistic carbon nanotube field-effect transistors. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, IEEE, v. 23, n. 10, p. 1411–1420, 2004.
- 47 DENG, J.; 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. *IEEE Transactions on Electron Devices*, IEEE, v. 54, n. 12, p. 3186–3194, 2007.
- 48 FRÉGONÈSE, S. et al. Computationally efficient physics-based compact cntfet model for circuit design. *IEEE Transactions on Electron Devices*, IEEE, v. 55, n. 6, p. 1317–1327, 2008.
- 49 FRÉGONÈSE, S.; MANEUX, C.; ZIMMER, T. A compact model for dual-gate one-dimensional fet: Application to carbon-nanotube fets. *IEEE transactions on electron devices*, IEEE, v. 58, n. 1, p. 206–215, 2010.
- 50 LUO, J. et al. Compact model for carbon nanotube field-effect transistors including nonidealities and calibrated with experimental data down to 9-nm gate length. *IEEE transactions on electron devices*, IEEE, v. 60, n. 6, p. 1834–1843, 2013.
- 51 GELAO, G. et al. A semiempirical spice model for n-type conventional cntfets. *IEEE Transactions on Nanotechnology*, IEEE, v. 10, n. 3, p. 506–512, 2010.
- 52 FRÉGONÈSE, S.; MANEUX, C.; ZIMMER, T. Implementation of tunneling phenomena in a cntfet compact model. *IEEE Transactions on Electron Devices*, IEEE, v. 56, n. 10, p. 2224–2231, 2009.
- 53 ALAM, A. U. et al. Rf linearity potential of carbon-nanotube transistors versus mosfets. *IEEE Transactions on Nanotechnology*, IEEE, v. 12, n. 3, p. 340–351, 2013.

- 54 SPECTRE Circuit Simulator. Data de acesso: 21/03/2019. Disponível em: <https://www.cadence.com/content/cadence-www/global/en_US/home/tools/custom-ic-analog-rf-design/circuit-simulation/spectre-circuit-simulator.html>.
- 55 ASHRAF, R. et al. Yield enhancement by tube redundancy in cnfet-based circuits. In: IEEE. *2010 17th IEEE International Conference on Electronics, Circuits and Systems*. [S.l.], 2010. p. 442–445.
- 56 HUANG, C.-Y. Package-on-package assembly yield assessment in the odm/ems environment using monte carlo simulation. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, IEEE, v. 3, n. 9, p. 1611–1620, 2013.
- 57 DOORN, T. et al. Importance sampling monte carlo simulations for accurate estimation of sram yield. In: IEEE. *ESSCIRC 2008-34th European Solid-State Circuits Conference*. [S.l.], 2008. p. 230–233.
- 58 UHLMANN, J. Simultaneous map building and localization for real time applications. transfer thesis, univ. Oxford, Oxford, UK, 1994.
- 59 MEDEIROS, J. E. G. d. *Unscented transform framework for quantization modeling in data conversion systems*. Tese (Doutorado) — Universidade de Brasília, 2017.
- 60 NGSPICE and ADMS for Verilog-AMS modeling. Data de acesso: 19/04/2019. Disponível em: <<http://ngspice.sourceforge.net/adms.html>>.
- 61 SCHRÖTER, M. et al. Cntfet-based rf electronics — state-of-the-art and future prospects. In: *2016 IEEE 16th Topical Meeting on Silicon Monolithic Integrated Circuits in RF Systems (SiRF)*. [S.l.: s.n.], 2016. p. 97–100.
- 62 HAFERLACH, M. et al. Trap-induced apparent linearity of cntfets. In: *Proc. WCM/NanoTech*. [S.l.: s.n.], 2014. p. 512–515.
- 63 AZEVEDO, H. A. de. Projeto de um misturador de frequências downconverter utilizando cntfets para um transceptor zigbee. 2018.
- 64 BARBOSA, L. L. A. Projeto e estimativa de desempenho/qualidade de um mixer upconverter para um transceptor zigbee utilizando transistores de efeito de campo de nanotubos de carbono. 2018.

Apêndice

A

APÊNDICE

Tabela A.1: Parâmetros elétricos do CCAM

Parâmetro	Descrição	Valor base	Restrições
Parâmetros DC de nanotubos semicondutores			
idstn	Fator de corrente de condução tipo N	0,3	$idstn \geq 0$
vthin	Tensão de limiar tipo N	-0,15	
vth0in	Parâmetro de ajuste de corrente tipo N	5	$vth0in > 0$
athin	Parâmetro de ajuste de corrente tipo N	2	$athin > 0$
sthin	Parâmetro de ajuste de inclinação sublimiar	0,2	$0 < sthin < 1$
idstp	Fator de condução de corrente tipo P	0,3	$idstp \geq 0$
vthip	Tensão de limiar tipo P	-0,03	
vth0ip	Parâmetro de ajuste de corrente tipo P	5	$vth0ip > 0$
athip	Parâmetro de ajuste de corrente tipo P	2	$athip > 0$
sthip	Parâmetro de ajuste de inclinação sublimiar	0,2	$0 < sthip < 1$
vdc	Parâmetro de ajuste de cálculo de corrente	2	$vdc > 0$
betdcr	Parâmetro de ajuste de inclinação sublimiar	1	$betdcr > 0$
smss	Parâmetro de transição para região sublimiar	10	$smss > 0$
facss	Parâmetro de ajuste DIBL	1	$facss \geq 0$
rscs	Resistência de contato da fonte de nanotubos semicondutores	20	$rscs \geq 0$
rdcs	Resistência de contato do dreno de nanotubos semicondutores	20	$rdcs \geq 0$
Parâmetros AC de nanotubos semicondutores			
ctn0	Fator de carga de elétrons	2e-13	$ctn0 \geq 0$
ctp0	Fator de carga de lacunas	2e-13	$ctp0 \geq 0$
vthqs	Tensão de limiar para Qs	-0,1	
vth0qs	Parâmetro de ajuste para cálculo de Qs	0,7	$vth0qs > 0$
athqs	Parâmetro de ajuste para cálculo de Qs	3	$athqs > 0$
vthqd	Tensão de limiar para Qd	-1	
vth0qd	Parâmetro de ajuste para cálculo de Qd	3	$vth0qd > 0$
athqd	Parâmetro de ajuste para cálculo de Qd	3	$athqd > 0$
pqsd	Fator de particionamento fonte/dreno	0,15	$0 < pqsd < 1$
Parâmetros de nanotubos metálicos			
rmta	Resistência em baixa tensão de nanotubos metálicos	30	$rmta \geq 0$
amt0	Parâmetro de saturação de nanotubos metálicos	25	$amt0 \geq 0$
cmt	Capacitância de nanotubos metálicos	2e-013	$cmt \geq 0$
pqmt	Fator de particionamento fonte/dreno	0,5	$0 < pqmt < 1$
rscm	Resistência de contato da fonte de nanotubos metálicos	8	$rscm \geq 0$
rdcm	Resistência de contato do dreno de nanotubos metálicos	8	$rdcm \geq 0$
Parâmetros para modelagem de ruído			
fanof	Fator Fano para cálculo de ruído de injeção em canal	1	$fanof \geq 0$

fanofmt	Fator Fano para cálculo de ruído em nanotubo metálico	1	fanofmt ≥ 0
hoogef	Parâmetro Hooge para cálculo de ruído 1/f	2e-3	hoogef ≥ 0
beta_fn	Inclinação de distribuição de frequência cintilante	1	beta_fn ≥ 0
Parâmetros de modelagem de confinamento			
atrap	Parâmetro de modelagem de confinamento	0,5	
btrap	Parâmetro de modelagem de confinamento	0,5	
ctrap	Parâmetro de modelagem de confinamento	2	
w0trap	Frequência inferior característica de confinamentos	0	w0trap ≥ 0
strap	Parâmetro de modelagem de confinamento	0	strap > 1
Parâmetros externos			
rsf	Resistência de fonte por dedo	0	rsf ≥ 0
rdf	Resistência de dreno por dedo	0	rdf ≥ 0
rg	Resistência de gate por dedo	10	rg ≥ 0
cgspar1	Capacitância parasita gate-fonte 1	0	cgspar1 ≥ 0
cgspar2	Capacitância parasita gate-fonte 2	3e-14	cgspar2 ≥ 0
cgdpar1	Capacitância parasita gate-dreno 1	8e-14	cgdpar1 ≥ 0
cgdpar2	Capacitância parasita gate-dreno 2	3e-14	cgdpar2 ≥ 0
cdspar	Capacitância parasita dreno-fonte	2e-14	cdspar ≥ 0

