

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

WALTER ENRIQUE CALIENES BARTRA

**Ferramentas para a Simulação de Falhas
Transientes**

Dissertação apresentada como requisito parcial
para obtenção do grau de Mestre em
Microeletrônica

Prof. Dr. Ricardo Augusto da Luz Reis
Orientador

Prof. Dra. Fernanda G. de Lima Kastensmidt
Co-orientador

Porto Alegre, dezembro de 2011

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Calienes Bartra, Walter Enrique

Ferramentas para a Simulação de Falhas Transientes / Walter Enrique Calienes Bartra. – Porto Alegre: PPGC da UFRGS, 2011.

111 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Microeletrônica, Porto Alegre, BR-RS, 2011. Orientador: Ricardo Augusto da Luz Reis; Co-orientador: Fernanda G. de Lima Kastensmidt.

1. Confiabilidade. 2. Simulação. 3. Falhas. 4. Circuitos Integrados. 5. LabVIEW. I. da Luz Reis, Ricardo Augusto. II. G. de Lima Kastensmidt, Fernanda. III. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Pós-Graduação: Prof. Aldo Bolten Lucion

Diretor do Instituto de Informática: Prof. Flávio Rech Wagner

Coordenador do PGMICRO: Prof. Ricardo Reis

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“Para os crentes, Deus é o começo.
Para os cientistas, é o fim de todas as suas reflexões.”*
— MAX PLANK

AGRADECIMENTOS

Agradeço no Conselho Nacional de Pesquisa CNPq pelo apoio econômico outorgado a mim durante a estada no mestrado. Agradeço aos meus colegas do laboratório 67 – 219, em especial a Jerson Paulo Guex, Gracieli Posser, Calebe Micael, Tiago Reimann e Guilherme Augusto Flach pela amizade, ajuda, colaboração e incentivo. Além de agradecer ao professor Marcelo Soares Lubaszewski e Sergio Bampi pelas idéias para melhorias futuras, a professora Fernanda Gusmão Lima de Kastermidt por sugerir-me este tema e o Ricardo Reis por sua orientação para realizar esta dissertação.

Agradeço a meus pais pela educação brindada toda sua vida. Agradeço também a Mario Raffo Jara, que convenci-me de vim a fazer mestrado no Brasil, a Jorge Tonfat Seclen por recebi-me e acomodar-me na República de Rubens, a o seu Rubens Simão Pra dono da Republica, e a David Cordova Vivas por explicar-me o básico do funcionamento dos Osciladores Controlados por Voltagem. Além de agradecer aos irmãos John e Elvis Inga, a Jaqueline Valverde, Frank Guzman, Omaira Sierra, Raúl Chipana, Raquel Dias, Rogerio Agnesi, Romerio Kunrath, Patricia Cunha, e a todos aqueles que fizeram de minha estada em Brasil mais agradável.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	9
LISTA DE FIGURAS	11
LISTA DE TABELAS	15
RESUMO	17
ABSTRACT	19
1 INTRODUÇÃO	21
1.1 Por que LabVIEW?	23
1.2 Simulação de Falhas	24
1.3 Objetivos do Trabalho	25
1.4 Motivação	26
1.5 Organização do Trabalho	26
2 ESTADO DA ARTE DO MODELAMENTO, INJEÇÃO E SIMULAÇÃO DE FALHAS EM CIS	27
2.1 Origem das Falhas Transientes em Circuitos Integrados	27
2.2 Injeção, Modelagem e Simulação das Falhas Transientes nos CIS	28
2.2.1 Nível Elétrico	28
2.2.2 Nível de Portas Lógicas	33
2.2.3 Nível do Sistemas	35
2.3 Outros Efeitos da Radiação em CIS	37
2.3.1 Deslocamento da Voltagem de Limiar	38
2.3.2 Degradação da Mobilidade	40
2.3.3 Variação da Voltagem Sub-Limiar	40
2.3.4 Correntes de Fuga	41
2.4 Métodos de Tolerância a Falhas	42
2.4.1 Registradores Triplicados	42
2.4.2 Códigos de Correção de Erros	43
2.4.3 Endurecimento de Transistores e Células SRAM	44
2.4.4 Mascaramento de Erros	44
2.5 Falhas Transientes em Circuitos Analógicos	45

3	FERRAMENTAS DESENVOLVIDAS	49
3.1	Simulação de Portas Lógicas	49
3.1.1	Simulação do Transistor para Portas Lógicas	52
3.1.2	Histerese, Carga Capacitiva e Faixa Proibida	54
3.1.3	Portas Lógicas	55
3.1.4	Geradores de Falhas	59
3.1.5	Geradores de Dados e Sinais de Relógio	62
3.2	Simulação em Lógica Booleana	67
3.2.1	Registradores	68
3.2.2	Outras Funções	69
3.3	Simulação Analógica	70
3.3.1	Simulação do Transistor	71
3.3.2	Circuitos Úteis	72
4	SIMULAÇÕES COM AS FERRAMENTAS DESENVOLVIDAS E RESULTADOS	79
4.1	Simulação de um Inversor com SET	79
4.2	Simulação de um Inversor com Múltiplos SETs	80
4.3	Simulação de Circuito Lógico com SET	83
4.4	Simulação do Bit-Flip em uma Célula de Memória Triplicada	84
4.5	Simulação do Bit-Flip em uma Célula de Memória com Redundância Triplicada no Tempo	89
4.6	Simulação de um VCO afetado por um Pulso LASER	92
5	CONCLUSÕES	97
5.1	Trabalho Futuro	98
ANEXO A	NETLISTS PARA HSPICE	99
A.1	Biblioteca lib_250nmEE141.lib da Berkeley University	99
A.2	Netlist Test.sp para a simulação de uma porta Inversora com SET no HSPICE	102
ANEXO B	O CRITERIO DE BARKHAUSEN	103
B.1	Criterio de Oscilação	103
B.2	Analise das Condições de Oscilação	104
REFERÊNCIAS		105

LISTA DE ABREVIATURAS E SIGLAS

ASEE	Analog Single-Event Effects
BD	Block Diagram (Diagrama de Blocos)
CI	Circuito Integrado
CPLD	Complex Programmable Logic Device
DUT	Device Under Test (Dispositivo Baixo Prova)
EDAC	Error Detection and Correction
ELDRS	Enhanced Low Dose Radiation Sensitivity
FOX	Field Oxide (Oxido do Campo)
FP	Front Panel (Painel Frontal)
FPGA	Field Programmable Gate Array
GND	Tensão de referência do circuito ou Terra
HDL	Hardware Description Language (Linguagem de Descrição de Hardware)
HW	Hardware
HWIFI	Hardware Implemented Fault Injection
LET	Linear Energy Transfer (Transferência Linear de Energia)
NLC	Non-Linear Counter (Contador Não Linear)
RAM	Random Access Memory (Memória de Acesso Aleatório)
SEB	Single-Event Burnout
SEE	Single-Event Effects
SEGR	Single-Event Gate Rupture
SHE	Single Hard Error
SEL	Single-Event Latch-up
SET	Single-Event Transient
SEU	Single-Event Upset
SOC	System on Chip
SOI	Silicon On Insulator

STI	Shallow Trench Isolation
SWIFI	Software Implemented Fault Injection
TAT	Trap Assisted Tunneling
VCO	Voltage Controlled Oscillator (Oscilador Controlado por Voltagem)
VDD	Tensão de alimentação do circuito
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
VI	Virtual Instrument (Instrumento Virtual)

LISTA DE FIGURAS

1.1	Diagrama do Gajski-Kuhn indicando os Níveis de Inserção de Falhas. Estes dados foram adaptados de (ANGHEL et al., 2007)	22
1.2	Partes de um Instrumento Virtual (VI)	25
2.1	Partícula Alfa de energia $E_{k\alpha}$ batendo em uma superfície do Silício n-p inversamente polarizada a um angulo de impacto θ	29
2.2	Simulação de um transiente de corrente. A carga equivalente deste é 31.5fC aproximadamente.	30
2.3	Simulação a nível SPICE do mecanismo de deposição de cargas . . .	31
2.4	Tipos de Impactos de pulsos transientes nos transistores de portas lógicas.	33
2.5	Modelagem de Pulsos Transientes a Nível de Sistema	36
2.6	Deslocamento da tensão e mudança da inclinação sub-limiar para transistores NMOS e PMOS são apresentadas junto com seus curvas antes da irradiação	39
2.7	Mobilidade na camada de Inversão v.s. Dose total de um MOSFET irradiado	41
2.8	Transistor com oxido do campo (FOX) como isolante entre transistores.	41
2.9	Transistores Parasitas induzidos pela TID sob um transistor com FOX.	42
2.10	Técnicas de Redundância Modular Triplicada (TMR).	43
2.11	Tipos de Transistores resistentes ao TID	44
2.12	Tipos de Mascaramentos de Transientes em CIs Digitais	45
2.13	Instrumentação típica de testes para ASEE	46
3.1	VI do Transistor MOSFET.vi.	52
3.2	Parâmetros Geométricos do Transistor MOSFET	53
3.3	Hysteresis.vi	54
3.4	Logic Level.vi	55
3.5	Capacitive Load.vi	55
3.6	BD do VI Capacitive Load.vi	56
3.7	VIs das Portas Lógicas simuladas.	56
3.8	BD do VI NOR.vi.	57
3.9	Circuito da porta NOR CMOS.	58
3.10	Equivalência da porta NOR CMOS em resistências quando é aplicada a seus entradas distintos estímulos válidos.	58
3.11	Circuito Equivalente da porta NOR CMOS com o pior caso de falha. .	59
3.12	SET.vi	60
3.13	FP do SET.vi	60

3.14	SETxn.vi	61
3.15	FP do SETxn.vi	61
3.16	RndSETxn.vi	62
3.17	FP do RndSETxn.vi	62
3.18	Clock Generator.vi	63
3.19	Complementary Clock Generator.vi	63
3.20	Delta t Clock Generator.vi	64
3.21	Sinais geradas pelo VI Delta t Clock Generator.vi para $f = 100\text{Hz}$	64
3.22	Forma de onda de um sinal de relógio real.	65
3.23	Real Clock Generator.vi	65
3.24	Efeito da entrada booleana Init (F='0') no Instrumento Virtual Real Clock Generator.vi	66
3.25	Efeito de Zeros na esquerda de um dado '10' decimal no VIdigital Pattern.vi	66
3.26	Digital Pattern.vi	67
3.27	Registadores de Deslocamento em uma estrutura <i>FOR</i>	67
3.28	Incomplete Register.vi	69
3.29	Incomplete Bus Register.vi	69
3.30	Random Digital Pattern.vi	69
3.31	Bit Flip.vi	70
3.32	Voter.vi	70
3.33	Instrumento Virtual Transistor MOS PtByPt.vi	72
3.34	Tank PtByPt.vi	73
3.35	NMOS Current Mirror PtByPt.vi	74
3.36	NMOS Cross Pair PtByPt.vi	75
3.37	BD do VIdigital NMOS Cross Pair PtByPt.vi	76
3.38	Configurações do Resistores Ativos com transistores MOS	76
3.39	VIs do Resistores Ativos	77
3.40	Capacitive Load PtByPt.vi	77
4.1	Circuito Inversor com SET.	79
4.2	BD da Simulação do Circuito Inversor com SET.	80
4.3	Simulação no LabVIEW do Inversor com SET	80
4.4	Comparação entre a Simulação no LabVIEW e a Simulação com HS- PICE	81
4.5	Comparação entre a Simulação no LabVIEW e a Simulação com HS- PICE a maior escala do pulso resultante na saída do circuito.	81
4.6	BD da Simulação de um Circuito Inversor com Múltiplos SETs.	81
4.7	Pulsos SET gerados pelo VI RndSETxn.vi para esta simulação.	82
4.8	Simulação no LabVIEW do Inversor com Múltiplos SET e relógio ideal	82
4.9	BD da Simulação de um Circuito Inversor com Múltiplos SETs e relógio real.	83
4.10	Simulação no LabVIEW do Inversor com Múltiplos SETs com um relógio real de $f = 10\text{MHz}$ com $t_r = t_f = 5\text{ns}$ e $T_h = 45\text{ns}$	83
4.11	Simulação no LabVIEW do Inversor com Múltiplos SETs com um relógio real de $f = 10\text{MHz}$ com $t_r = t_f = 1\text{ns}$ e $T_h = 49\text{ns}$	84
4.12	Circuito Combinacional afetado pelo SET.	84

4.13	BD da Simulação do Circuito Combinacional da Figura 4.12.	85
4.14	FP da simulação do Circuito Combinacional da figura 4.12.	85
4.15	Circuito TMR afeitado pelo <i>Bit-Flip</i>	86
4.16	<i>BD</i> da Simulação do Circuito da figura 4.15.	86
4.17	FP da simulação do TMR à $f_{CLK} = 10\text{MHz}$	86
4.18	Circuito TMR afeitado por dois <i>Bit-Flip</i>	87
4.19	BD da Simulação do Circuito da figura 4.18 com Registradores afe- tados pela mesma fonte de <i>Bit-Flip</i>	87
4.20	FP da simulação do TMR à $f_{CLK} = 10\text{MHz}$ do BD da figura 4.19. . .	88
4.21	Resultados comparativos para diferentes frequências do Relógio. . . .	88
4.22	BD da Simulação do Circuito da figura 4.18 com dois fontes de <i>Bit- Flip</i> diferentes.	88
4.23	FP da simulação do TMR à $f_{CLK} = 10\text{MHz}$ do BD da figura 4.22. . .	89
4.24	Resultados comparativos para diferentes frequências do Relógio. . . .	89
4.25	Circuito TMR com Redundância Temporal afetado pelo <i>Bit-Flip</i> . . .	90
4.26	BD da Simulação do Circuito da figura 4.25.	90
4.27	FP da simulação do TMR Com Redundância Temporal à $\Delta t = 90^\circ$ e $f_{CLK} = 10\text{MHz}$	91
4.28	Resultados comparativos para diferentes Δt do Relógio em Graus. . .	91
4.29	Pulso LASER Simulado. Sua carga equivalente é de 37, 125pC. . . .	92
4.30	Esquemático do Circuito VCO. O raio indica o transistor ao qual se inserira a falha descrita na figura 4.29.	93
4.31	Diagrama de Blocos da Simulação de um VCO afeitado por um pulso LASER.	93
4.32	Simulação de um VCO afeitado por um pulso LASER.	94
4.33	Resultados obtidos por (CHEN et al., 2003).	94
4.34	Assinatura de Amortecimento do VCO.	94
4.35	Varrimento 3D da Assinatura de Amortecimento do VCO.	95
4.36	THD+N da Assinatura de Amortecimento do VCO.	95
B.1	Diagrama de Blocos de um circuito com Realimentação Positiva. . .	103
B.2	Quebra do laço de realimentação para calcular o ganho do laço. . . .	104

LISTA DE TABELAS

2.1	Principais fontes de radiação natural do espaço	28
2.2	Principais Radio-Isótopos usados para produzir falhas nos CIs.	33
2.3	Íons típicos usados durante os Experimentos de SEEs em Circuitos Analógicos	47
3.1	Resumo de Funções Lógicas e Funções Úteis do pacote de Simulação de Portas Lógicas.	50
3.2	Resumo de Geradores de Funções Lógicas do pacote de Simulação de Portas Lógicas.	51
3.3	Parâmetros Físicos da Tecnologia de 250nm para os transístores de Portas Lógicas.	52
3.4	Larguras das portas dos Transistores Simulados para cada VI criado. .	57
3.5	Equações para V_{out} e R_{eq} para modelar as portas desenvolvidas com falhas SEU/SET.	59
3.6	Resumo de Funções do pacote de Simulação em Lógica Booleana. . .	68
3.7	Resumo de Funções do pacote de Simulação em Lógica Booleana. . .	71
3.8	Parâmetros Físicos da Tecnologia de 250nm para os transistores de Simulação Analógica	72

RESUMO

Atualmente, a simulação de falhas é um estágio importante em qualquer desenvolvimento de Circuitos Integrados. A predição de falhas comportamentais em qualquer estágio do processo é essencial para garantir que o chip desenvolvido seja bem implementado. Vários problemas podem ser conferidos e solucionados enquanto se executa a simulação.