Tabela A.2: Parâmetros físicos base e parâmetros elétricos equivalentes

Parâmetros físicos	Parâmetros elétricos
n_gf = 10	idstn = 0.20473
w_g = 4.1	idstp = 0
l_g = 0.18	ctn0 = 8.2195e-14
l_gsd = 0.225	ctp0 = 8.2195e-14
l_sd = 6	vthin = -0.17594
d_cnt = 1.5	vthip = -0.032389
n_ta = 40	vth0in = 35.662
p_mt = 0.0001	athin = 18.029
h_mg = 0.19	sthin = 0.20000
h_msd = 0.15	vth0ip = 84.762
h_m1 = 0.07	athip = 9.0461
t_ox = 25	sthip = 0.20000
w_ms = 6	rmta = 165295.71220
w_mg = 5	amto = 147431.52439
w_do = 3.1	cmt = 1.4522e-17
eps_is = 4	pqmt = 0.49425
eps_ox = 10	rscs = 33.981
eps_ins = 4	rdcs = 15.230
	rscm = 59132.78049
	rdcm = 59132.78049
	vdcr = 1.5854
	betdcr = 1.0266

```
smss = 10
facss = 1
pqsd = 0.16667
vthqs = -0.10000
vth0qs = 0.66693
athqs = 3
vthqd = -1.1000
vth0qd = 3.0386
athqd = 2.6667
```

A.1 NETLISTS CRIADAS PARA A DISSERTAÇÃO

A.1.1 CNTFET_curva_de_transferencia.cir

Arquivo da *netlist* da curva de transferência do CNTFET

```
1 curva_de_transferencia
2
3 *** MODELS
4 * CNTFET
5 .MODEL mod1 nmos level=99
6 +idstn=M1_idstn vthin=M1_vthin vth0in=M1_vth0in athin=M1_athin
7 +sthin=M1_sthin idstp=M1_idstp vthip=M1_vthip vth0ip=M1_vth0ip ...
   athip=M1_athip
8 +sthip=M1_sthip vdcrcr=M1_vdcrcr betdcr=M1_betdcr smss=M1_smss ...
   facss=M1_facss
9 +rscs=M1_rscs rdcs=M1_rdcs ctn0=M1_ctn0 ctp0=M1_ctp0
10 +vthqs=M1_vthqs vth0qs=M1_vth0qs athqs=M1_athqs vthqd=M1_vthqd ...
   vth0qd=M1_vth0qd
11 +athqd=M1_athqd pqsd=M1_pqsd rmta=M1_rmta amto=M1_amto
12 +cmt=M1_cmt pqmt=M1_pqmt rscm=M1_rscm rdcm=M1_rdcm
13 +fanof=M1_fanof fanofmt=M1_fanofmt hoogef=M1_hoogef ...
   beta_fn=M1_beta_fn rsf=M1_rsf rdf=M1_rdf
14 +rg=M1_rg cgsparl=M1_cgsparl cgsparr2=M1_cgsparr2 cgdpar1=M1_cgdpar1
15 +cgdpar2=M1_cgdpar2 cdspar=M1_cdspar tnom=M1_tnom rths=M1_rths ...
   rthm=M1_rthm
16 +zetrmta=M1_zetrmta zetamto=M1_zetamto alvthin=M1_alvthin ...
   alvth0in=M1_alvth0in alvthip=M1_alvthip alvth0ip=M1_alvth0ip
17 +alidst=M1_alidst alct0=M1_alct0 alvthqs=M1_alvthqs ...
   alvth0qs=M1_alvth0qs alvthqd=M1_alvthqd alvth0qd=M1_alvth0qd
18 +alrscon=M1_alrscon alrmcon=M1_alrmcon atrap=M1_atrap ...
   btrap=M1_btrap ctrap=M1_ctrapped w0trap=M1_w0trap strap=M1_strap
19
```

```

20 *****
21
22 *** Circuit description
23
24 M1 k g 0 0 mod1
25
26 * Voltage sources
27 VGS g 0 0
28 VDS d 0 0.6
29
30 VIDS k d
31
32 *****
33
34 *** Simulation
35
36 .control
37 dc VGS 0 0.4 0.01
38
39 set wr_vecnames
40 set wr_singlescale
41 wrdata rawspice.raw -vids#branch
42
43 .endc
44 *****

```

A.1.2 CNTFET_curva_de_saida.cir

Arquivo da *netlist* da curva de saída do CNTFET

```

1 curva_de_saida
2
3 *** MODELS
4 * CNTFET
5 .MODEL mod1 nmos level=99
6 +idstn=M1_idstn vthin=M1_vthin vth0in=M1_vth0in athin=M1_athin
7 +sthin=M1_sthin idstp=M1_idstp vthip=M1_vthip vth0ip=M1_vth0ip ...
   athip=M1_athip
8 +sthip=M1_sthip vdcr=M1_vdcr betdcr=M1_betdcr smss=M1_smss ...
   facss=M1_facss
9 +rscs=M1_rscs rdcs=M1_rdcs ctn0=M1_ctn0 ctp0=M1_ctp0
10 +vthqs=M1_vthqs vth0qs=M1_vth0qs athqs=M1_athqs vthqd=M1_vthqd ...

```

```

    vth0qd=M1_vth0qd
11 +athqd=M1_athqd pqsd=M1_pqsd rmta=M1_rmta amto=M1_amto
12 +cmt=M1_cmt pqmt=M1_pqmt rscm=M1_rscm rdcM=M1_rdcM
13 +fanof=M1_fanof fanofmt=M1_fanofmt hoogef=M1_hoogef ...
    beta_fn=M1_beta_fn rsf=M1_rsf rdf=M1_rdf
14 +rg=M1_rg cgsparl=M1_cgsparl cgsparl2=M1_cgsparl2 cgdparl=M1_cgdparl
15 +cgdpar2=M1_cgdpar2 cdspar=M1_cdspar tnom=M1_tnom rths=M1_rths ...
    rthm=M1_rthm
16 +zetrmta=M1_zetrmta zetamto=M1_zetamto alvthin=M1_alvthin ...
    alvth0in=M1_alvth0in alvthip=M1_alvthip alvth0ip=M1_alvth0ip
17 +alidst=M1_alidst alct0=M1_alct0 alvthqs=M1_alvthqs ...
    alvth0qs=M1_alvth0qs alvthqd=M1_alvthqd alvth0qd=M1_alvth0qd
18 +alrscon=M1_alrscon alrmcon=M1_alrmcon atrap=M1_atrap ...
    btrap=M1_btrap ctrap=M1_ctrapped w0trap=M1_w0trap strap=M1_strap
19
20 *****
21
22 *** Circuit description
23
24 M1 d g 0 0 mod1
25
26 * Voltage sources
27 VGS g 0 0.4
28 VDS d 0 0
29
30 *VIGS k g
31
32 *****
33
34 *** Simulation
35
36 .control
37 dc VDS 0 1 0.01
38
39 set wr_vecnames
40 set wr_singlescale
41 wrdata rawspice.raw -vds#branch
42
43 .endc
44 *****

```

A.1.3 DC.cir

Arquivo da *netlist* do circuito *buffer* CML

```
1 curva_de_transferencia_CML_buffer
2
3 *** MODELS
4 * CNTFET
5 .MODEL mod1 nmos level=99
6 +idstn=M1_idstn vthin=M1_vthin vth0in=M1_vth0in athin=M1_athin
7 +sthin=M1_sthin idstp=M1_idstp vthip=M1_vthip vth0ip=M1_vth0ip ...
   athip=M1_athip
8 +sthip=M1_sthip vdcr=M1_vdcr betdcr=M1_betdcr smss=M1_smss ...
   facss=M1_facss
9 +rscs=M1_rscs rdcs=M1_rdcs ctn0=M1_ctn0 ctp0=M1_ctp0
10 +vthqs=M1_vthqs vth0qs=M1_vth0qs athqs=M1_athqs vthqd=M1_vthqd ...
   vth0qd=M1_vth0qd
11 +athqd=M1_athqd pqsd=M1_pqsd rmta=M1_rmta amto=M1_amto
12 +cmt=M1_cmt pqmt=M1_pqmt rscm=M1_rscm rdcm=M1_rdcn
13 +fanof=M1_fanof fanofmt=M1_fanofmt hoogef=M1_hoogef ...
   beta_fn=M1_beta_fn rsf=M1_rsf rdf=M1_rdf
14 +rg=M1_rg cgsparl=M1_cgsparl cgsparr2=M1_cgsparr2 cgdparl=M1_cgdparl
15 +cgdpar2=M1_cgdpar2 cdspar=M1_cdspar tnom=M1_tnom rths=M1_rths ...
   rthm=M1_rthm
16 +zetrmta=M1_zetrmta zetamto=M1_zetamto alvthin=M1_alvthin ...
   alvth0in=M1_alvth0in alvthip=M1_alvthip alvth0ip=M1_alvth0ip
17 +alidst=M1_alidst alct0=M1_alct0 alvthqs=M1_alvthqs ...
   alvth0qs=M1_alvth0qs alvthqd=M1_alvthqd alvth0qd=M1_alvth0qd
18 +alrscon=M1_alrscon alrmcon=M1_alrmcon atrap=M1_atrap ...
   btrap=M1_btrap ctrap=M1_ctrapp w0trap=M1_w0trap strap=M1_strap
19
20 .MODEL mod2 nmos level=99
21 +idstn=M2_idstn vthin=M2_vthin vth0in=M2_vth0in athin=M2_athin
22 +sthin=M2_sthin idstp=M2_idstp vthip=M2_vthip vth0ip=M2_vth0ip ...
   athip=M2_athip
23 +sthip=M2_sthip vdcr=M2_vdcr betdcr=M2_betdcr smss=M2_smss ...
   facss=M2_facss
24 +rscs=M2_rscs rdcs=M2_rdcs ctn0=M2_ctn0 ctp0=M2_ctp0
25 +vthqs=M2_vthqs vth0qs=M2_vth0qs athqs=M2_athqs vthqd=M2_vthqd ...
   vth0qd=M2_vth0qd
26 +athqd=M2_athqd pqsd=M2_pqsd rmta=M2_rmta amto=M2_amto
27 +cmt=M2_cmt pqmt=M2_pqmt rscm=M2_rscm rdcm=M2_rdcn
28 +fanof=M2_fanof fanofmt=M2_fanofmt hoogef=M2_hoogef ...
   beta_fn=M2_beta_fn rsf=M2_rsf rdf=M2_rdf
29 +rg=M2_rg cgsparl=M2_cgsparl cgsparr2=M2_cgsparr2 cgdparl=M2_cgdparl
```

```

30 +cgdpar2=M2_cgdpar2 cdspar=M2_cdspar tnom=M2_tnom rths=M2_rths ...
    rthm=M2_rthm
31 +zetrmta=M2_zetrmta zetamto=M2_zetamto alvthin=M2_alvthin ...
    alvth0in=M2_alvth0in alvthip=M2_alvthip alvth0ip=M2_alvth0ip
32 +alidst=M2_alidst alct0=M2_alct0 alvthqs=M2_alvthqs ...
    alvth0qs=M2_alvth0qs alvthqd=M2_alvthqd alvth0qd=M2_alvth0qd
33 +alrscon=M2_alrscon alrmcon=M2_alrmcon atrap=M2_atrap ...
    btrap=M2_btrap ctrap=M2_ctrp w0trap=M2_w0trap strap=M2_strap
34
35 *****
36
37 *** Circuit description
38
39 M1 out1 in1 net4 0 mod1
40 M2 out2 in2 net4 0 mod2
41
42 R1 net1 out1 333
43 R2 net1 out2 333
44 R3 net3 net5 1e-12
45 R4 net4 0 1e12
46
47 C1 net1 out1 58fF
48 C2 net1 out2 58fF
49
50
51 * Voltage sources
52 Vgslin in1 net2 0
53 Egs2in in2 net5 in1 net2 -1
54 Vgs1 net2 0 0.8
55 Vgs2 net3 0 0.8
56 Vdd net1 0 1.8
57
58 * Current sources
59 I1 net4 0 1.8mA
60
61 *****
62
63 *** Simulation
64
65 .control
66 *dc Vgslin -0.2 0.2 0.004
67 dc Vgslin 0 0 1
68
69 set wr_vecnames
70 set wr_singlescale

```

```

71 wrdata rawspice.raw out1, out2
72
73 *plot in1, in2
74 *plot out1, out2
75
76 .endc
77 *****

```

A.1.4 freq_3_nocap.cir

Arquivo da *netlist* do circuito oscilador em anel de três estágios

```

1 CML 3-stage ring Oscillator
2
3 *** MODELS
4 * CNTFET
5 .MODEL mod1 nmos level=99
6 +idstn=M1_idstn vthin=M1_vthin vth0in=M1_vth0in athin=M1_athin
7 +sthin=M1_sthin idstp=M1_idstp vthip=M1_vthip vth0ip=M1_vth0ip ...
   athip=M1_athip
8 +sthip=M1_sthip vdcr=M1_vdcr betdcr=M1_betdcr smss=M1_smss ...
   facss=M1_facss
9 +rscs=M1_rscs rdcs=M1_rdcs ctn0=M1_ctn0 ctp0=M1_ctp0
10 +vthqs=M1_vthqs vth0qs=M1_vth0qs athqs=M1_athqs vthqd=M1_vthqd ...
   vth0qd=M1_vth0qd
11 +athqd=M1_athqd pqsd=M1_pqsd rmta=M1_rmta amto=M1_amto
12 +cmt=M1_cmt pqmt=M1_pqmt rscm=M1_rscm rdcm=M1_rdcn
13 +fanof=M1_fanof fanofmt=M1_fanofmt hoogef=M1_hoogef ...
   beta_fn=M1_beta_fn rsf=M1_rsf rdf=M1_rdf
14 +rg=M1_rg cgsparl=M1_cgsparl cgsparr=M1_cgsparr cgdparl=M1_cgdparl
15 +cgdparr=M1_cgdparr cdspar=M1_cdspar tnom=M1_tnom rths=M1_rths ...
   rthm=M1_rthm
16 +zetrmta=M1_zetrmta zetamto=M1_zetamto alvthin=M1_alvthin ...
   alvth0in=M1_alvth0in alvthip=M1_alvthip alvth0ip=M1_alvth0ip
17 +alidst=M1_alidst alct0=M1_alct0 alvthqs=M1_alvthqs ...
   alvth0qs=M1_alvth0qs alvthqd=M1_alvthqd alvth0qd=M1_alvth0qd
18 +alrscon=M1_alrscon alrmcon=M1_alrmcon atrap=M1_atrap ...
   btrap=M1_btrap ctrap=M1_ctrapped w0trap=M1_w0trap strap=M1_strap
19
20 .MODEL mod2 nmos level=99
21 +idstn=M2_idstn vthin=M2_vthin vth0in=M2_vth0in athin=M2_athin
22 +sthin=M2_sthin idstp=M2_idstp vthip=M2_vthip vth0ip=M2_vth0ip ...