As falhas transientes mais conhecidas são os *Single-Event-Upset* (SEU), as quais acontecem nos circuitos de memória, e as *Single-Event Transient* (SET), que acontecem em circuitos de lógica combinacional. A análise do comportamento do circuito sob falhas é fundamental para a escolha de técnicas de proteção e medição da susceptibilidade aos diferentes tipos de falhas.

Neste trabalho, apresenta-se uma ferramenta para simular os efeitos que acontecem quando uma fonte de falha é inserida num circuito digital, especialmente falhas SEU. Além disso, é desenvolvido o método TMR que pode verificar a existência de uma falha e inibir que esta se propague pelo circuito todo. Foram desenvolvidos módulos para simulação de circuitos analógicos como o Oscilador Controlado por Voltagem (VCO) permitindo a visualização dos efeitos de falhas nestes circuitos.

A ferramenta LabVIEW[®] da National Instruments é usada para criar o conjunto de Instrumentos Virtuais (VIs) para simular os SEUs. Esta é também usada pela simulação de SETs. Foram feitas várias simulações com as ferramentas desenvolvidas para validar sua funcionalidade os quais mostram resultados semelhantes aos descritos na literatura.

As ferramentas desenvolvidas para simulação de falhas transientes em portas lógicas inserem falhas SET de forma automática sem análise prévia do sinal de saída. Usando as ferramentas de Lógica Booleana é possível obter resultados para fazer estudos estatísticos dos erros acontecidos e determinar tendências no comportamento das técnicas de Redundância Modular Triplo (TMR) e TMR com redundância no tempo. O modelo desenvolvido para a análise de falhas do VCO apresenta uma melhor semelhança com o resultado real que com o simulado com ferramentas comerciais.

Palavras-chave: Confiabilidade, Simulação, Falhas, Circuitos Integrados, LabVIEW.

Transient Fault Simulation Toolkit

ABSTRACT

Nowadays, the fault simulation is an important step in any IC design. Predicting the behavioral faults of any process step is essential to ensure that the design is well implemented. During the simulation various problems can be detected and corrected.

The transient faults are the most well known Single-Event-Upset (SEU), which affect memory circuits, and Single-Event Transient (SET), which affect combinational logic circuits. The analyses of the circuit under faults is crucial to the choice of protection techniques and measurement of susceptibility to different types of failures.

In this work a tool to simulate the effects that occur when a source of fault is inserted in a digital circuit, especially SEU faults is presented. In addition to modeling a fault, it is developed a Triple Modular Redundancy (TMR) method capable of verifying the existence of a fault preventing it from spreading through the whole circuit. It is also developed a Voltage Controlled Oscillator (VCO) to view fault effects in analog circuit.

LabVIEW[®] is used to create a set of virtual instruments to simulate SEUs. It is efficient in modeling the characteristics of SETs. It is possible with this toolkit to replicate the effects of SEUs and SETs described in the literature.

The tools developed for simulation of transient faults in logic gates insert SET failures automatically without output signal prior analysis. Using the tools of Boolean Logic is possible to obtain results to make statistical studies of the errors that occurred and determine trends in the behavior of TMR with and without redundancy in time. The model developed for failure analysis of the VCO is similar to the real result with that simulated with commercial tools.

Keywords: Reliability, Simulation, Faults, Integrated Circuits, LabVIEW.

1 INTRODUÇÃO

Com as falhas acontecidas no satélite Telstar, em 1962, devido a um teste nuclear de grande altitude, foi possível conhecer alguns efeitos da radiação em dispositivos eletrônicos. O avanço da tecnologia e a diminuição dos dispositivos faz com que os circuitos ficassem mais susceptíveis a erros devido a efeitos de radiação. As fontes de radiação podem ser de origem espacial provenientes de explosões solares, faixas de Van Allen, vento solar ou raios cósmicos (VELAZCO; FOUILLAT; REIS, 2007), ou de fontes eletromagnéticas e radioativas a nível da Terra. Uma vez o circuito sendo exposto a uma fonte de radiação pode ter seu valor lógico alterado ou as características do circuito alteradas definitivamente ou mesmo inutilizando o dispositivo, dependendo da quantidade de radiação a qual o circuito foi exposto. Estes são os principais motivos pelos quais é importante e necessário prever o comportamento do circuito sob falhas.

Há diversas maneiras de fazer injeção de falhas simulando estes fenômenos em distintas partes do projeto de um circuito integrado (CI). Estas inserções de falhas são baseadas no uso de estímulos tanto reais como simulados. Estas podem ser feitas a Nível de Dispositivo, a Nível Elétrico ou a Nível de Porta Lógica (ANGHEL et al., 2007), dependendo em que etapa do projeto é desenvolvido o teste e do nível de abstração considerado. Na figura 1.1 é mostrado o diagrama “Y” de Gajski-Kuhn (GEREZ, 1999). Este diagrama mostra os três eixos do domínio de projeto de circuitos de altíssima escala de integração (VLSI) e seus distintos níveis de abstração. Neste esquema estão marcados os níveis do projeto VLSI onde normalmente são inseridas as falhas para estudo.

Uma perturbação feita a Nível de Dispositivo precisa de instalações adequadas para sua realização, devido à necessidade de equipamento Radiológico ou Laser de alta precisão para poder inserir falhas reais nos dispositivos sob teste (DUT). Para poder fazer isso também pode-se precisar equipamento mecânico adicional, de aquisição de dados, etc. Estas perturbações também podem ser simuladas nos computadores.

Em uma inserção de falhas feito a Nível de Porta Lógica é preciso modificar a descrição do hardware (HW) a ser testado. Esta descrição usa alguma Linguagem de Descrição de Hardware (HDL) como Verilog ou VHDL que rode em alguma plataforma industrial ou científica feita com dispositivos FPGA ou CPLD. Também é necessário o software que faz a inserção do *Bit-Stream* ou configuração do dispositivo.

Uma inserção de falhas a Nível Elétrico é baseada em simuladores elétricos rodados em computadores que inserem as falhas nos transistores para ver o comportamento destes ante estes estímulos. Estas falhas são descritas mediante uma descrição do circuito muito mais detalhada e profunda que a descrição para as inserções de falhas a Nível de Portas Lógicas.

As falhas por radiação podem causar algum dos seguintes fenômenos dentro dos CI e sistemas que contem (BOUDENOT, 2007): Efeitos de Dose de Ionização, Efeitos de

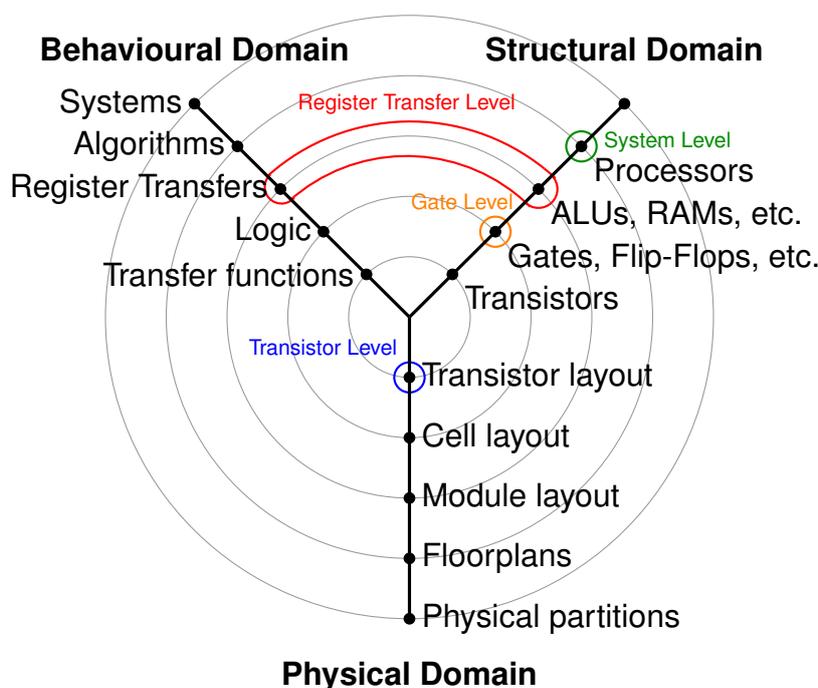


Figura 1.1: Diagrama do Gajski-Kuhn indicando os Níveis de Inserção de Falhas. Estes dados foram adaptados de (ANGHEL et al., 2007)

Deslocamento, Efeitos de Íons Pesados e Efeitos de Prótons. Cada um causa distintos fenômenos dentro dos CIs em diferentes níveis e dependem da natureza da partícula, de sua Energia Cinética e da Transferência Linear de Energia (LET) que possui.

Uma boa forma de caracterizar um ambiente de inserção de falhas é considerar a classificação “FARM” (ANGHEL et al., 2007), a qual possui os seguintes atributos:

- F: O conjunto de falhas (*faults*) que serão inseridas de propósito no sistema.
- A: O conjunto de modos de ativação (*activation trajectories*) que especificam o domínio usado para o funcionamento do sistema.
- R: O conjunto de saídas de leitura (*readouts*) que correspondem ao comportamento do sistema.
- M: O conjunto de medições (*measures*) que correspondem às medições dependentes através da inserção de falhas.

Pode-se melhorar ainda mais este modelo agregando um conjunto de cargas de trabalho (*workloads: W*), isto é, um conjunto de estímulos nas entradas para que se estimule os recursos do sistema enquanto a inserção de falha é realizada. O modelo “FARM” é considerado como um modelo abstrato que descreve os atributos de um experimento de inserção de falhas, mas não considera a técnica adotada que se usa para inserir as falhas. O “FARM” pode ser usado para diferentes métodos de inserção de falhas, mas não considera o caráter invasivo do experimento, a velocidade e o custo do mesmo (ANGHEL et al., 2007).

Na literatura, as falhas causadas por Íons Pesados são representadas pela denominação de Single-Event Effects (SEE), dos quais os mais comuns são (BOUDENOT, 2007):

- Single-Event Transient (SET): Falha transiente em Circuitos Combinacionais.
- Single-Event Upset (SEU): Falha transiente em Circuitos Sequenciais e de Memória.
- Single-Event Latchup (SEL): Falha que pode destruir o dispositivo, afetando a estrutura CMOS.
- Single-Event Burnout (SEB): Falha destrutiva que afeta os dispositivos MOSFET de potência.
- Single-Event Gate Rupture (SEGR): Falha que afeta a estrutura *sub-micron*. É potencialmente destrutiva.
- Single Hard Error (SHE): Falha destrutiva em dispositivos de hardware complexos.

Aquelas denominadas transientes são somente as falhas SEU e SET, devido à duração no tempo destas é muito curto. O SEU é uma falha que altera um bit de um registrador ou célula de memória, por exemplo, um registrador com um valor lógico “1” passa a ter o valor lógico “0” após ser afetado e vice-versa. As falhas SEU também são conhecidas como *Soft Errors*. Já o SET afeta a funcionalidade dos transistores, criando uma corrente anômala que pode afetar o resultado final de uma porta lógica. A probabilidade de ocorrência dessas falhas muda com o aumento de falhas simultâneas em um sistema eletrônico sequencial ou combinacional (MISKOV-ZIVANOV; MARCULESCU, 2010). Também a distribuição aleatória de dopantes no silício afeta as ocorrências de *soft errors* devido às variações da voltagem de limiar (BALASUBRAMANIAN et al., 2007).

1.1 Por que LabVIEW?

Atualmente, as linguagens de programação são muitas e são para aplicações específicas, simulação ou são de propósito geral. Alguns destas linguagens podem gerar pacotes para rodar em certas aplicações ou sistemas operacionais. Um destas linguagens é LabVIEW, a qual não precisa de código escrito para fazer um programa, porque o código tem que ser desenhado como se fizesse um fluxograma. Além disto, uma vantagem do LabVIEW, o que lhe faz semelhante ao VHDL ou Verilog, é sua programação totalmente concorrente. O programa se assemelha a um circuito elétrico, onde as funções são os elementos do circuito e os cabos que conectam estas funções servem para passar os dados de uma função a outra. Este tipo de programação é conhecida como “Programação de Fluxo de Dados” (TRAVIS; KRING, 2006), um tipo de programação que contrasta com a “Programação de Fluxo de Controle” típica do linguagens como o Assembler, Basic ou C. Uma das grandes vantagens da “Programação de Fluxo de Dados” é sua natureza concorrente. Esta vantagem também permite que os programas feitos com “Programação de Fluxo de Dados” possam rodar facilmente em múltiplos processadores.

Originalmente, o LabVIEW foi criado com a ideia de ajudar a técnicos e cientistas na automação de processos industriais e laboratórios, como em (SINGH; PODDAR, 2008). A ideia surgiu da necessidade de criar uma ferramenta totalmente visual, algo que as linguagens de propósito geral e livres como o C ou C++ não fornecem. Também é de fácil e rápido aprendizagem, devido ao fato de os símbolos usados serem símbolos padrão das ciências e engenharias atuais. A depuração dos programas feitos no LabVIEW também é mais fácil, devido a não possuem erros de sintaxe e tão pouco de um estruturamento

rígido do código (JOHNSON; JENNINGS, 2006). LabVIEW também é uma linguagem totalmente hierárquica, o que faz evidente (e necessário) quando se tem programas muito grandes. Isto faz que com o LabVIEW seja possível criar aplicações muito mais rápido que com as linguagens de programação comuns (TRAVIS; KRING, 2006; MCMICKELL et al., 2010).