```

```

    athip=M2_athip
23 +sthip=M2_sthip vdcr=M2_vdcr betdcr=M2_betdcr smss=M2_smss ...
    facss=M2_facss
24 +rscs=M2_rscs rdcs=M2_rdcs ctn0=M2_ctn0 ctp0=M2_ctp0
25 +vthqs=M2_vthqs vth0qs=M2_vth0qs athqs=M2_athqs vthqd=M2_vthqd ...
    vth0qd=M2_vth0qd
26 +athqd=M2_athqd pqsd=M2_pqsd rmta=M2_rmta amto=M2_amto
27 +cmt=M2_cmt pqmt=M2_pqmt rscm=M2_rscm rdcm=M2_rdcm
28 +fanof=M2_fanof fanofmt=M2_fanofmt hoogef=M2_hoogef ...
    beta_fn=M2_beta_fn rsf=M2_rsf rdf=M2_rdf
29 +rg=M2_rg cgsparl=M2_cgsparl cgspar2=M2_cgspar2 cgdpar1=M2_cgdpar1
30 +cgdpar2=M2_cgdpar2 cdspar=M2_cdspar tnom=M2_tnom rths=M2_rths ...
    rthm=M2_rthm
31 +zetrmta=M2_zetrmta zetamto=M2_zetamto alvthin=M2_alvthin ...
    alvth0in=M2_alvth0in alvthip=M2_alvthip alvth0ip=M2_alvth0ip
32 +alidst=M2_alidst alct0=M2_alct0 alvthqs=M2_alvthqs ...
    alvth0qs=M2_alvth0qs alvthqd=M2_alvthqd alvth0qd=M2_alvth0qd
33 +alrscon=M2_alrscon alrmcon=M2_alrmcon atrap=M2_atrap ...
    btrap=M2_btrap ctrap=M2_ctrapp w0trap=M2_w0trap strap=M2_strap
34
35 .MODEL mod3 nmos level=99
36 +idstn=M3_idstn vthin=M3_vthin vth0in=M3_vth0in athin=M3_athin
37 +sthin=M3_sthin idstp=M3_idstp vthip=M3_vthip vth0ip=M3_vth0ip ...
    athip=M3_athip
38 +sthip=M3_sthip vdcr=M3_vdcr betdcr=M3_betdcr smss=M3_smss ...
    facss=M3_facss
39 +rscs=M3_rscs rdcs=M3_rdcs ctn0=M3_ctn0 ctp0=M3_ctp0
40 +vthqs=M3_vthqs vth0qs=M3_vth0qs athqs=M3_athqs vthqd=M3_vthqd ...
    vth0qd=M3_vth0qd
41 +athqd=M3_athqd pqsd=M3_pqsd rmta=M3_rmta amto=M3_amto
42 +cmt=M3_cmt pqmt=M3_pqmt rscm=M3_rscm rdcm=M3_rdcm
43 +fanof=M3_fanof fanofmt=M3_fanofmt hoogef=M3_hoogef ...
    beta_fn=M3_beta_fn rsf=M3_rsf rdf=M3_rdf
44 +rg=M3_rg cgsparl=M3_cgsparl cgspar2=M3_cgspar2 cgdpar1=M3_cgdpar1
45 +cgdpar2=M3_cgdpar2 cdspar=M3_cdspar tnom=M3_tnom rths=M3_rths ...
    rthm=M3_rthm
46 +zetrmta=M3_zetrmta zetamto=M3_zetamto alvthin=M3_alvthin ...
    alvth0in=M3_alvth0in alvthip=M3_alvthip alvth0ip=M3_alvth0ip
47 +alidst=M3_alidst alct0=M3_alct0 alvthqs=M3_alvthqs ...
    alvth0qs=M3_alvth0qs alvthqd=M3_alvthqd alvth0qd=M3_alvth0qd
48 +alrscon=M3_alrscon alrmcon=M3_alrmcon atrap=M3_atrap ...
    btrap=M3_btrap ctrap=M3_ctrapp w0trap=M3_w0trap strap=M3_strap
49
50 .MODEL mod4 nmos level=99
51 +idstn=M4_idstn vthin=M4_vthin vth0in=M4_vth0in athin=M4_athin

```



```

52 +sthin=M4_sthin idstp=M4_idstp vthip=M4_vthip vth0ip=M4_vth0ip ...
    athip=M4_athip
53 +sthip=M4_sthip vdcr=M4_vdcr betdcr=M4_betdcr smss=M4_smss ...
    facss=M4_facss
54 +rscs=M4_rscs rdcs=M4_rdcs ctn0=M4_ctn0 ctp0=M4_ctp0
55 +vthqs=M4_vthqs vth0qs=M4_vth0qs athqs=M4_athqs vthqd=M4_vthqd ...
    vth0qd=M4_vth0qd
56 +athqd=M4_athqd pqsd=M4_pqsd rmta=M4_rmta amto=M4_amto
57 +cmt=M4_cmt pqmt=M4_pqmt rscm=M4_rscm rdcm=M4_rdcm
58 +fanof=M4_fanof fanofmt=M4_fanofmt hoogef=M4_hoogef ...
    beta_fn=M4_beta_fn rsf=M4_rsf rdf=M4_rdf
59 +rg=M4_rg cgsparl=M4_cgsparl cgspar2=M4_cgspar2 cgdpar1=M4_cgdpar1
60 +cgdpar2=M4_cgdpar2 cdspar=M4_cdspar tnom=M4_tnom rths=M4_rths ...
    rthm=M4_rthm
61 +zetrmta=M4_zetrmta zetamto=M4_zetamto alvthin=M4_alvthin ...
    alvth0in=M4_alvth0in alvthip=M4_alvthip alvth0ip=M4_alvth0ip
62 +alidst=M4_alidst alct0=M4_alct0 alvthqs=M4_alvthqs ...
    alvth0qs=M4_alvth0qs alvthqd=M4_alvthqd alvth0qd=M4_alvth0qd
63 +alrscon=M4_alrscon alrmcon=M4_alrmcon atrap=M4_atrap ...
    btrap=M4_btrap ctrap=M4_ctrp w0trap=M4_w0trap strap=M4_strap
64
65 .MODEL mod5 nmos level=99
66 +idstn=M5_idstn vthin=M5_vthin vth0in=M5_vth0in athin=M5_athin
67 +sthin=M5_sthin idstp=M5_idstp vthip=M5_vthip vth0ip=M5_vth0ip ...
    athip=M5_athip
68 +sthip=M5_sthip vdcr=M5_vdcr betdcr=M5_betdcr smss=M5_smss ...
    facss=M5_facss
69 +rscs=M5_rscs rdcs=M5_rdcs ctn0=M5_ctn0 ctp0=M5_ctp0
70 +vthqs=M5_vthqs vth0qs=M5_vth0qs athqs=M5_athqs vthqd=M5_vthqd ...
    vth0qd=M5_vth0qd
71 +athqd=M5_athqd pqsd=M5_pqsd rmta=M5_rmta amto=M5_amto
72 +cmt=M5_cmt pqmt=M5_pqmt rscm=M5_rscm rdcm=M5_rdcm
73 +fanof=M5_fanof fanofmt=M5_fanofmt hoogef=M5_hoogef ...
    beta_fn=M5_beta_fn rsf=M5_rsf rdf=M5_rdf
74 +rg=M5_rg cgsparl=M5_cgsparl cgspar2=M5_cgspar2 cgdpar1=M5_cgdpar1
75 +cgdpar2=M5_cgdpar2 cdspar=M5_cdspar tnom=M5_tnom rths=M5_rths ...
    rthm=M5_rthm
76 +zetrmta=M5_zetrmta zetamto=M5_zetamto alvthin=M5_alvthin ...
    alvth0in=M5_alvth0in alvthip=M5_alvthip alvth0ip=M5_alvth0ip
77 +alidst=M5_alidst alct0=M5_alct0 alvthqs=M5_alvthqs ...
    alvth0qs=M5_alvth0qs alvthqd=M5_alvthqd alvth0qd=M5_alvth0qd
78 +alrscon=M5_alrscon alrmcon=M5_alrmcon atrap=M5_atrap ...
    btrap=M5_btrap ctrap=M5_ctrp w0trap=M5_w0trap strap=M5_strap
79
80 .MODEL mod6 nmos level=99