Esta ferramenta permite uma prototipação de ferramentas e acessórios de forma ágil e eficiente. Uma das características que esta ferramenta possui é sua capacidade de executar a programação inserida de forma concorrente, além disso, sua programação é completamente visual. Os programas feitos com *LabVIEW* se chamam Instrumentos Virtuais (VIs). Um VI contém as seguintes partes chave (JOHNSON; JENNINGS, 2006):

- *Front Panel* (FP): O Painel Frontal o Interface do Usuário. Esta parte do VI contém objetos textuais, decorações, controles e indicadores. Quando um VI é usado como um subVI (algo parecido a usar o VI como uma subrotina) o código do FP não é carregado na memória. Um exemplo é mostrado na figura 1.2a.
- *BlockDiagram* (BD): O Diagrama de Blocos. Este é o fluxograma que representa o código do VI de maneira gráfica. Na figura 1.2b é mostrado um Diagrama de Blocos.
- *Data*: O espaço de dados do VI inclui os controles e indicadores padrão do FP, as constantes e os *buffers* de memória necessários. O espaço de dados do VI sempre é carregado na memória.
- *Code*: Este é o código compilado de cada subVI que contém o VI e sempre é carregado na memória.

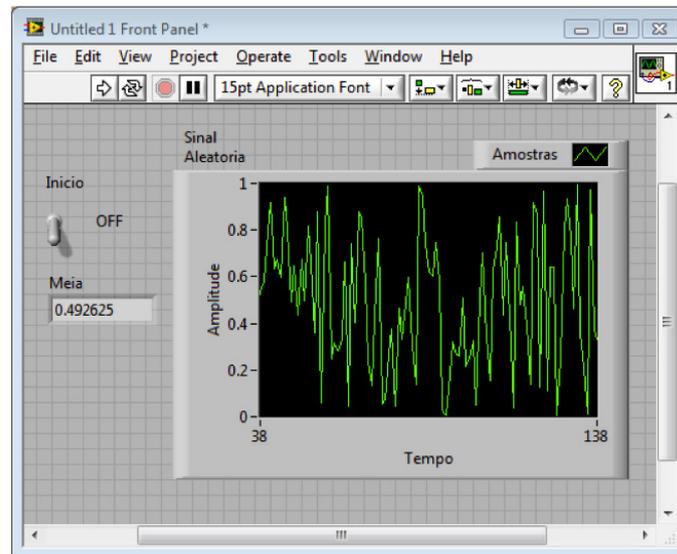
Além de possuir ferramentas de programação básica e estruturas muito semelhantes às fornecidas pelo C++, LabVIEW possui ferramentas de matemática avançada, estatística, otimização, análise, processamento de cadeias de caracteres, manejo de arquivos, gráficas 2D e 3D, processamento de imagens e sinais e conectividade com o sistema operativo (TRAVIS; KRING, 2006).

O LabVIEW também tem a capacidade de executar código externo como C, Matlab, XMath e DLLs. Isto é muito útil para quem quer fazer reuso do código escrito em LabVIEW e aproveitar suas ferramentas.

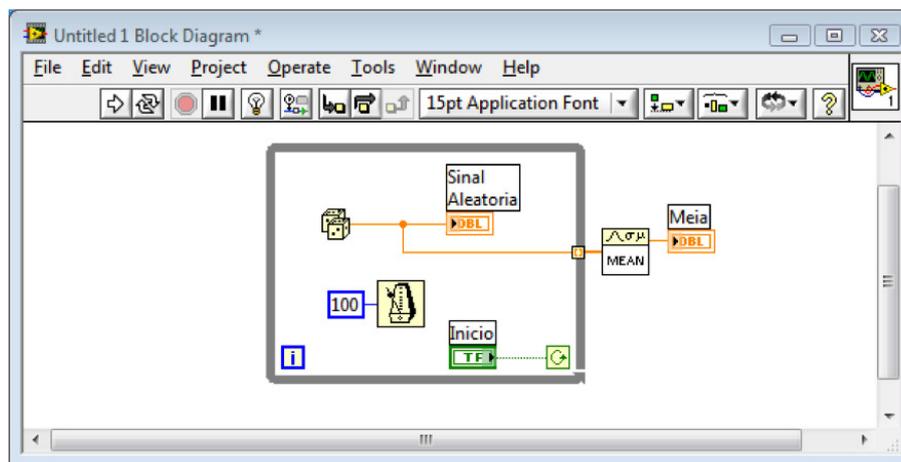
Além de tudo isso, LabVIEW tem conectividade com muitos tipos de portas de hardware externo e protocolos, como RS-232, USB, Porto Paralelo, IrDA, Bluetooth, TCP/IP, UDP, GPIB, VISA e mais. Também muitas companhias de instrumentação na indústria criam *drivers* para que seus instrumentos possam ser controlados usando LabVIEW, como é o caso de Hewlett Packard, Agilent e TekTronics.

1.2 Simulação de Falhas

O Modelamento e Simulação do efeitos da radiação ionizante tem sido largamente usado para melhorar o entendimento dos efeitos da radiação sobre CIs e sistemas. Nos últimos 25 anos, devido ao progresso em codificação e melhoria na performance dos computadores, a simulação tem alcançado um grande interesse. Devido a sua capacidade de predição, a simulação oferece a possibilidade de refazer os experimentos com radiação de um laboratório num teste de dispositivos hipotéticos ou condições, que não são possíveis



(a) Painel Frontal (FP)



(b) Diagrama de Blocos (BD)

Figura 1.2: Partes de um Instrumento Virtual (VI)

(ou de difícil medição) por experimentação (MUNTEANU; AUTRAN, 2008). As simulações numéricas baseadas em fenômenos físicos a nível de transistor são uma ferramenta indispensável para o análise destes fenômenos em todo tipo de tecnologia, onde a pesquisa experimental está claramente limitada. Neste caso, a simulação é uma ferramenta que provê de fatos físicos e prediz a operação de futuros dispositivos.

1.3 Objetivos do Trabalho

O objetivo deste trabalho é o desenvolvimento de um conjunto de ferramentas ou *tool-kit* para a simulação de falhas a nível de Portas Lógicas e Transferência de Registradores na plataforma *LabVIEW* (versão 8.20 ou superior). Este *toolkit* está organizado como uma biblioteca de VIs que pode ser facilmente distribuída e compilada para as versões posteriores a 8.20. Deve ser cuidado para que o desenvolvimento dos diagramas de blocos feitos com estas ferramentas, na medida do possível, sejam iguais aos circuitos que podem ser implementados graficamente usando programas como o *HSPICE* ou *PSPICE*. Estas ferramentas desenvolvidas não pretendem substituir a funcionalidade e potência

dos programas e ferramentas especializadas previamente mencionados, só pretendem dar uma introdução aos conceitos de falhas em CIs e dar uma visão inicial das técnicas de mitigação das mesmas.

1.4 Motivação

Após do estudo e entendimento de como funciona o LabVIEW e analisar os pacotes que inclui, conclui-se que realmente é possível produzir programas dedicados a uso exclusivo de pesquisa e desenvolvimento. Esta ferramenta possui pacotes de Controle, Análise de Ordem, etc. As empresas e laboratórios que usam extensivamente esta ferramenta desenvolvem pacotes proprietários em curto tempo para resolver problemas específicos. No campo da pesquisa, esta ferramenta é muito apreciada devido a sua capacidade de conectividade com HW e usar pacotes de análise como o IMAQ (INSTRUMENTS, 2005) para avaliar sólidos ou escolher partes de uma imagem no estudo, como é o caso de (CHOI et al., 2008) onde se faz um estudo de variabilidade do processo de 90nm usando o LabVIEW. Mas ainda ninguém fez um pacote, nem proprietário nem acadêmico, que possa fazer simulação de falhas transientes em CIs para esta plataforma, algo necessário para a indústria e pesquisa neste campo, o que está sendo necessário devido ao escalonamento da tecnologia.

1.5 Organização do Trabalho

O capítulo 2 mostra o panorama de modelamento, injeção e simulação de falhas transientes atual, explicando este desde os três níveis de abstração da injeção de falhas: nível elétrico, nível de portas lógicas e nível sistema. No capítulo 3 será explicado em forma detalhada as funcionalidades do pacote desenvolvido, indicando todos seus componentes, BDs e detalhes destes. No capítulo 4 mostram-se alguns resultados e considerações obtidos durante as simulações utilizando as ferramentas desenvolvidas. No capítulo 5 apresenta as conclusões e trabalhos futuros.

2 ESTADO DA ARTE DO MODELAMENTO, INJEÇÃO E SIMULAÇÃO DE FALHAS EM CIS

Nesta seção se mostram os temas que serviram para o desenvolvimento deste trabalho de dissertação de mestrado. De todos estes temas vários deles foram desenvolvidos. Os demais são deixados para futuras pesquisas e desenvolvimentos na mesma área.

2.1 Origem das Falhas Transientes em Circuitos Integrados

Durante a Guerra Fria, os testes nucleares se fizeram muito mais comuns devido a dois motivos: o armamentismo das grandes potências e os estudos dos efeitos nos seres vivos e infra-estruturas devido a estas novas armas. Paralelamente acontecia a Corrida Espacial entre os Estados Unidos e a extinta União Soviética. Na data 10 de julho de 1962 foi lançado o Telstar, um satélite de comunicações desenhado pela Bell Telephone Laboratories em conjunto com a AT&T e suportado pela NASA. Este satélite foi o primeiro a ser equipado com *transponders* de sinal para transmissão de Televisão em vivo, convertendo a este no primeiro satélite de comunicações ativo. Um dia antes do lançamento, os Estados Unidos realizou um teste nuclear a grande altitude. Os altos níveis de radiação induzida pelos elétrons injetados nos cinto de Radiação de Van Allen causou degradação em certos componentes do Telstar. Finalmente, o satélite foi perdido em 21 de fevereiro do 1963. Este foi o primeiro incidente conhecido de um satélite perdido devido aos efeitos da radiação (ECOFFET, 2007).

As falhas transientes, as quais são só um dos tantos tipos de falhas causadas pela radiação, são falhas que só perduram uma pequena parte do tempo. O efeito de radiação em circuitos integrados podem ser desde ruídos de baixa magnitude até pulsos tão grandes que podem inclusive, danificar permanentemente um equipamento eletrônico, em questão de segundos. Também existem os efeitos cumulativos que faz que os equipos falhem de maneira gradual até sua completa perda. Estas fontes de falhas podem ser classificadas em dois grandes grupos: Aquelas de origem natural e aquelas de origem artificial ou causadas pelo homem. Mas pode-se fazer uma classificação mais adequada baseando-se no local do origem das radiações, deste modo pode-se classificar as fontes de radiação que danificam os CIs em aquelas do Espaço e aquelas provenientes da Terra. Na tabela 2.1 relata-se as principais fontes de radiação de origem espacial que danificam os CIs.

Em terra, os efeitos de radiação foram observados pela primeira vez na Intel Corp. no ano 1978 enquanto eles desenvolviam as primeiras memórias de acesso aleatório dinâmico (DRAMs). Se percebeu que estas ligavam sozinhas devido ao conteúdo de traços de radioisótopos de ${}_{84}\text{Po}^{210}$ (um emissor alfa, produto da desintegração do ${}_{82}\text{Pb}^{210}$) nos componentes dos encapsulamentos das memórias. O pessoal da Intel resolveu chamar

Fonte	Particula	Energia Típica Associada
Faixas de Radiação Van Allen	Eléctrons	1eV - 10MeV
	Protons	1keV - 500MeV
Explosões Solares	Protons	1keV - 500MeV
	Ions	1 a varios 10MeV/n
Raios Cosmicos	Protons e Ions	Fluxo máximo perto de 300MeV/n

Tabela 2.1: Principais fontes de radiação natural do espaço (ECOFFET, 2007).

este fenômeno de *single-event effect* (SEE) pois acontecia somente quando uma única partícula alfa batia sobre uma célula da DRAM. A denominação do SEE foi finalmente formalizada na norma JEDEC JESD89A (MUNTEANU; AUTRAN, 2008). Outros efeitos devido aos nêutrons foram percebidas durante o uso do Boro para o implante de dopantes pois este contem uma quantidade de quase 20% em abundância de ${}^5\text{B}^{10}$, o qual quando captura um nêutron, este se desintegra em ${}^3\text{Li}^7$ mais uma partícula alfa, a qual pode causar um SEE (BALASUBRAMANIAN et al., 2007). Em geral, é possível encontrar impurezas de ${}_{92}\text{U}^{238}$, ${}_{92}\text{U}^{234}$ e ${}_{90}\text{Th}^{232}$ em outros materiais usados na fabricação de CIs (WROBEL et al., 2009).

Todo o exposto até agora, combinando o fato de escalabilidade da tecnologia na fabricação do transistores e ao modelagem estatístico em tecnologias *sub-micron*, faz que um íon com baixa energia cinética possa produzir uma falha transiente que pode-se espalhar por tudo o CI (MISKOV-ZIVANOV; MARCULESCU, 2010).

2.2 Injeção, Modelagem e Simulação das Falhas Transientes nos CIs

A modelagem e simulação de efeitos de ionização devido a radiação é usado para ter um melhor entendimento destes efeitos sobre o funcionamento de dispositivos e circuitos eletrônicos. Utiliza-se de mecanismos de injeção que ajudam a fazer estas simulações possíveis. Nos últimos anos, devido aos avanços em codificação para simulação e melhora na velocidade dos computadores a simulação tem tido muito interesse no campo científico.

A injeção, modelagem e simulação das falhas transientes podem ser feitas nos três domínios do desenvolvimento tecnológico, mas não em todos os níveis, como é mostrado na figura 1.1. Para este caso, a injeção e simulação pode ser feita a Nível Elétrico, a nível de Portas Lógicas ou de Transferencia de Registradores (RTL) e a nível de Dispositivo ou Sistema.

2.2.1 Nível Elétrico

Este nível, também conhecido como Nível de Transistor, só cobre o nível do leiaute do transistor do domínio físico do diagrama Gajski-Kuhn da figura 1.1. Este é o nível de abstração mais baixo da modelagem e inserção de falhas.