```

```

81 +idstn=M6_idstn vthin=M6_vthin vth0in=M6_vth0in athin=M6_athin
82 +sthin=M6_sthin idstp=M6_idstp vthip=M6_vthip vth0ip=M6_vth0ip ...
    athip=M6_athip
83 +sthip=M6_sthip vdcr=M6_vdcr betdcr=M6_betdcr smss=M6_smss ...
    facss=M6_facss
84 +rscs=M6_rscs rdcs=M6_rdcs ctn0=M6_ctn0 ctp0=M6_ctp0
85 +vthqs=M6_vthqs vth0qs=M6_vth0qs athqs=M6_athqs vthqd=M6_vthqd ...
    vth0qd=M6_vth0qd
86 +athqd=M6_athqd pqsd=M6_pqsd rmta=M6_rmta amto=M6_amto
87 +cmt=M6_cmt pqmt=M6_pqmt rscm=M6_rscm rdcm=M6_rdcm
88 +fanof=M6_fanof fanofmt=M6_fanofmt hoogef=M6_hoogef ...
    beta_fn=M6_beta_fn rsf=M6_rsf rdf=M6_rdf
89 +rg=M6_rg cgsparl=M6_cgsparl cgspar2=M6_cgspar2 cgdparl=M6_cgdparl
90 +cgdpar2=M6_cgdpar2 cdspar=M6_cdspar tnom=M6_tnom rths=M6_rths ...
    rthm=M6_rthm
91 +zetrmta=M6_zetrmta zetamto=M6_zetamto alvthin=M6_alvthin ...
    alvth0in=M6_alvth0in alvthip=M6_alvthip alvth0ip=M6_alvth0ip
92 +alidst=M6_alidst alct0=M6_alct0 alvthqs=M6_alvthqs ...
    alvth0qs=M6_alvth0qs alvthqd=M6_alvthqd alvth0qd=M6_alvth0qd
93 +alrscon=M6_alrscon alrmcon=M6_alrmcon atrap=M6_atrap ...
    btrap=M6_btrap ctrap=M6_ctrapped w0trap=M6_w0trap strap=M6_strap
94
95 * Switch
96 .MODEL switch1 sw vt=0 vh=0 ron=1e-3 roff=1e12
97
98 *****
99 *** Circuit description
100 * Transistors
101 * MXXXXXXX nd ng ns nb mname
102 M1 net16 net1 net17 0 mod1
103 M2 net18 net5 net17 0 mod2
104 M3 out1 net20 net21 0 mod3
105 M4 out2 net23 net21 0 mod4
106 M5 net13 in1 net24 0 mod5
107 M6 net12 in2 net24 0 mod6
108
109 * Resistors
110 R1 net19 net16 333
111 R2 net18 net19 333
112 R3 net22 out2 333
113 R4 out1 net22 333
114 R5 net25 net13 333
115 R6 net12 net25 333
116 Ri1 net17 0 1e9
117 Ri2 net21 0 1e9

```

```

118 Ri3 net24 0 1e9
119
120 * Capacitors
121 *C1 net19 net16 58f
122 *C2 net19 net18 58f
123 *C3 net22 out2 58f
124 *C4 net22 out1 58f
125 *C5 net25 net13 58f
126 *C6 net25 net12 58f
127
128 * Voltage sources
129 Vdd1 net19 0 1.8
130 Vdd2 net22 0 1.8
131 Vdd3 net25 0 1.8
132 V1 net12 net5 0.7
133 V2 net13 net1 0.7
134 V3 net16 net20 0.7
135 V4 net18 net23 0.7
136 V5 out1 in1 0.7
137 V6 out2 in2 0.7
138 V7 net26 0 pulse(0.6 0 0.1ns 0.01ns 0.01ns 10000ns 10000ns)
139 V8 net27 0 pulse(1.2 0 0.1ns 0.01ns 0.01ns 10000ns 10000ns)
140
141 * Current sources
142 I1 net17 0 1.8m
143 I2 net21 0 1.8m
144 I3 net24 0 1.8m
145
146 * Voltage-controlled switches (start-up circuit)
147 S1 net26 in1 net27 0 switch1 on
148 S2 net27 in2 net27 0 switch1 on
149
150 *****
151 *** Simulation
152
153 .control
154 tran 0.005ns 300ns 50ns
155 set wr_vecnames
156 set wr_singlescale
157 wrdata rawspice.raw out1 out2 out1-out2
158
159 .endc
160 *****

```

A.1.5 freq_5_nocap.cir

Arquivo da *netlist* do circuito oscilador em anel de cinco estágios

```
1 CML 5-ring Oscillator
2
3 *** MODELS
4 * CNTFET
5 .MODEL mod1 nmos level=99
6 +idstn=M1_idstn vthin=M1_vthin vth0in=M1_vth0in athin=M1_athin
7 +sthin=M1_sthin idstp=M1_idstp vthip=M1_vthip vth0ip=M1_vth0ip ...
   athip=M1_athip
8 +sthip=M1_sthip vdcr=M1_vdcr betdcr=M1_betdcr smss=M1_smss ...
   facss=M1_facss
9 +rscs=M1_rscs rdcs=M1_rdcs ctn0=M1_ctn0 ctp0=M1_ctp0
10 +vthqs=M1_vthqs vth0qs=M1_vth0qs athqs=M1_athqs vthqd=M1_vthqd ...
   vth0qd=M1_vth0qd
11 +athqd=M1_athqd pqsd=M1_pqsd rmta=M1_rmta amto=M1_amto
12 +cmt=M1_cmt pqmt=M1_pqmt rscm=M1_rscm rdcm=M1_rdcm
13 +fanof=M1_fanof fanofmt=M1_fanofmt hoogef=M1_hoogef ...
   beta_fn=M1_beta_fn rsf=M1_rsf rdf=M1_rdf
14 +rg=M1_rg cgsparl=M1_cgsparl cgsparr2=M1_cgsparr2 cgdpar1=M1_cgdpar1
15 +cgdpar2=M1_cgdpar2 cdspar=M1_cdspar tnom=M1_tnom rths=M1_rths ...
   rthm=M1_rthm
16 +zetrmta=M1_zetrmta zetamto=M1_zetamto alvthin=M1_alvthin ...
   alvth0in=M1_alvth0in alvthip=M1_alvthip alvth0ip=M1_alvth0ip
17 +alidst=M1_alidst alct0=M1_alct0 alvthqs=M1_alvthqs ...
   alvth0qs=M1_alvth0qs alvthqd=M1_alvthqd alvth0qd=M1_alvth0qd
18 +alrscon=M1_alrscon alrmcon=M1_alrmcon atrap=M1_atrap ...
   btrap=M1_btrap ctrap=M1_ctrapp w0trap=M1_w0trap strap=M1_strap
19
20 .MODEL mod2 nmos level=99
21 +idstn=M2_idstn vthin=M2_vthin vth0in=M2_vth0in athin=M2_athin
22 +sthin=M2_sthin idstp=M2_idstp vthip=M2_vthip vth0ip=M2_vth0ip ...
   athip=M2_athip
23 +sthip=M2_sthip vdcr=M2_vdcr betdcr=M2_betdcr smss=M2_smss ...
   facss=M2_facss
24 +rscs=M2_rscs rdcs=M2_rdcs ctn0=M2_ctn0 ctp0=M2_ctp0
25 +vthqs=M2_vthqs vth0qs=M2_vth0qs athqs=M2_athqs vthqd=M2_vthqd ...
   vth0qd=M2_vth0qd
26 +athqd=M2_athqd pqsd=M2_pqsd rmta=M2_rmta amto=M2_amto
27 +cmt=M2_cmt pqmt=M2_pqmt rscm=M2_rscm rdcm=M2_rdcm
28 +fanof=M2_fanof fanofmt=M2_fanofmt hoogef=M2_hoogef ...
   beta_fn=M2_beta_fn rsf=M2_rsf rdf=M2_rdf
29 +rg=M2_rg cgsparl=M2_cgsparl cgsparr2=M2_cgsparr2 cgdpar1=M2_cgdpar1
```

```

30 +cgdpar2=M2_cgdpar2 cdspar=M2_cdspar tnom=M2_tnom rths=M2_rths ...
    rthm=M2_rthm
31 +zetrmta=M2_zetrmta zetamto=M2_zetamto alvthin=M2_alvthin ...
    alvth0in=M2_alvth0in alvthip=M2_alvthip alvth0ip=M2_alvth0ip
32 +alidst=M2_alidst alct0=M2_alct0 alvthqs=M2_alvthqs ...
    alvth0qs=M2_alvth0qs alvthqd=M2_alvthqd alvth0qd=M2_alvth0qd
33 +alrscon=M2_alrscon alrmcon=M2_alrmcon atrap=M2_atrap ...
    btrap=M2_btrap ctrap=M2_ctrapped w0trap=M2_w0trap strap=M2_strap
34
35 .MODEL mod3 nmos level=99
36 +idstn=M3_idstn vthin=M3_vthin vth0in=M3_vth0in athin=M3_athin
37 +sthin=M3_sthin idstp=M3_idstp vthip=M3_vthip vth0ip=M3_vth0ip ...
    athip=M3_athip
38 +sthip=M3_sthip vdcr=M3_vdcr betdcr=M3_betdcr smss=M3_smss ...
    facss=M3_facss
39 +rscs=M3_rscs rdcs=M3_rdcs ctn0=M3_ctn0 ctp0=M3_ctp0
40 +vthqs=M3_vthqs vth0qs=M3_vth0qs athqs=M3_athqs vthqd=M3_vthqd ...
    vth0qd=M3_vth0qd
41 +athqd=M3_athqd pqsd=M3_pqsd rmta=M3_rmta amto=M3_amto
42 +cmt=M3_cmt pqmt=M3_pqmt rscm=M3_rscm rdcm=M3_rdcm
43 +fanof=M3_fanof fanofmt=M3_fanofmt hoogef=M3_hoogef ...
    beta_fn=M3_beta_fn rsf=M3_rsf rdf=M3_rdf
44 +rg=M3_rg cgsparl=M3_cgsparl cgspar2=M3_cgspar2 cgdpar1=M3_cgdpar1
45 +cgdpar2=M3_cgdpar2 cdspar=M3_cdspar tnom=M3_tnom rths=M3_rths ...
    rthm=M3_rthm
46 +zetrmta=M3_zetrmta zetamto=M3_zetamto alvthin=M3_alvthin ...
    alvth0in=M3_alvth0in alvthip=M3_alvthip alvth0ip=M3_alvth0ip
47 +alidst=M3_alidst alct0=M3_alct0 alvthqs=M3_alvthqs ...
    alvth0qs=M3_alvth0qs alvthqd=M3_alvthqd alvth0qd=M3_alvth0qd
48 +alrscon=M3_alrscon alrmcon=M3_alrmcon atrap=M3_atrap ...
    btrap=M3_btrap ctrap=M3_ctrapped w0trap=M3_w0trap strap=M3_strap
49
50 .MODEL mod4 nmos level=99
51 +idstn=M4_idstn vthin=M4_vthin vth0in=M4_vth0in athin=M4_athin
52 +sthin=M4_sthin idstp=M4_idstp vthip=M4_vthip vth0ip=M4_vth0ip ...
    athip=M4_athip
53 +sthip=M4_sthip vdcr=M4_vdcr betdcr=M4_betdcr smss=M4_smss ...
    facss=M4_facss
54 +rscs=M4_rscs rdcs=M4_rdcs ctn0=M4_ctn0 ctp0=M4_ctp0
55 +vthqs=M4_vthqs vth0qs=M4_vth0qs athqs=M4_athqs vthqd=M4_vthqd ...
    vth0qd=M4_vth0qd
56 +athqd=M4_athqd pqsd=M4_pqsd rmta=M4_rmta amto=M4_amto
57 +cmt=M4_cmt pqmt=M4_pqmt rscm=M4_rscm rdcm=M4_rdcm
58 +fanof=M4_fanof fanofmt=M4_fanofmt hoogef=M4_hoogef ...
    beta_fn=M4_beta_fn rsf=M4_rsf rdf=M4_rdf