2.2.1.1 Modelagem Físico de Falhas Transientes

O meio é composto por muitas partículas geradas pela atividade tanto solar, cósmica ou terrestre (MESSENGER, 1982; GUSMÃO KASTENSMIDT; CARRO; REIS, 2006). Estas partículas podem ser de dois tipos: partículas carregadas (como os elétrons, prótons e íons pesados) e radiação eletromagnética (formada por fótons como raios X ou Gamma). Quando uma destas partículas carregadas atravessa o silício de um chip, esta perde energia

devido à produção de pares elétron-lacuna. Os prótons e nêutrons também podem causar reações nucleares e ionizar o silício da mesma maneira. Este efeito ioniza o silício no trajeto percorrido pela partícula carregada que é ilustrado na figura 2.1. Em resumo, pode-se descrever o mecanismo básico de um transiente devido a uma partícula, em três estágios no tempo: (1) Deposição de Carrega, (2) Transporte de carga e (3) Coleção de carga (MUNTEANU; AUTRAN, 2008). De acordo com (ALEXANDER, 2003), estes fenômenos são devido a geração de foto-correntes no silício quando a radiação bate sobre ele.

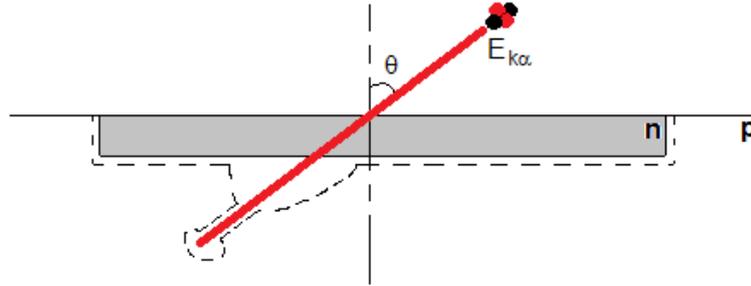


Figura 2.1: Partícula Alfa de energia $E_{k\alpha}$ batendo em uma superfície do Silício n-p inversamente polarizada a um ângulo de impacto θ . A linha pontilhada representa o campo elétrico da união metalúrgica.

As cargas criadas pelo impacto destas partículas variam dependendo do tipo de partícula, do ângulo de incidência e do local de impacto (MESSENGER, 1982). Estas cargas causam uma corrente adicional transiente $I_p(t)$ e uma carga anômala Q_p na estrutura do silício. A modelagem dessa corrente transitória é descrito pelas seguintes equações:

$$I_p(t) = I_0 (e^{-t/\tau_F} - e^{-t/\tau_R}) \quad (2.1a)$$

$$Q_p = \int_0^{\infty} I_p(t) dt \quad (2.1b)$$

onde I_0 é a máxima corrente entregue pelas cargas, τ_R é a constante de tempo de coleção de cargas na junção e τ_F é a constante de tempo para estabelecer a trilha do íon pesado. Na figura 2.2 mostra a corrente $I_p(t)$ para $I_0 = 350\mu A$, $\tau_R = 10ps$ e $\tau_F = 100ps$. Os termos da equação (2.1a), de acordo com (MESSENGER, 1982) podem ser desenvolvidos da seguinte maneira:

$$I_0(t) = q \bar{\mu} N E_0 \sec \theta \quad (2.2a)$$

$$\tau_F = \left[\bar{\mu} \frac{dE}{dX} \right]^{-1} \quad (2.2b)$$

onde $q = 1.602 \times 10^{-19}C$ é a carga do elétron, $\bar{\mu}$ é a mobilidade meia que depende do campo elétrico E (que depende além da concentração de dopantes N_D e N_A e da temperatura T (KANO, 1998)), N é a densidade linear do pares elétron-lacuna (em cm^{-1}), E_0 é o campo elétrico inicial na junção n-p (posição $X = 0$), θ é o ângulo de impacto da partícula e dE/dX é o gradiente do campo elétrico respeito da posição. A densidade linear dos pares elétron-lacuna N depende da Transferencia Linear de Energia Absoluta da partícula (LET) dada em MeV/cm . O LET depende da energia da partícula $E_{k\alpha}$:

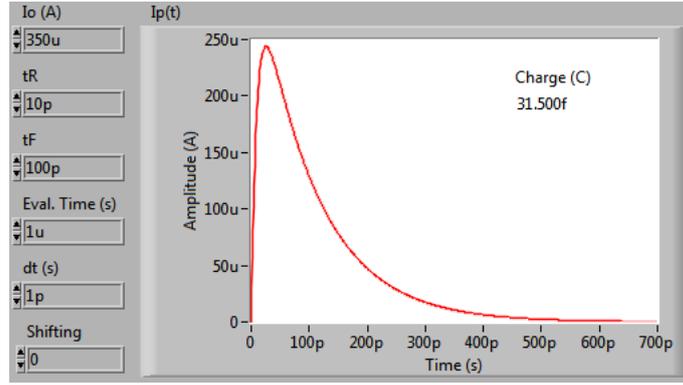


Figura 2.2: Simulação de um transiente de corrente. A carga equivalente deste é 31.5fC aproximadamente.

$$N = \frac{\text{LET}}{E_{\text{gSi}}} \quad (2.3)$$

onde $E_{\text{gSi}} = 3.6\text{eV}$ é a energia necessária para criar um par elétron-lacuna no silício. Define-se Transferencia Linear de Energia Relativa ao material de uma partícula dada em $\text{MeV} \cdot \text{cm}^2/\text{mg}$ como:

$$\text{LET}_M = \frac{\text{LET}}{\rho_M} \quad (2.4)$$

onde ρ_M é a densidade volumétrica do material que a partícula ou radiação vai atravessar e tem por unidades mg/cm^3 . Em particular, a densidade do silício é $\rho_{\text{Si}} = 2329 \text{mg}/\text{cm}^3$. É possível aproximar a densidade linear de carga depositada no material pelo transiente com o LET relativo: $1\text{pC}/\mu\text{m} \cong 100\text{MeV} \cdot \text{cm}^2/\text{mg}$ no Silício (MUNTEANU; AUTRAN, 2008).

Além da corrente transiente gerada, pode-se observar o efeito do canalizado (*funneling*), que consiste na deformação do campo elétrico da união metalúrgica inversamente polarizada da figura 2.1. Este fenômeno foi estudado em (HSIEH; MURLEY; O'BRIEN, 1981a) onde o campo elétrico e a tensão das linhas de força do campo diminui com a profundidade. A modelagem mostrada em (MESSENGER, 1982) demonstra que este canalizado do campo elétrico é baseado em uma integral elíptica de primeira espécie definida. Os estudos feitos por (SHANFIELD et al., 1987) sob dispositivos de Silício e Arseniato de Gálio demonstram que os processos dinâmicos de coleção de cargas (HSIEH; MURLEY; O'BRIEN, 1981b) acontecidos durante o canalizado ainda não podem ser modelados e que o canalizado também depende do ângulo de incidência da partícula. Mas foi desenvolvido um modelo mais simples para a representação do campo elétrico durante o canalizado em (MUSSEAU, 1991), o qual pode-se modelar da seguinte forma em função da longitude de trilha t_z percorrida pela partícula dentro do silício:

$$E(t_z) = E_0 e^{-\frac{qN_A M}{\varepsilon_{\text{Si}}} t_z} \quad (2.5)$$

onde ε_{Si} é a permitividade do silício, $M = \mu_n \mu_p / (\mu_n + \mu_p)$, μ_n é a mobilidade dos doadores e μ_p é a mobilidade dos aceitadores.

Com os dados coletados de todas as experiências, foram desenvolvidas técnicas de predição destas falhas, como é o caso de (SRINIVASAN; TANG; MURLEY, 1994) que

desenvolveram um método de modelagem preditivo das falhas ocasionadas por hadrones contidos nos raios cósmicos baseado no Método Montecarlo.

2.2.1.2 Modelagem e Injeção de Falhas Transientes no SPICE

O SPICE é o simulador elétrico mais usado para simular circuitos eletrônicos e esta baseado no Algoritmo de Nodo de Corrente ou Análise Nodal (HO; RUEHLI; BRENNAN, 1975). Este algoritmo converte o circuito descrito em texto em uma matriz e resolve uma equação linear que é criada a partir da lei de Correntes de Kirchoff.

Para o caso da simulação de SETs num circuito descrito no SPICE, deve-se usar o análise transitório para poder obter resultado ao longo do tempo de simulação (NENZI; VOGT, 2010; SYNOPSIS, 2005) e inserir uma fonte de corrente transiente tipo exponencial aos transistores afetados. Na figura 2.3 pode-se ver esta modelagem a nível SPICE do SET sobre o transistor PMOS de uma porta Inversora (WIRTH; VIEIRA; KASTENSMIDT, 2007). Este método tem o inconveniente de demandar a escrita dos transientes um a um no *netlist* SPICE, além de que é só serve para inserir SETs só nas redes *pull-up* das porta CMOS e quando a o sinal de saída está no nível baixo para o caso da figura 2.3.

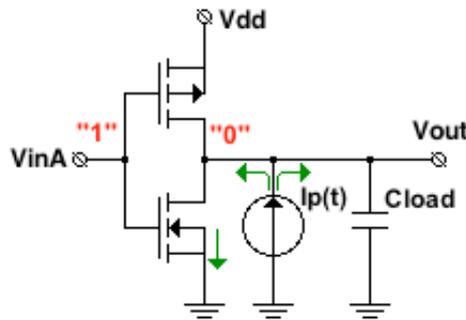


Figura 2.3: Simulação a nível SPICE do mecanismo de deposição de cargas. Esta é modelada usando uma fonte transiente baseada na equação (2.1a).

Para este caso é possível saber a tensão na saída do inversor $V_{out} = V(t)$ da figura 2.3, assumindo a R_n como a resistência do transistor NMOS quando está ligado, que τ_R é muito pequeno e que a carga capacitiva é $C_{load} = C_L$:

$$V(t) = \frac{I_0 \tau_F R_n}{\tau_F - R_n C_L} (e^{-t/\tau_F} - e^{-t/R_n C_L}) \quad (2.6)$$

O tempo onde a tensão $V(t)$ chega até o máximo valor t_{peak} é possível de ser obtido derivando a equação (2.6) e igualando-a zero:

$$t_{peak} = \frac{\tau_F R_n C_L \ln(\tau_F / R_n C_L)}{\tau_F - R_n C_L} \quad (2.7)$$

Substituindo (2.7) na equação (2.6) obtém-se o valor da tensão de pico na saída do circuito:

$$V_{peak} = \frac{I_0 \tau_F R_n}{\tau_F - R_n C_L} \left(\left(\frac{\tau_F}{R_n C_L} \right)^{R_n C_L / (R_n C_L - \tau_F)} - \left(\frac{\tau_F}{R_n C_L} \right)^{\tau_F / (R_n C_L - \tau_F)} \right) \quad (2.8)$$

Então, a mínima carga injetada Q_C ($Q_C = I_0 \tau_F$) para obter um pulso de tensão transiente de valor máximo V_{peak} é de:

$$Q_C = \frac{V_{peak}(\tau_F - R_n C_L)}{R_n [(\tau_F/R_n C_L)^{R_n C_L/(R_n C_L - \tau_F)} - (\tau_F/R_n C_L)^{\tau_F/(R_n C_L - \tau_F)}]} \quad (2.9)$$

Olhando o comportamento assintótico de (2.9) o modelo pode ser simplificado (WIRTH; VIEIRA; KASTENSMIDT, 2007) ainda mais:

$$Q_C = V_{peak} C_L + (V_{peak}/R_n) \tau_F \quad (2.10a)$$

$$T_D = t_{peak} - R_n C_L \ln \left(\frac{V_{dd}/2}{V_{peak}} \right) - \tau_F \ln \left(\frac{V_{dd}/2}{V_{peak}} \right) \quad (2.10b)$$

$$V_{peak} = \frac{I_0 \tau_F R_n}{\tau_F + R_n C_L} \quad (2.10c)$$

onde T_D é o tempo de duração do pulso transiente. A equação (2.10c) é mais eficiente computacionalmente que a equação (2.8). Este modelo de V_{peak} é mais simples que o modelo de V_{max} mostrado por (DAHLGREN; LIDEN, 1995).

Atualmente, devido ao avanço tecnológico, a modelagem e injeção de falhas transientes no SPICE deve se ser mudado. Devido a mudança de comportamento do pulso transiente em tecnologias por embaixo dos 90nm, este já não pode ser modelado com a equação (2.1a). Agora, em vez de usar uma fonte exponencial deve-se usar uma fonte linear continua a partes para modelar os pulsos transientes em tecnologias *sub-micron* (TUROWSKI et al., 2008; ARTOLA et al., 2010).

2.2.1.3 Modelagem em 3D de Efeitos Transientes

A evolução dos sistemas informáticos, algoritmos 3D e melhoras nas interfaces visuais permitiu o desenvolvimento de programas de simulação que podem mostrar em 3D os efeitos de uma partícula radioativa batendo contra um dispositivo a nível elétrico. Estas simulações são necessárias quando são necessários estudos preditivos mais exatos dos fenômenos acontecidos (MUNTEANU; AUTRAN, 2008). Uma destas ferramentas acadêmicas é o CFDRC NanoTCAD (TUROWSKI et al., 2008), que pode simular tanto a partícula inicial como as reações nucleares secundárias acontecidas no material. Existem outros simuladores 3D no mercado, como o DaVinci (DODD et al., 2004) ou TCAD (ARTOLA et al., 2010) da Synopsys. Com estas ferramentas também foi possível o estudo dos efeitos transientes sobre MOSFETS de Silício em Isolante (SOI) para estudar os efeitos dos SEE sobre transistores desta tecnologia (COLLADANT et al., 2002). Mas, de acordo com (SCHÖNE et al., 1998), ainda estas ferramentas tem algumas imperfeições na reprodução nos efeitos dos pulsos transientes de corrente.

2.2.1.4 Injeção de Falhas Transientes em Laboratórios

Uma das melhores formas de injetar falhas transientes devidas a efeitos de radiação é usando radiação. Este método é muito mais preciso que outros métodos, mas é custoso devido ao custo do equipamento e os riscos do uso de fontes radioativas. Tudo isto melhora o estudo dos mecanismos do geração dos SEEs (DUZELLIER; BERGER, 2007). Na atualidade o uso deste equipamento é muito difundido pela exatidão das medições e semelhança com o meio ambiente industrial ou espacial onde os equipamentos finais serão localizados. Na tabela 2.2 mostra alguns dos radioisótopos mais conhecidos para a injeção de falhas em CI (MARTIN et al., 2011). Estes radioisótopos são escolhidos devido a Energia da Partícula que emitem e a sua Meia-Vida.

Radio-Isótopo	Part. Emitida	Energia Típica	Meia-Vida	Prod. de Desintegração
$^{27}\text{Co}^{60}$	β e γ	2.824MeV	5.27 anos	$^{28}\text{Ni}^{60}$
$^{55}\text{Cs}^{137}$	β e γ	1.176MeV	30.07 anos	$^{56}\text{Ba}^{137}$
$^{95}\text{Am}^{251}$	α	5.638MeV	432.2 anos	$^{93}\text{Np}^{237}$
$^{98}\text{Cf}^{252}$	α	6.217MeV	2.645 anos	$^{96}\text{Cm}^{248}$

Tabela 2.2: Principais Radio-Isótopos usados para produzir falhas nos CIs.

Existem mais outros métodos usando radiação para injetar falhas transientes, como o uso do ciclotrones, aceleradores Van de Graff, raios X, fontes Am-Be para geração de nêutrons, etc.

Outro método usado para injetar transientes nos CIs em teste é o uso do Laser de pulsos para excitar o silício e criar e modelar corrente transiente nos CIs (FOUILLAT et al., 2007).

Na atualidade, o Instituto Fraunhofer na Alemanha realiza testes com todas as fontes de irradiação conhecidas, incluindo raios X, prótons e Laser (MARTIN et al., 2011).

2.2.2 Nível de Portas Lógicas

Este nível também é conhecido como Nível RTL. Este nível agrupa o Nível de Portas Lógicas e o Nível de Transferencia de Registradores nos domínios Comportamental e Estrutural do diagrama Gajski-Kuhn da figura 1.1.

2.2.2.1 Modelagem de Falhas Transientes

Nas Portas Lógicas CMOS, um transistor só é sensível na coleção de cargas de um pulso transiente SET quando está inversamente polarizado e desligado. Quando está ligado, a carga coletada na união inversamente polarizada é simplesmente conduzida para os barramentos de alimentação do circuito. Dependendo da locação do impacto do íon, podem acontecer dois tipos de impactos: impactos tipo n e impactos tipo p (XUEYAN; LIYUN; JINMEI, 2011). Na figura 2.4 são mostrados estes tipos de impactos sobre os transistores de uma porta inversora. Um impacto tipo n cria um pulso de tensão de baixo-para-cima, enquanto um impacto tipo p o cria de cima-para-baixo. Uma boa aproximação de modelagem dos pulsos de tensão causados pelos pulsos transientes é o uso de pulsos quadrados de tensão, os quais devem ter uma duração igual que o pulso transiente original. Estes pulsos podem ser afetados por mascaramentos e efeitos de atenuação de pulsos.

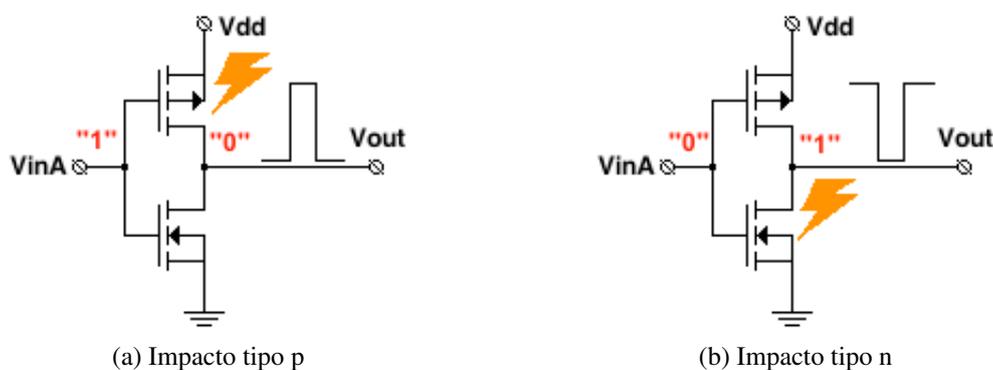


Figura 2.4: Tipos de Impactos de pulsos transientes nos transistores de portas lógicas.

Devido à baixa velocidade de execução de uma simulação SPICE de um circuito relativamente grandes, foram estudadas maneiras de modelar as falhas transientes a nível das Portas Lógicas da forma mais real possível. O simulador TIFAS (CHA et al., 1996) usa a equação (2.1a) para modelar o pulso transiente devido ao impacto de partículas alfa foram definidos parâmetros de tempo da equação ($\tau_F^{-1} = 6.1 \times 10^9 s^{-1}$ e $\tau_R^{-1} = 2.0 \times 10^{10} s^{-1}$) e ficou variável o valor e sinal de I_0 dependendo se o impacto da partícula é do tipo p ou tipo n. Este pulso de corrente causa um pulso de tensão na saída das portas lógicas afetadas, a largura deste pulso W_{pulse} é modelado da seguinte forma:

$$W_{pulse} = A \cdot Q_{total} + B \cdot C_{out} + C \quad (2.11)$$

onde Q_{total} é a carga total depositada pela corrente transiente (descrita na equação (2.1b)) e C_{out} pode ser descrita em unidades de carga padrão, onde uma unidade de carga padrão corresponde ao valor da capacitância de saída de uma porta lógica inversora. Os valores A, B e C são obtidos usando um análise de regressão linear dos dados de simulação SPICE (CHA; PATEL, 1993).

Outra técnica é o uso de registradores de deslocamento para simular os atrasos das portas lógicas reais mediante o uso de descrição HDL e assim poder inserir um pulso retangular que modele o pulso transiente. Este uso do Atraso Quantizado (VALDERAS et al., 2007) modela a duração real de um pulso transiente em um pulso retangular determinado por um numero de “quantas” de tempo. Cada um destes quantas tem uma duração de tempo Q_{n_r} é determinado da seguinte maneira:

$$Q_{n_r} = t_{crit}/n_r \quad (2.12)$$

onde t_{crit} e o tempo do caminho critico do circuito a simular e n_r e a máxima quantidade de registradores de deslocamento que se pode botar como máximo nas entradas o saídas de uma porta lógica. Por exemplo, se um circuito tem um tempo no caminho crítico de $t_{crit} = 818ps$ e o máximo valor de registradores de deslocamento permitidos para modelar os atrasos é de $n_r = 16$, se tem que $Q_{16} = 51ps$. Agora, se um pulso transiente real tem uma duração de $t_{pulse} = 1.173ns$, então é possível representar este pulso como um pulso retangular de $t_{pulse}/Q_{16} = 23$ “quantas” de tempo.

É possível enxergar outros fenômenos quando um conjunto de portas lógicas é alcançado por um pulso transiente, como é o mascaramento do pulso (MUNTEANU; AUTRAN, 2008) e a “ampliação” (*broadening*) da duração do mesmo (DODD et al., 2004; RIBEIRO; WIRTH; KASTENSMIDT, 2008; CAVROIS et al., 2008; STERPONE et al., 2011). Para modelar estes fenômenos deve-se ter em consideração as capacitâncias parasitas, o compartimento de cargas através dos corpos de transistores vizinhos, cargas passivas e ativas conectadas as saídas das portas lógicas, etc.

2.2.2.2 Injeção e Simulação de Falhas Transientes

A nível de portas lógicas, as falhas transientes podem ser injetadas e simuladas usando as mesmas técnicas do nível elétrico, como o uso de SPICE ou simuladores 3D (XU-EYAN; LIYUN; JINMEI, 2011) ou uma combinação destes. A dificuldade é devido que a descrição do circuito fica mais complexa devido na descrição de portal lógicas e sub-circuitos baseados em transistores. O uso de comandos de hierarquização de SPICE podem ser de ajuda para tornar a descrição mais modular (SYNOPSISYS, 2005). Se deseja-se inserir falhas numa biblioteca de portas lógicas proprietário, deve-se estudar a biblioteca e

modificar estas portas a nível SPICE para poder inserir falhas nos nodos desejados usando o mesmo método de inserção de pulso transiente de corrente descrito na seção 2.2.1.2.

Existem métodos de simulação de falhas que usam os FPGAs para sintetizar um circuito dentro dele e inserir-lhe falhas transientes, tal como é explicado em (VALDERAS et al., 2007). É possível usar também a reconfiguração dinâmica de um FPGA para inserir a nível de portas lógicas falhas em tempo real a um circuito lógico sintetizado nele (KAFKA; NOVÁK, 2006). Em (FAURE; VELAZCO; PERONNARD, 2005) é modelado a distribuição de probabilidade de acontecer SEUs na memória RAM de um circuito, a qual é uma distribuição de Poisson.

Existem outros tipos de simuladores acadêmicos que inserem outros tipos de falhas não transientes. O *SIFU!* (CORREIA; LUBASZEWSKI; REIS, 2003) é um programa que pode realizar até duas simulações em paralelo de circuitos combinacionais com falhas *Stuck-at*. Outro simulador, o *Fsimac* (SUR-KOLAY et al., 2000), não só faz simulação de falhas *stuck-at* e por atraso a nível de portas lógicas sobre circuitos combinacionais e portas assíncronas como elementos *Muller-C*, também simula falhas sobre portas complexas dominó que são usadas para projetos de alta velocidade.

2.2.3 Nível do Sistemas

Este nível só ocupa uma pequena parte do diagrama Gajski-Kuhn da figura 1.1 no Domínio Estrutural. Este nível, além de ser o nível mais alto de injeção e modelagem de falhas, não só contém Processadores, também contém Sistemas Embarcados em CI (SOC), Micro-controladores, circuitos complexos de Radio-Frequência, CIs industriais, etc.

2.2.3.1 Modelagem de Falhas Transientes

Os sistemas são circuitos que podem conter desde simples portas lógicas até processadores e circuitos de radio-frequência. Um exemplo de um sistema é um micro-controlador ou SOC (*System on Chip*), o qual possui em seu interior um micro-processador, memória Flash, ALU, portas E/S, osciladores, etc.

A modelagem de falhas transientes devidas a radiação a nível sistema é muito parecido ao modelagem desta a nível de portas lógicas. Um modo de fazer-lo é usando o método descrito em (VALDERAS et al., 2007), onde expõe o uso de modelo de atraso quantizado usando a equação (2.12), mas este modelo não inclui um modelagem do mascaramento elétrico. No trabalho do (ENTRENA et al., 2009) é melhorado o modelo de quantização de atraso agregando um Contador Não Linear (NLC), que tem um período de relógio de Q_{nr} . Este NLC serve para modelar, mediante tabelas, a forma do pulso tomando como referência o atraso da porta t_p : (1) Se a duração do pulso T da entrada na porta é $T > t_p$ e T é suficientemente grande para permitir um *swing* completo de tensão então o pulso é propagado e processado normalmente sem ter em consideração o atraso. (2) Se $T \geq t_p$ (T muito perto de t_p) então o pulso digital é reduzido a uma duração de $T - t_p$. (3) Se $T < t_p$ então o pulso digital propagado tem uma duração de 0. Na figura 2.5a é mostrado estes três casos de degradação de um pulso transiente e na figura 2.5b a implementação do atraso usando uma NLC.

As memórias Flash dos sistemas também são susceptíveis aos transientes devidos a impactos de íons pesados. Estas memórias são necessárias devido a capacidade de manter informação útil por anos dentro de um SOC. Devido a que os transistores de porta flutuante em que se baseiam estas memórias são cada vez mais pequenos devido ao avanço tecnológico, a quantidade de elétrons que as portas flutuantes destes transistores podem

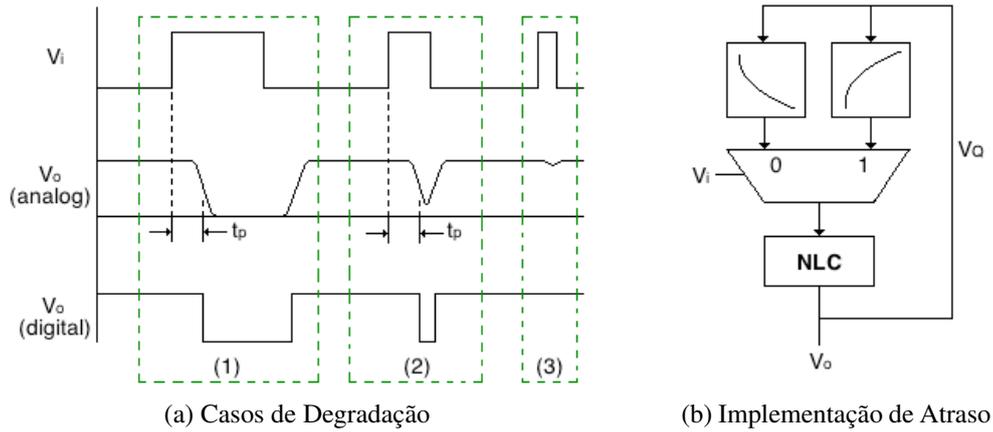


Figura 2.5: Modelagem de Pulsos Transientes (ENTRENA et al., 2009)

armazenar chegam, com sorte, a 1000 (BUTT; ALAM, 2008). Um íon pesado pode bater na porta deste transistor e causar os seguintes efeitos que podem apagar a informação que contem: (1) Tunelamento Assistido por Armadilha (TAT) que acontece quando a partícula gera túneis no óxido que servem como armadilhas para as cargas da porta flutuante. (2) Modelo de Trilha Condutora, onde a partícula cria um caminho transiente de curta duração entre a porta flutuante e o corpo de transistor. (3) Geração/Recombinação/Transporte no Óxido, onde a partícula cria lacunas entre as capas de oxido que rodeiam a porta flutuante, as quais podem recombinar-se com os elétrons desta porta. (4) Emissão Eletrônica, quando os elétrons da porta flutuante ganham energia do íon pesado e são emitidas por este por foto-emissão. Além disso, a mudança de tensão da porta flutuante dependente no tempo $\Delta V_{FG}(t)$ e modelado a seguir:

$$\Delta V_{FG}(t) = \frac{\Delta Q_{FG}(t)}{C_T} \quad (2.13)$$

onde $\Delta Q_{FG}(t)$ é a perda de carga na porta flutuante devido ao fluxo de pares elétron-lacuna gerados pela partícula e C_T é a capacitância total da porta flutuante.

Outras partes sensíveis aos efeitos de correntes transientes nos sistemas são as Memórias RAM. A injeção destas falhas transientes podem ser modeladas de duas formas: No Domínio da Partícula e No Domínio do Tempo (FAURE; VELAZCO; PERONNARD, 2005). No Domínio da Partícula, a probabilidade de que $N_{SEU} = k$ SEUs aconteçam numa RAM $\mathcal{P}(N_{SEU} = k)$ é modelado a seguir:

$$\mathcal{P}(N_{SEU} = k) = \binom{F}{k} \sigma_t^k (1 - \sigma_t)^{F-k} \quad (2.14)$$

onde F é a fluência da partícula e $\sigma_t = N_{SEU}/F$ é a seção de choque (*cross section*) do ponto da RAM exposto a radiação. O valor da seção de choque (ou parâmetro “prova-tório” da Distribuição Binomial) é tal que $\sigma_t \in [0, 1]$. Isto significa que a quantidade de SEUs geradas por partículas tem uma distribuição de Probabilidade Binomial $B(\sigma_t, F)$.