```

```

59 +rg=M4_rg cgsparl=M4_cgsparl cgspar2=M4_cgspar2 cgdpar1=M4_cgdpar1
60 +cgdpar2=M4_cgdpar2 cdspar=M4_cdspar tnom=M4_tnom rths=M4_rths ...
    rthm=M4_rthm
61 +zetrmta=M4_zetrmta zetamto=M4_zetamto alvthin=M4_alvthin ...
    alvth0in=M4_alvth0in alvthip=M4_alvthip alvth0ip=M4_alvth0ip
62 +alidst=M4_alidst alct0=M4_alct0 alvthqs=M4_alvthqs ...
    alvth0qs=M4_alvth0qs alvthqd=M4_alvthqd alvth0qd=M4_alvth0qd
63 +alrscon=M4_alrscon alrmcon=M4_alrmcon atrap=M4_atrap ...
    btrap=M4_btrap ctrap=M4_ctrapped w0trap=M4_w0trap strap=M4_strap
64
65 .MODEL mod5 nmos level=99
66 +idstn=M5_idstn vthin=M5_vthin vth0in=M5_vth0in athin=M5_athin
67 +sthin=M5_sthin idstp=M5_idstp vthip=M5_vthip vth0ip=M5_vth0ip ...
    athip=M5_athip
68 +sthip=M5_sthip vdcr=M5_vdcr betdcr=M5_betdcr smss=M5_smss ...
    facss=M5_facss
69 +rscs=M5_rscs rdcs=M5_rdcs ctn0=M5_ctn0 ctp0=M5_ctp0
70 +vthqs=M5_vthqs vth0qs=M5_vth0qs athqs=M5_athqs vthqd=M5_vthqd ...
    vth0qd=M5_vth0qd
71 +athqd=M5_athqd pqsd=M5_pqsd rmta=M5_rmta amto=M5_amto
72 +cmt=M5_cmt pqmt=M5_pqmt rscm=M5_rscm rdcm=M5_rdcm
73 +fanof=M5_fanof fanofmt=M5_fanofmt hoogef=M5_hoogef ...
    beta_fn=M5_beta_fn rsf=M5_rsf rdf=M5_rdf
74 +rg=M5_rg cgsparl=M5_cgsparl cgspar2=M5_cgspar2 cgdpar1=M5_cgdpar1
75 +cgdpar2=M5_cgdpar2 cdspar=M5_cdspar tnom=M5_tnom rths=M5_rths ...
    rthm=M5_rthm
76 +zetrmta=M5_zetrmta zetamto=M5_zetamto alvthin=M5_alvthin ...
    alvth0in=M5_alvth0in alvthip=M5_alvthip alvth0ip=M5_alvth0ip
77 +alidst=M5_alidst alct0=M5_alct0 alvthqs=M5_alvthqs ...
    alvth0qs=M5_alvth0qs alvthqd=M5_alvthqd alvth0qd=M5_alvth0qd
78 +alrscon=M5_alrscon alrmcon=M5_alrmcon atrap=M5_atrap ...
    btrap=M5_btrap ctrap=M5_ctrapped w0trap=M5_w0trap strap=M5_strap
79
80 .MODEL mod6 nmos level=99
81 +idstn=M6_idstn vthin=M6_vthin vth0in=M6_vth0in athin=M6_athin
82 +sthin=M6_sthin idstp=M6_idstp vthip=M6_vthip vth0ip=M6_vth0ip ...
    athip=M6_athip
83 +sthip=M6_sthip vdcr=M6_vdcr betdcr=M6_betdcr smss=M6_smss ...
    facss=M6_facss
84 +rscs=M6_rscs rdcs=M6_rdcs ctn0=M6_ctn0 ctp0=M6_ctp0
85 +vthqs=M6_vthqs vth0qs=M6_vth0qs athqs=M6_athqs vthqd=M6_vthqd ...
    vth0qd=M6_vth0qd
86 +athqd=M6_athqd pqsd=M6_pqsd rmta=M6_rmta amto=M6_amto
87 +cmt=M6_cmt pqmt=M6_pqmt rscm=M6_rscm rdcm=M6_rdcm
88 +fanof=M6_fanof fanofmt=M6_fanofmt hoogef=M6_hoogef ...

```

```

      beta_fn=M6_beta_fn rsf=M6_rsf rdf=M6_rdf
89 +rg=M6_rg cgspar1=M6_cgspar1 cgspar2=M6_cgspar2 cgdpar1=M6_cgdpar1
90 +cgdpar2=M6_cgdpar2 cdspar=M6_cdspar tnom=M6_tnom rths=M6_rths ...
      rthm=M6_rthm
91 +zetrmta=M6_zetrmta zetamto=M6_zetamto alvthin=M6_alvthin ...
      alvth0in=M6_alvth0in alvthip=M6_alvthip alvth0ip=M6_alvth0ip
92 +alidst=M6_alidst alct0=M6_alct0 alvthqs=M6_alvthqs ...
      alvth0qs=M6_alvth0qs alvthqd=M6_alvthqd alvth0qd=M6_alvth0qd
93 +alrscon=M6_alrscon alrmcon=M6_alrmcon atrap=M6_atrap ...
      btrap=M6_btrap ctrap=M6_ctrapped w0trap=M6_w0trap strap=M6_strap
94
95 .MODEL mod7 nmos level=99
96 +idstn=M7_idstn vthin=M7_vthin vth0in=M7_vth0in athin=M7_athin
97 +sthin=M7_sthin idstp=M7_idstp vthip=M7_vthip vth0ip=M7_vth0ip ...
      athip=M7_athip
98 +sthip=M7_sthip vdcr=M7_vdcr betdcr=M7_betdcr smss=M7_smss ...
      facss=M7_facss
99 +rscs=M7_rscs rdcs=M7_rdcs ctn0=M7_ctn0 ctp0=M7_ctp0
100 +vthqs=M7_vthqs vth0qs=M7_vth0qs athqs=M7_athqs vthqd=M7_vthqd ...
      vth0qd=M7_vth0qd
101 +athqd=M7_athqd pqsd=M7_pqsd rmta=M7_rmta amto=M7_amto
102 +cmt=M7_cmt pqmt=M7_pqmt rscm=M7_rscm rdcm=M7_rdcm
103 +fanof=M7_fanof fanofmt=M7_fanofmt hoogef=M7_hoogef ...
      beta_fn=M7_beta_fn rsf=M7_rsf rdf=M7_rdf
104 +rg=M7_rg cgspar1=M7_cgspar1 cgspar2=M7_cgspar2 cgdpar1=M7_cgdpar1
105 +cgdpar2=M7_cgdpar2 cdspar=M7_cdspar tnom=M7_tnom rths=M7_rths ...
      rthm=M7_rthm
106 +zetrmta=M7_zetrmta zetamto=M7_zetamto alvthin=M7_alvthin ...
      alvth0in=M7_alvth0in alvthip=M7_alvthip alvth0ip=M7_alvth0ip
107 +alidst=M7_alidst alct0=M7_alct0 alvthqs=M7_alvthqs ...
      alvth0qs=M7_alvth0qs alvthqd=M7_alvthqd alvth0qd=M7_alvth0qd
108 +alrscon=M7_alrscon alrmcon=M7_alrmcon atrap=M7_atrap ...
      btrap=M7_btrap ctrap=M7_ctrapped w0trap=M7_w0trap strap=M7_strap
109
110 .MODEL mod8 nmos level=99
111 +idstn=M8_idstn vthin=M8_vthin vth0in=M8_vth0in athin=M8_athin
112 +sthin=M8_sthin idstp=M8_idstp vthip=M8_vthip vth0ip=M8_vth0ip ...
      athip=M8_athip
113 +sthip=M8_sthip vdcr=M8_vdcr betdcr=M8_betdcr smss=M8_smss ...
      facss=M8_facss
114 +rscs=M8_rscs rdcs=M8_rdcs ctn0=M8_ctn0 ctp0=M8_ctp0
115 +vthqs=M8_vthqs vth0qs=M8_vth0qs athqs=M8_athqs vthqd=M8_vthqd ...
      vth0qd=M8_vth0qd
116 +athqd=M8_athqd pqsd=M8_pqsd rmta=M8_rmta amto=M8_amto
117 +cmt=M8_cmt pqmt=M8_pqmt rscm=M8_rscm rdcm=M8_rdcm