No Domínio do Tempo deve-se ter em conta o tempo de duração do experimento de exposição à radiação T_e , pela expressão:

$$\sigma_t = \frac{N_{SEU}/T_e}{F/T_e} = \frac{R_{SEU}}{\Phi} \quad (2.15)$$

onde $R_{SEU} = \sigma_t \times \Phi$ é a taxa de transtornos (*upsets*) por secundo e Φ é o fluxo das partículas. Para uma seção de choque do ponto σ_t , um fluxo de partículas Φ , uma duração do experimento T_e e uma fluência $F = \Phi \times T_e$, a probabilidade de que $N_{SEU} = k$ SEUs aconteçam numa RAM $\mathcal{P}(N_{SEU} = k)$ é modelado a seguir:

$$\mathcal{P}(N_{SEU} = k) = \frac{e^{-\bar{\mu}}(\bar{\mu})^k}{k!} \quad (2.16)$$

onde $\bar{\mu} = R_{SEU} \times T_e$ é o parâmetro de “meia” da distribuição. Isto significa que a quantidade de SEUs geradas por partículas num tempo determinado tem uma distribuição de Probabilidade de Poisson $P(\bar{\mu})$.

2.2.3.2 Injeção e Simulação de Falhas Transientes

A Nível de Sistema, a injeção de falhas pode-se classificar tradicionalmente em dois tipos: as implementadas por hardware (HWIFI) e as implementadas por software (SWIFI) ou baseadas em simulação (JEITLER; DELVAI; REICHÖR, 2009).

Para injetar e simular falhas transientes a Nível de Sistema pode-se usar o método descrito por (VALDERAS et al., 2007), o qual é uma ferramenta HWIFI, só que é muito devagar para a simulação de sistemas muito complexos.

Outra forma de realizar estas tarefas é usando programas ou ferramentas SWIFI que modifiquem a descrição dos sistemas e circuitos, como o *AMATISTA* (GONZÁLES; BERREROJO, 2001), o software que tem um conjunto de ferramentas para desenvolver circuitos tolerantes a falhas e para injetar falhas nesses circuitos desenvolvidos para ambientes industriais. Outro programa, o *FPGA-based fault simulator* (KAFKA; NOVÁK, 2006) usa reconfiguração dinâmica parcial para simular falhas SEU por mudança das interconexões internas dos sistemas descritos no FPGA. Também se tem o *LIFTING* (BOSIO; NATALE, 2008), um programa que realiza simulações lógicas e de falhas tipo *stuck-at* e SEU sobre circuitos digitais. Estas ferramentas demandam descrições de entrada usando linguagens HDL para realizar as tarefas de injeção e simulação. Para acelerar estas injeções e simulações se criou o *FuSE* (JEITLER; DELVAI; REICHÖR, 2009) o qual usa o programa base da *SEmulator*[®] da Altera.

As memórias RAM são circuitos com uma tendência a falhar nas condições de ambientes espaciais e terrestres. Uma partícula cargada que bate sobre alguns dos transistores de uma célula de memória pode alterar o valor armazenado nela produzindo um *Bit-Flip*. O software *RAMSES* (WU; HUANG; WU, 1999), baseado no algoritmos de March, pode rapidamente simular as falhas neste tipo de memória com o fator de complexidade de $\mathcal{O}(n^2)$. Também se tem o *RAMSES-D* (HSING; WU; WU, 2007) o qual é uma melhora do *RAMSES* que permite simular falhas em DRAMs e usa o método de peso de falha por acoplamento. O *RAMSES-FT* (CHENG et al., 2002) simula falhas nas memórias *Flash* dos sistemas, é baseado na norma IEEE 1005, tem uma complexidade $\mathcal{O}(W^2)$ e propõe um novo algoritmo de March de 14N para simular as falhas neste tipo de memórias.

2.3 Outros Efeitos da Radiação em CIs

Os dispositivos MOS são mais susceptíveis a radiação devido a sua geometria e uso de isolante na porta (frequentemente o SiO_2). Na seguinte seção de descrevem efeitos que podem o não ser associados com Efeitos Transientes e sim pela Dose Total de Ionização (TID). Dependendo do tipo da radiação e sua energia podem acontecer um o mais das seguintes efeitos (SCHRIMPF, 2007) em maior ou menor grau:

- Deslocamento da Voltagem de Limiar.
- Degradação da Mobilidade.
- Variação da Voltagem Sub-Limiar.
- Correntes de Fuga.

2.3.1 Deslocamento da Voltagem de Limiar

Este é, possivelmente, o efeito mais recorrente devido ao TID. É devido principalmente ao efeito eletrostático das cargas aprisionadas e cargas capturadas nas interfaces dos dispositivos. A carga aprisionada no óxido (qN_{ot}) é usualmente positiva, entretanto a carga das interfaces (qN_{it}) depende do potencial de polarização das interfaces e da natureza física do efeito responsável por estas cargas. Estes efeitos eletrostáticos são descritos pela equação de Poisson de uma dimensão:

$$\frac{dE}{dx} = \frac{d^2V}{dx^2} = \nabla^2 V = -\frac{\rho}{\varepsilon} \quad (2.17)$$

onde E é o campo elétrico, V é o potencial eletrostático, ρ é a densidade volumétrica de carga e ε é a permissividade dielétrica do meio.

Quando o MOSFET é exposto a uma dose de radiação são criados pares elétron-lacuna na quantidade dada pela equação (2.3). Isto é descrito pela Dose Total de Ionização (TID) dada geralmente em Grays (1 Joule/kg) ou rads (100 erg/g). Para criar um par elétron-lacuna no óxido da porta do MOSFET é necessário de uma energia $E_{gsio2} = 18eV$. Muitos destes pares se recombinam, os que não geram correntes transientes que afetam a funcionalidade do dispositivo por um curto período de tempo. Mas no óxido os elétrons são relativamente móveis e, devido a certas condições de polarização, deixam o óxido sob a influencia de um campo elétrico. As lacunas se movimentam devagar devido aos defeitos do óxido. Estes defeitos no óxido ou armadilhas tem também níveis de energia. A quantidade de armadilhas é maior nas interfaces Si/SiO₂. Quando se polariza a porta com uma voltagem positiva (estado de ligado para um n-MOS), as lacunas viajam para a interface e uma quantidade significativa destas fica presa nas armadilhas. Além disto, a radiação ionizante afeta criando estados eletrônicos no silício das interfaces criando mais armadilhas devido a liberação do hidrogênio (H₂) do óxido, o qual aporta mais cargas positivas que são presas pelas armadilhas das interfaces e mudando os níveis de energia do silício.

A voltagem de limiar V_{th} para o MOSFET quando não tem cargas no óxido é dada pela seguinte equação:

$$V_{th} = \begin{cases} \Phi_{MS} + 2\phi_F + C_{ox}^{-1} \sqrt{2\varepsilon_{Si} q N_A (2\phi_F)} & \text{canal N} \\ \Phi_{MS} + 2\phi_F - C_{ox}^{-1} \sqrt{2\varepsilon_{Si} q N_D (2|\phi_F|)} & \text{canal P} \end{cases} \quad (2.18)$$

onde Φ_{MS} é a diferença da funções trabalho do metal e do semiconductor, ϕ_F é o potencial do corpo do transistor (ou *bulk*), C_{ox} é a capacitância por unidade de área (F/m^2), ε_{Si} é a permissividade dielétrica do silício, N_A é a concentração do dopantes receptores (Si tipo p) e N_D é a concentração do dopantes doadores (Si tipo n).

A variação da voltagem de limiar devida a radiação ΔV_{th} depende da densidade volumétrica da carga aprisionada no óxido ρ_{ox} e a posição física x :

$$\Delta V_{th} = -\frac{1}{\varepsilon_{ox}} \int_0^{t_{ox}} x \rho_{ox}(x) dx \quad (2.19)$$

onde ε_{ox} é a permitividade dielétrica do óxido do silício e t_{ox} é a espessura do óxido. Se a carga se localiza na interface Si/SiO₂, então o efeito é maximizado e o incremento da voltagem de limiar é:

$$\Delta V_{th} = \frac{Q_{int}}{C_{ox}} \quad (2.20)$$

onde Q_{int} é a densidade total de carga superficial na interface e $C_{ox} = \varepsilon_{ox}/t_{ox}$. É possível representar a Q_{int} da seguinte maneira:

$$Q_{int} = Q_{it} + Q_{ot} \quad (2.21)$$

onde Q_{it} é a densidade de cargas aprisionadas na interface Si/SiO₂ e Q_{ot} é a densidade de cargas capturadas no óxido. Então:

$$\Delta V_{th} = \frac{Q_{it}}{C_{ox}} + \frac{Q_{ot}}{C_{ox}} = V_{it} + V_{ot} \quad (2.22)$$

onde $V_{it} = Q_{it}/C_{ox}$ é a tensão de deslocamento devida às cargas na interface e $V_{ot} = Q_{ot}/C_{ox}$ é a tensão de deslocamento devida às cargas do interior do óxido. Os efeitos destas tensões sob a curva I_D vs. V_{GS} podem ser observadas na figura 2.6.

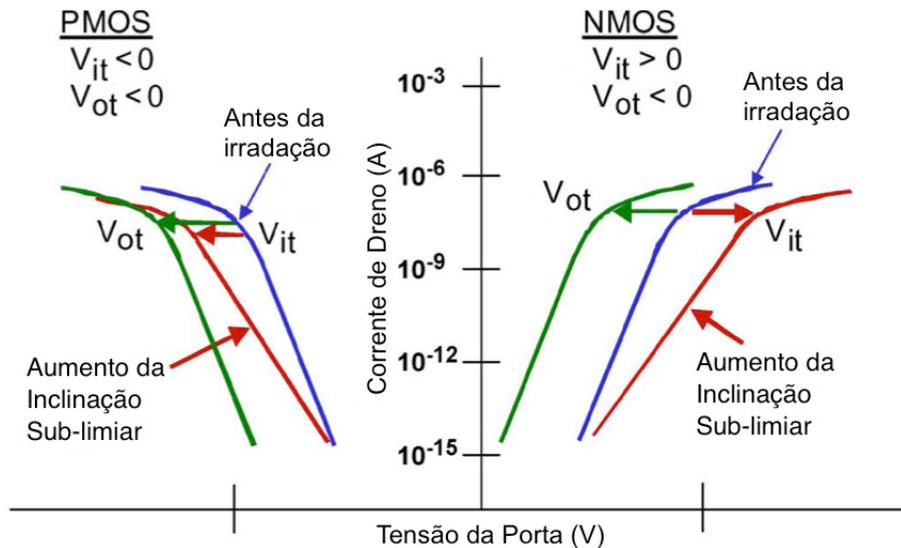


Figura 2.6: Deslocamento da tensão e mudança da inclinação sub-limiar para transistores NMOS e PMOS são apresentadas junto com suas curvas antes da irradiação (LACOE, 2008).

Também o ΔV_{th} depende da espessura do óxido de acordo com a seguinte relação (LACOE, 2008), conhecida como momento eletrostático (SCHRIMPF, 2007):

$$\Delta V_{th} \propto t_{ox}^2 \quad (2.23)$$

Então a variação da tensão de limiar se volta mais pequena quando a espessura do óxido é pequena, algo que acontece nas tecnologias mais recentes.

2.3.2 Degradação da Mobilidade

O silício com dopantes tipo n tem uma mobilidade μ_n e com dopantes do tipo p uma mobilidade μ_p , e de acordo com (KANO, 1998) tem-se as seguintes expressões dependentes da temperatura ambiental T :

$$\mu_n = 88T_n^{-0.57} + \frac{7.4 \times 10^8 T^{-2.33}}{1 + \left[\frac{N}{1.26 \times 10^{17} T_n^{2.4}} \right] 0.88 T_n^{-0.146}} \quad (2.24a)$$

$$\mu_p = 54.3T_n^{-0.57} + \frac{1.36 \times 10^8 T^{-2.23}}{1 + \left[\frac{N}{2.35 \times 10^{17} T_n^{2.4}} \right] 0.88 T_n^{-0.146}} \quad (2.24b)$$

onde $N = N_A + N_D$ é a densidade total do impurezas (em cm^{-3}) e $T_n = T/300$. Para $T = 300K$ as expressões (2.24) se reduzem a:

$$\mu_n = 88 + \frac{1252}{1 + 0.698 \times 10^{-17} N} \quad (2.25a)$$

$$\mu_p = 54.3 + \frac{407}{1 + 0.374 \times 10^{-17} N} \quad (2.25b)$$

No sistema metal-óxido-semicondutor, a mobilidade na camada de inversão pode mudar de forma significativa devido ao espalhamento das cargas induzidas pela radiação. Estas cargas dependem só da proximidade das cargas na interface. Seja μ_0 a mobilidade geral do sistema quando não tem radiação aplicada, a qual pode ser μ_n ou μ_p dependendo do caso, então se define a μ como a mobilidade geral quando existe irradiação, da seguinte maneira (SCHRIMPF, 2007):

$$\mu = \frac{\mu_0}{1 + \alpha_{it} N_{it} + \alpha_{ot} N_{ot}} \quad (2.26)$$

onde α_{it} e α_{ot} são parâmetros que quantificam os efeitos das armadilhas na interface e cargas capturadas no óxido, respectivamente, sob a mobilidade. Devido a que as armadilhas na interface tem maior efeito na mobilidade que as cargas capturadas no oxido $\alpha_{it} > \alpha_{ot}$. A figura 2.7 mostra o efeito da dose total de radiação sob a mobilidade normalizada $\bar{\mu}_0$. Este modelo melhora o modelo de (GALLOWAY; GAITAN; RUSSELL, 1984), o qual não considera a N_{ot} . Se tem evidência que o modelo BSIM3 que usa o SPICE não faz uma modelagem correta para o parâmetro N_{ot} (PAUCAR; SILVA, 2011) na zona de inversão forte, o qual pode ser importante para uma simulação mais precisa do fenômeno de degradação de mobilidade quando os MOSFETs trabalham nesta região.

2.3.3 Variação da Voltagem Sub-Limiar

A corrente do dreno I_D não some abruptamente com a tensão de porta V_{GS} , se não que decresce para menos da tensão de limiar V_{th} . A região onde $V_{GS} < V_{th}$ é chamada de região sub-limiar. A corrente I_D vem dada pela seguinte equação (CAVERLY, 2007):

$$I_D = I_M \frac{W}{L} e^{\frac{V_{GS} - V_T}{n\phi_T}} \left[1 - e^{\frac{V_{DS}}{\phi_T}} \right] \quad (2.27)$$

onde I_M é a corrente específica da região sub-limiar, n é a constante de forma da curva de tensão sub-limiar e $\phi_T = kT/q$ é a tensão térmica.

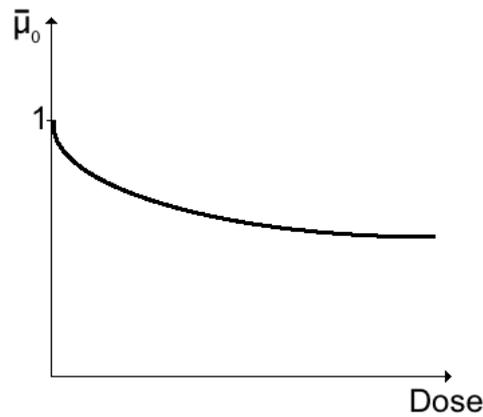


Figura 2.7: Mobilidade na camada de Inversão v.s. Dose total de um MOSFET irradiado (SCHRIMPF, 2007).

Usando a aproximação de que a superfície de potencial está perto do valor de inversão forte permite o cálculo da inversa da inclinação sub-limiar S (SCHRIMPF, 2007):

$$S = \left(\frac{\partial \log I_D}{\partial V_{GS}} \right)^{-1} = \phi_T \ln(10) \left[1 + \frac{C_D}{C_{ox}} + \frac{C_{it}}{C_{ox}} \right] \quad (2.28)$$

onde C_D é a capacitância da camada de depleção por unidade de área e C_{it} é a capacitância por unidade da área associada às armadilhas da interface. Como se mostra na seção 2.3.1, a densidade de cargas Q_{it} se incrementa devido à irradiação, a inclinação sub-limiar então se faz mais grande e o MOSFET fica mais devagar. A figura 2.6 mostra esta tendência devida a $V_{it} = Q_{it}/C_{ox}$.

2.3.4 Correntes de Fuga

Na seção 2.3.1 foram mostrados que os efeitos da TID são menores conforme se escala na tecnologia, mas o óxido de isolamento entre os transistores é espesso e capaz de reter cargas e produzir correntes de fuga indesejáveis. A figura 2.8 mostra um transistor com isolamento de óxido do campo (FOX). A combinação do polisilício da porta, a espessura do óxido de isolamento na região de transição Porta/Isolamento e a difusão $p+$ embaixo da porta formam um transistor parasita na borda, o qual atrapalha lacunas depois da exposição do dispositivo na radiação (LACOE, 2008).

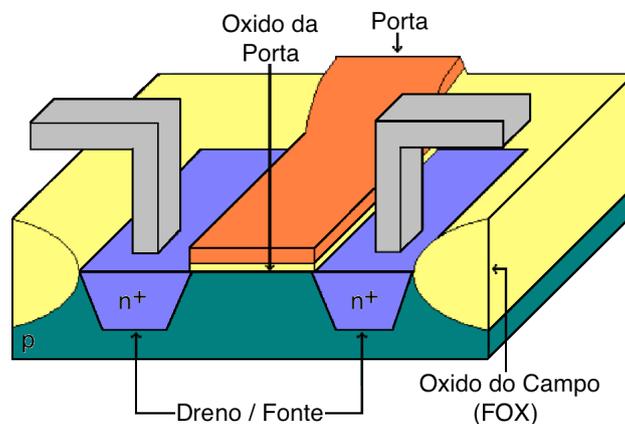


Figura 2.8: Transistor com oxido do campo (FOX) como isolante entre transistores.

Na figura 2.9 é mostrado este efeito indesejável. A radiação induzida atrapalha lacunas em estes Transistores de Borda produzindo uma zona de inversão ou canal parasita, o que resulta num deslocamento da voltagem de limiar, grande demais para ligar estes, produzindo uma corrente de fuga quando o transistor está desligado. Este efeito é mais crítico em tecnologias de Isolamento por Trincheiras Profundas (STI) devido a que o área da interface é maior.

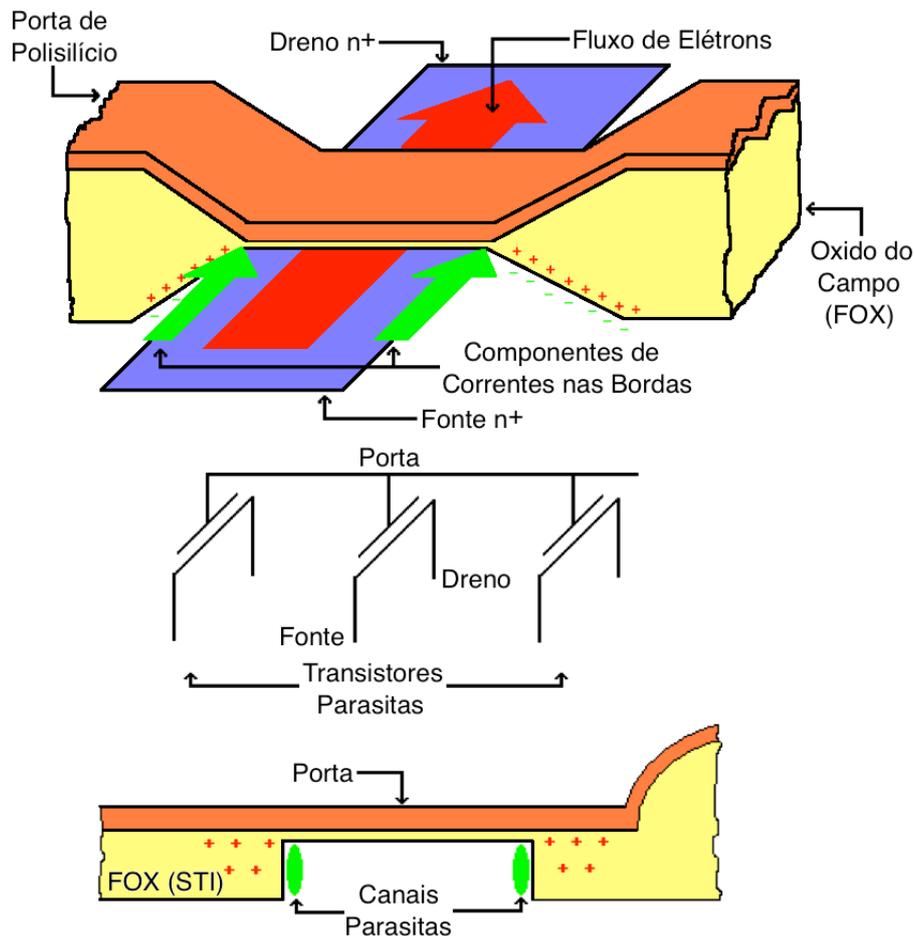


Figura 2.9: Transistores Parasitas induzidos pela TID sob um transistor com FOX.

2.4 Métodos de Tolerância a Falhas

A tolerância a falhas em CIs é um tema importante na área industrial e espacial. Os ambientes industriais e no espaço exterior contêm muitas fontes que podem danificar os dados e os CIs de forma momentânea ou permanente. A seguir são apresentados os métodos mais conhecidos para poder mitigar os efeitos causados pelas falhas transientes:

2.4.1 Registradores Triplicados

Existem técnicas para fazer um circuito tolerante as falhas causadas por SET, SEU e efeitos de segundo e terceira ordem destes (GUSMÃO KASTENSMIDT; CARRO; REIS, 2006), uma destas é triplicar o hardware, conhecida pelas siglas TMR (*Triple Modular Redundancy*). O TMR se baseia em repetir três vezes o mesmo hardware e usar um circuito chamado de Votador de maioria ou *Voter*, no qual, de três valores de entrada

o votador simplesmente joga para saída o resultado que coincida com pelo menos dois valores de entrada iguais. As saídas de cada um dos circuitos redundantes é ligada às entradas deste circuito que tem a seguinte equação lógica:

$$Z = P_1P_2 + P_2P_3 + P_1P_3 = \overline{\overline{P_1P_2} \overline{P_2P_3} \overline{P_1P_3}} \quad (2.29)$$

onde P_1 , P_2 e P_3 são as saídas de cada um dos circuitos redundantes e Z é a saída do *Voter*. Se acontecer algum *Bit-Flip* em um dos circuitos redundantes, o *Voter* faz o trabalho de escolher o valor certo supondo que só um dos circuitos redundantes é afetado pelo SEU. Esta técnica tem a desvantagem de aumento de área do circuito final em mais de três vezes, além de que se os três dados saídos dos registradores estão errados, o resultado Z também estará errado. Os resultados obtidos experimentalmente colocam em evidência que algumas falhas na memória de configuração de FPGAs baseados em SRAM podem provocar uma “Mutaç o” da aplica o, que resulta na incapacidade do *Voter* em detectar a falha (FOUCARD; PERONNARD; VELAZCO, 2010). Esta “muta o” pode ser desde a mudan a de uma das fun o es l gicas internas do *Voter* at  a mudan a total do bloco. Esta t cnica   mostrada na figura 2.10a.

Outra t cnica, baseada na anterior,   usar o TMR com Redund ncia de Tempo, onde cada circuito triplicado tem um rel gio de mesma frequ ncia mas com fase deslocada (VELAZCO; FOUILLAT; REIS, 2007). Esta t cnica   baseada no princ pio de mascaramento temporal do dado, o qual ser  exposto na se o 2.4.4, e melhora a resposta do TMR Cl ssico. Este circuito   mostrado na figura 2.10b.

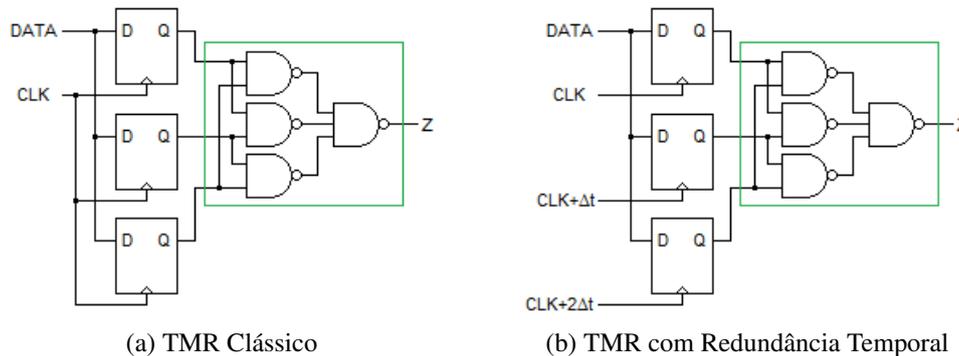


Figura 2.10: T cnicas de Redund ncia Modular Triplicada (TMR).

2.4.2 C digos de Corre o de Erros

Uma das formas de proteger os dados contra as mudan as causadas por ca da de bits durante a transmiss o   armazenamento em mem rias RAM devido a SEE, ru do, imped ncia no cabo de transmiss o, etc.   o uso de C digos de Detec o e Corre o de Erros (EDAC). Estes c digos se baseiam no uso de bits adicionais para colocar junto com os bits de dados. Desta forma   poss vel usar estes bits para poder detectar erros na transmiss o e eventualmente corrigi-los. Estes m todos s o usados comumente a N vel de Sistema.

Historicamente o primeiro c digo de detec o de erros era o “Bit de Paridade”, o qual se baseava no uso de uma porta XOR que criava um bit de paridade a partir dos bits da informa o a enviar. Este m todo, combinado com a Codifica o Gray (LUCAL, 1959) aumentava a probabilidade dos dados chegarem corretos a seu destino. Os *Checksums* (baseados na soma de todos os bits da informa o) e *Cyclic Redundancy Checks* (baseados

em algoritmos Hash) foram criados depois para melhorar a detecção de erros e foram usados para melhorar a recepção de dados via radio (TERVO; MWAKATAPANYA, 1993). Quando um destes códigos detecta um erro, ativa-se um sinal para que a informação volte a ser transmitida.

Surgiu a necessidade de criar códigos que também sirvam para corrigir os erros. Os códigos Hamming dividem a informação em subconjuntos Abelianos e geram bits de detecção e correção de erros para cada subconjunto, incluso para os bits criados (REED, 1954). Outros códigos, como os Reed-Solomon (SOLOMON, 1968) que descrevem uma forma sistemática de gerar códigos que podem detectar e corrigir erros múltiplos aleatórios em blocos de dados.

Também surgiram os códigos adaptativos com o mostrado em (CHEN; WU, 2010) para garantir o conteúdo das RAM. Este código está baseado no uso do construtor de códigos Hsiao que usa matrizes para fazer a codificação.

2.4.3 Endurecimento de Transistores e Células SRAM

Outra forma de fazer os dispositivos menos sensíveis aos efeitos da radiação a Nível Elétrico é mudando a forma dos transistores. O trabalho de (ZHOU; CHOUDHURY; MOHANRAM, 2006) mostra uma forma para dimensionar os transistores usando as equações do modelo de (DAHLGREN; LIDEN, 1995). O objetivo é o re-dimensionamento da largura W dos transistores para voltar-lhes insensíveis aos efeitos dos SETs usando programação geométrica. Este método relato um incremento de área muito parecida as técnicas tradicionais. Existem outros tipos de transistores que resistem melhor os efeitos do TID (FACCIO, 2007; LACOE, 2008), mas estes tem alguns inconvenientes, como o grande consumo de área e não ter bibliotecas para o desenho digital. Estes desenhos de transistores são conhecidos como “Radiation Hardener by Design”(RHBD).

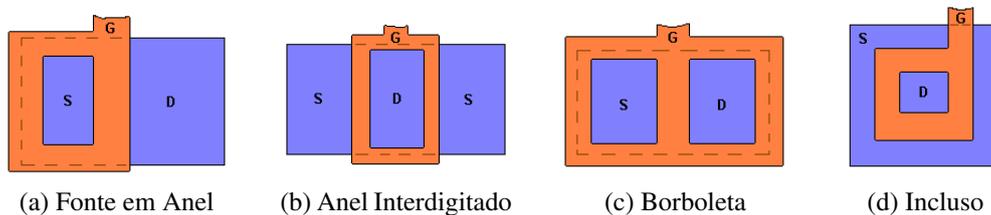


Figura 2.11: Tipos de Transistores resistentes ao TID (FACCIO, 2007).

Para fazer as células RAM e latches insensíveis aos efeitos transientes se desenvolveram as células DICE (*Dual Interlock Cell*) (FACCIO, 2007). Estas células tem a característica de poder armazenar a informação duplicada, só que a área consumida é do dobro das células convencionais já que usa 12 transistores. Em (LIN; KIM; LOMBARDI, 2011) se propõe o uso de células de 13 transistores que tem uma probabilidade de falha muito menor que as DICE.