```

```

118 +fanof=M8_fanof fanofmt=M8_fanofmt hoogef=M8_hoogef ...
      beta_fn=M8_beta_fn rsf=M8_rsf rdf=M8_rdf
119 +rg=M8_rg cgspar1=M8_cgspar1 cgspar2=M8_cgspar2 cgdpar1=M8_cgdpar1
120 +cgdpar2=M8_cgdpar2 cdspar=M8_cdspar tnom=M8_tnom rths=M8_rths ...
      rthm=M8_rthm
121 +zetrmta=M8_zetrmta zetamto=M8_zetamto alvthin=M8_alvthin ...
      alvth0in=M8_alvth0in alvthip=M8_alvthip alvth0ip=M8_alvth0ip
122 +alidst=M8_alidst alct0=M8_alct0 alvthqs=M8_alvthqs ...
      alvth0qs=M8_alvth0qs alvthqd=M8_alvthqd alvth0qd=M8_alvth0qd
123 +alrscon=M8_alrscon alrmcon=M8_alrmcon atrap=M8_atrap ...
      btrap=M8_btrap ctrap=M8_ctrapp w0trap=M8_w0trap strap=M8_strap
124
125 .MODEL mod9 nmos level=99
126 +idstn=M9_idstn vthin=M9_vthin vth0in=M9_vth0in athin=M9_athin
127 +sthin=M9_sthin idstp=M9_idstp vthip=M9_vthip vth0ip=M9_vth0ip ...
      athip=M9_athip
128 +sthip=M9_sthip vdcr=M9_vdcr betdcr=M9_betdcr smss=M9_smss ...
      facss=M9_facss
129 +rscs=M9_rscs rdcs=M9_rdcs ctn0=M9_ctn0 ctp0=M9_ctp0
130 +vthqs=M9_vthqs vth0qs=M9_vth0qs athqs=M9_athqs vthqd=M9_vthqd ...
      vth0qd=M9_vth0qd
131 +athqd=M9_athqd pqsd=M9_pqsd rmta=M9_rmta amto=M9_amto
132 +cmt=M9_cmt pqmt=M9_pqmt rscm=M9_rscm rdcm=M9_rdcm
133 +fanof=M9_fanof fanofmt=M9_fanofmt hoogef=M9_hoogef ...
      beta_fn=M9_beta_fn rsf=M9_rsf rdf=M9_rdf
134 +rg=M9_rg cgspar1=M9_cgspar1 cgspar2=M9_cgspar2 cgdpar1=M9_cgdpar1
135 +cgdpar2=M9_cgdpar2 cdspar=M9_cdspar tnom=M9_tnom rths=M9_rths ...
      rthm=M9_rthm
136 +zetrmta=M9_zetrmta zetamto=M9_zetamto alvthin=M9_alvthin ...
      alvth0in=M9_alvth0in alvthip=M9_alvthip alvth0ip=M9_alvth0ip
137 +alidst=M9_alidst alct0=M9_alct0 alvthqs=M9_alvthqs ...
      alvth0qs=M9_alvth0qs alvthqd=M9_alvthqd alvth0qd=M9_alvth0qd
138 +alrscon=M9_alrscon alrmcon=M9_alrmcon atrap=M9_atrap ...
      btrap=M9_btrap ctrap=M9_ctrapp w0trap=M9_w0trap strap=M9_strap
139
140 .MODEL mod10 nmos level=99
141 +idstn=M10_idstn vthin=M10_vthin vth0in=M10_vth0in athin=M10_athin
142 +sthin=M10_sthin idstp=M10_idstp vthip=M10_vthip vth0ip=M10_vth0ip ...
      athip=M10_athip
143 +sthip=M10_sthip vdcr=M10_vdcr betdcr=M10_betdcr smss=M10_smss ...
      facss=M10_facss
144 +rscs=M10_rscs rdcs=M10_rdcs ctn0=M10_ctn0 ctp0=M10_ctp0
145 +vthqs=M10_vthqs vth0qs=M10_vth0qs athqs=M10_athqs vthqd=M10_vthqd ...
      vth0qd=M10_vth0qd
146 +athqd=M10_athqd pqsd=M10_pqsd rmta=M10_rmta amto=M10_amto

```



```

147 +cmt=M10_cmt pqmt=M10_pqmt rscm=M10_rscm rdcM=M10_rdcM
148 +fanof=M10_fanof fanofmt=M10_fanofmt hoogef=M10_hoogef ...
      beta_fn=M10_beta_fn rsf=M10_rsf rdf=M10_rdf
149 +rg=M10_rg cgspar1=M10_cgspAr1 cgspar2=M10_cgspAr2 cgdpar1=M10_cgdpar1
150 +cgdpar2=M10_cgdpar2 cdspar=M10_cdspar tnom=M10_tnom rths=M10_rths ...
      rthm=M10_rthm
151 +zetrmta=M10_zetrmta zetamto=M10_zetamto alvthin=M10_alvthin ...
      alvth0in=M10_alvth0in alvthip=M10_alvthip alvth0ip=M10_alvth0ip
152 +alidst=M10_alidst alct0=M10_alct0 alvthqs=M10_alvthqs ...
      alvth0qs=M10_alvth0qs alvthqd=M10_alvthqd alvth0qd=M10_alvth0qd
153 +alrscon=M10_alrscon alrmcon=M10_alrmcon atrap=M10_atrap ...
      btrap=M10_btrap ctrap=M10_ctrAp w0trap=M10_w0trap strap=M10_strap
154
155 * Switch
156 .MODEL switch1 sw vt=0 vh=0 ron=1e-3 roff=1e12
157
158 *****
159 *** Circuit description
160 * Transistors
161 * MXXXXXXX nd ng ns nb mname
162 M1 net0 net1 net2 0 mod1
163 M2 net3 net5 net2 0 mod2
164 M3 net6 net7 net8 0 mod3
165 M4 net9 net11 net8 0 mod4
166 M5 net16 net15 net17 0 mod5
167 M6 net18 net14 net17 0 mod6
168 M7 out1 net20 net21 0 mod7
169 M8 out2 net23 net21 0 mod8
170 M9 net13 in1 net24 0 mod9
171 M10 net12 in2 net24 0 mod10
172
173 * Resistors
174 R1 net4 net0 333
175 R2 net3 net4 333
176 R3 net10 net6 333
177 R4 net9 net10 333
178 R5 net19 net16 333
179 R6 net18 net19 333
180 R7 net22 out2 333
181 R8 out1 net22 333
182 R9 net25 net13 333
183 R10 net12 net25 333
184 Ri1 net2 0 1e9
185 Ri2 net8 0 1e9
186 Ri3 net17 0 1e9

```

```

187 Ri4 net21 0 1e9
188 Ri5 net24 0 1e9
189
190 * Capacitors
191 *C1 net4 net0 58f
192 *C2 net4 net3 58f
193 *C3 net10 net6 58f
194 *C4 net10 net9 58f
195 *C5 net19 net16 58f
196 *C6 net19 net18 58f
197 *C7 net22 out2 58f
198 *C8 net22 out1 58f
199 *C9 net25 net13 58f
200 *C10 net25 net12 58f
201
202 * Voltage sources
203 Vdd1 net4 0 1.8
204 Vdd2 net10 0 1.8
205 Vdd3 net19 0 1.8
206 Vdd4 net22 0 1.8
207 Vdd5 net25 0 1.8
208 V1 net3 net11 0.7
209 V2 net0 net7 0.7
210 V3 net12 net5 0.7
211 V4 net13 net1 0.7
212 V5 net9 net14 0.7
213 V6 net6 net15 0.7
214 V7 net16 net20 0.7
215 V8 net18 net23 0.7
216 V9 out1 in1 0.7
217 V10 out2 in2 0.7
218 V11 net26 0 pulse(0.6 0 0.001ns 0.001ns 0.01ns 10000ns 10000ns)
219 V12 net27 0 pulse(1.2 0 0.001ns 0.001ns 0.01ns 10000ns 10000ns)
220
221 * Current sources
222 I1 net2 0 1.8m
223 I2 net8 0 1.8m
224 I3 net17 0 1.8m
225 I4 net21 0 1.8m
226 I5 net24 0 1.8m
227
228 * Voltage-controlled switches (start-up circuit)
229 S1 net26 in1 net27 0 switch1 on
230 S2 net27 in2 net27 0 switch1 on
231

```

```
232 *****
233 *** Simulation
234
235 .control
236 tran 0.005ns 300ns 50ns
237 set wr_vecnames
238 set wr_singlescale
239 wrdata rawspice.raw out1 out2 out1-out2
240
241 .endc
242 *****
```