2.4.4 Mascaramento de Erros

O mascaramento de erros não é uma técnica propriamente dita para eliminar os pulsos de tensão que se propagam por uma rede lógica, é um mecanismo próprio das portas lógicas e da lógica mesma que não percebe o erro. Isto é mais crítico se o transiente tem suficiente carga e chega a propaga-se por toda a rede até um *latch* ou memória que captura ele como dado válido, transmitindo o erro a outros estágios do circuito. Um transiente

pode não ser capturado por o circuito de memória se acontece um mascaramento devido a um destes fenômenos (MUNTEANU; AUTRAN, 2008):

- Mascaramento Elétrico (figura 2.12a). Acontece quando um transiente tem um comprimento de banda muito alto que a frequência de corte da porta CMOS. Devido a isso o pulso é reduzido (efeito filtro). Mas com a existência de portas lógicas atuais de comportamento não linhal a ganho de tensão, os pulsos de baixa frequência podem ser amplificados.
- Mascaramento Lógico (figura 2.12b). Acontece quando uma partícula bate numa porção da lógica combinacional que não afeita a saída devido a que o valor da função lógica é completamente determinado pelo valor da outra entrada de essa mesma porta.
- Mascaramento Temporal (figura 2.12c). Acontece quando a o pulso transiente alcança ao elemento de memória, mas não a transição de relógio que serve para capturar o dado de entrada. E dizer, o pulso não coincide com a *latching window*.

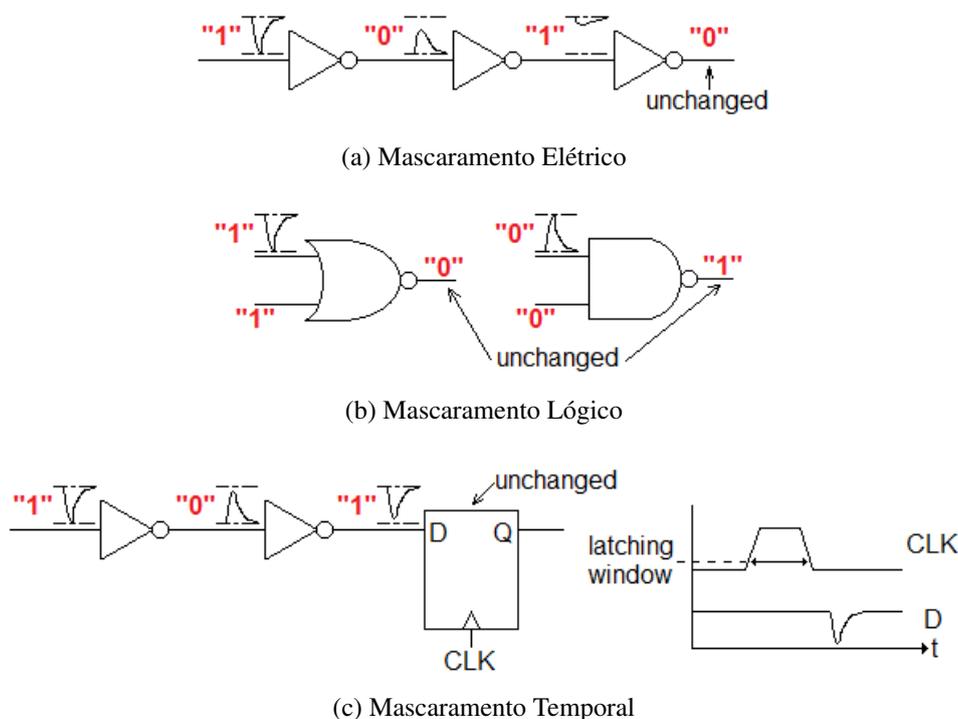


Figura 2.12: Tipos de Mascaramentos de Transientes em CIs Digitais (MUNTEANU; AUTRAN, 2008)

2.5 Falhas Transientes em Circuitos Analógicos

Estes efeitos alcançam também aos circuitos analógicos como amplificadores operacionais, VCOs, PLLs, osciladores, etc. Assim os dispositivos de comunicações e navegação do equipes espaciais resultam danificados temporal ou permanentemente (ECOFFET, 2007). Os danos temporais resultam em ruído rosado (de densidade espectral de potência proporcional a $1/f$) nos dados internos e transmissões desde estes aparelhos, enquanto os

danos permanentes resultam em troca permanente das características dos dispositivos até a destruição dos mesmos.

A modelagem, simulação e injeção de falhas transientes em circuitos analógicos é um tema muito mais novo devido a que os atributos usados para os SEEs em circuitos digitais não são os mesmos dos circuitos analógicos (TURFLINGER, 1996). Isso é o motivo pelo qual se fala de “*Analog Single-Event Effects*” (ASEE). Para isto deve lembrar-se os três estágios acontecidos durante o SEE: (1) Impacto do íon, (2) Coleção de cargas e (3) Resposta do circuito ao fenômeno. Os primeiros dois estágios são inerentemente fenômenos analógicos, mas o terceiro é analógico ou digital. Além disso, deve considera-se que os circuitos analógicos um SEE pode comprometer a mais da metade dos transistores de um circuito tudo, comprometendo a funcionalidade do dispositivo. Para a injeção de falhas em circuitos analógicos se precisa de equipamento de alta precisão, filtros, discriminadores, etc. adequados para o equipe de irradiação. Na figura 2.13 se mostra a instrumentação típica de um laboratório de testes de ASEE.

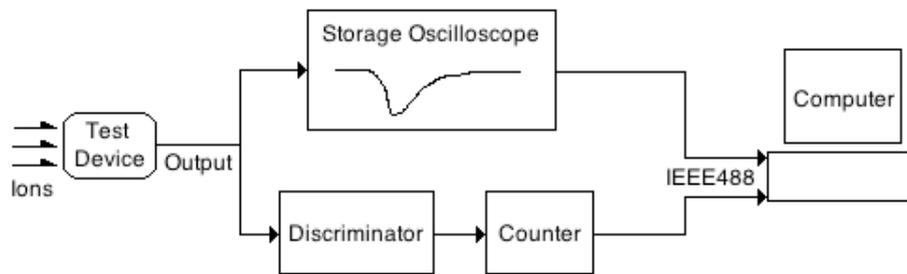


Figura 2.13: Instrumentação típica de testes para ASEE (TURFLINGER, 1996).

Os primeiros modelagens é testes em circuitos analógicos foram feitos sobre Amplificadores Operacionais e Conversores Analógicos-Digitais do tipos Multi-Estágio e Σ - Δ (TURFLINGER, 1996). Para estes experimentos foi necessário criar circuitos de polarização e prova para estes dispositivos. Para fazer os análises de dados se modificou a equação da seção de choque σ_t mostrada em (FAURE; VELAZCO; PERONNARD, 2005):

$$\sigma_t = \frac{N_{upset}}{F \cdot \cos \theta \cdot R} \quad (2.30)$$

onde N_{upset} é o número de *upsets* acontecidos no circuito, F é a fluência, θ é angulo de impacto da partícula e R é um fator da racionalização que permite analisar a σ_t de forma conveniente (TURFLINGER, 1996).

Se desenvolveram metodologias para o análise de transientes que afeitam os circuitos analógicos lineares e métodos de simulação das mesmas, como as desenvolvidas em (ADELL et al., 2000) e (POIVEY et al., 2001), as quais contemplam tanto a parte experimental de bombardeio do DUT com íons pesados e a simulação de dispositivos lineares por computador para finalmente fazer comparações entre os resultados destes experimentos com as simulações.

Se fizeram também testes sob dispositivos não lineares, como no caso de (CHEN et al., 2003). Os experimentos realizados demonstraram que os tempos transientes de rompimento inseridos dependem do tempo de inicialização do circuito e do tempo de recuperação deste. Também se faz o estudo do espectro de frequência para determinar a Assinatura de Amortecimento, uma característica dos circuitos osciladores que pode revelar qual e a mínima energia ou carga críticas para que este tipo de circuito analógico falhe.

Em (CHEN et al., 2006) se expõe o uso de transistores RHBD de 130nm tipo anel da figura 2.11a (mostrados na seção 2.4.3) para a criação de DUT. Os resultados demonstraram que os efeitos TID e SET devidos a radiação são menores que em um DUT de transistores tradicionais.

Para fazer que os dispositivos analógicos sejam resistentes aos efeitos de radiação, as companhias que fabricam estes devem garantir que os circuitos comprem com evitar uma Maior Sensibilidade a Radiações de Baixa Dose (ELDRS). Experimentos como os mostrados em (POIVEY et al., 2001) e (KRUCKMEYER; BUCHNER; DASGUPTA, 2009) testam estes circuitos para aplicações militares, aeronáuticas e espaciais. Na tabela 2.3 se mostram alguns dos íons usados para fazer estes testes. É possível variar o angulo de impacto destes íon e assim variar a carga do transiente produzido. Também é usado o Laser para inserir falhas transientes nestes circuitos e testa-lhes (POIVEY et al., 2001; CHEN et al., 2003, 2006).

Íon	Energia (MeV)	LET _{Si} (MeV cm ² /mg)	Trilha no Si (μm)
O	130	2.6	143
F	142	3.4	122
Mg	165	6	86
Cl	215	11.4	65
Ti	230	18.7	48
Br	285	11.4	36
I	365	60	34
Ne	15	2.9	246
Ar	15	9	162
Kr	15	30	108

Tabela 2.3: Íons típicos usados durante os Experimentos de SEEs em Circuitos Analógicos (POIVEY et al., 2001).

3 FERRAMENTAS DESENVOLVIDAS

A biblioteca de Instrumentos Virtuais (VIs) foi desenvolvida para atender uma simulação sobre portas lógicas sobre um certo tipo de falhas, envolvendo variações de tensão. Para realizar a simulação lógica foi desenvolvida uma série de VIs que compõem o programa principal, que consiste na utilização de valores binários ao contrário do item anterior que trata da variação de tensão do sinal que passa pelos circuitos. Estas duas formas de simulação serão melhor detalhadas nas seções seguintes. Todos os VIs foram desenvolvidos em *NI LabVIEW v8.20*.

3.1 Simulação de Portas Lógicas

Este conjunto de VIs se baseia na utilização de *arrays* de ponto flutuante de dupla precisão que simulam a evolução dos sinais no circuito ou porta lógica. Este tipo de simulação tem por objetivo a análise da evolução da onda de forma analógica, verificando variações da amplitude de tensão que distinguem os sinais digitais “0” e “1”. Além destas portas lógicas, foram criados geradores de SET baseados na equação (2.1a), um simulador de carga de uma capacitância, geradores de relógio simples e triplo com deslocamento, um *Flip-Flop* tipo D, um gerador de padrão digital, simuladores de faixa proibida de tensão um *buffer* com histerese e mais. Todos estes VIs estão desenvolvidos pensando na tecnologia CMOS de 250nm, mas é possível mudar os parâmetros tecnológicos e geométricos para dispor de outras tecnologias. O resumo destes VIs são mostrados nas tabelas 3.1 e 3.2, os quais serão detalhados mais adiante. Na tabela 3.3 se relata os parâmetros físicos usados para esta tecnologia (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a) em cada um dos VIs criados. Para esta tecnologia, a espessura do oxido é de $t_{ox} = 4.50\text{nm}$ (KAWAMOTO; CHO; DUTTON, 2001) e uma largura da porta fixa $L = 250\text{nm}$, mas para este caso se usará o valor de $t_{ox} = 5.60\text{nm}$ (típicos) exposto no arquivo de tecnologia SPICE da Universidade de Berkeley (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003b).

Ícone do VI	Nome do VI	Comentário
	Transistor MOSFET.vi	Simula o Transistor MOSFET para Portas Lógicas.
	Hysteresis.vi	Simula o princípio do cruzamento das faixas de tensão abaixo de $1/3$ de V_{DD} e acima de $1/3$ de V_{SS} .
	Logic Level.vi	Simulação do <i>band gap</i> das Portas Lógicas.
	Capacitive Load.vi	Simula uma Carga Capacitiva aterrada.
	NAND.vi	Simula uma Porta Lógica NAND com falha inserida.
	NOR.vi	Simula uma Porta Lógica NOR com falha inserida.
	NOT.vi	Simula uma Porta Lógica Inversora com falha inserida.
	D Register.vi	Simula um Registrador D com falha inserida.

Tabela 3.1: Resumo de Funções Lógicas e Funções Úteis do pacote de Simulação de Portas Lógicas.

Ícone do VI	Nome do VI	Comentário
	SET.vi	Cria um pulso SET de acordo com (2.1).
	SETxn.vi	Cria “n” pulsos SET iguais em posições de tempo determinadas.
	RndSETxn.vi	Cria “n” pulsos SET aleatórios em posições de tempo aleatórias.
	Clock Generator.vi	Cria um sinal de Relógio ideal.
	Complementary Clock Generator.vi	Cria dois sinais de Relógio ideal, uma e o complemento da outra.
	Delta t Clock Generator.vi	Cria três sinais de Relógio ideal, deslocadas uma quantidade de graus Δt .
	Real Clock Generator.vi	Cria um sinal de Relógio real com tempos de subida t_r e descida t_f .
	Digital Pattern.vi	Cria um padrão digital a partir de um número inteiro inserido.

Tabela 3.2: Resumo de Geradores de Funções Lógicas do pacote de Simulação de Portas Lógicas.

Parametro	Símbolo	PMOS	NMOS	Unidade
Voltagem de Limiar à $V_{GS} = 0$	V_{th0}	-0.40	0.43	V
Efeito Corpo de Transistor	γ	0.40	0.40	$V^{0.5}$
Potencial do Trabalho	ϕ_F	-0.30	0.30	V
Dopagem Total	$N = N_D + N_A$	1×10^{16}	1×10^{16}	cm^{-3}
Comprimento de Canal	λ	-0.10	0.06	V^{-1}
Velocidade da Saturação	v_{sat}	100000	100000	m/s

Tabela 3.3: Parâmetros Físicos da Tecnologia de 250nm para os transistores de Portas Lógicas.

3.1.1 Simulação do Transistor para Portas Lógicas

Para poder simular as Portas Lógicas usando o LabVIEW, precisa-se primeiro de simular um transistor MOSFET usando as equações apresentadas em (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a) e (CAVERLY, 2007) para um transistor que só trabalha nas regiões de corte e saturação. A figura 3.1 mostra o VI Transistor MOSFET.vi que simula este transistor. Este VI pode-se considerar do tipo *Poit-By-Point* devido a que suas entradas são do tipo numérico de ponto flutuante.

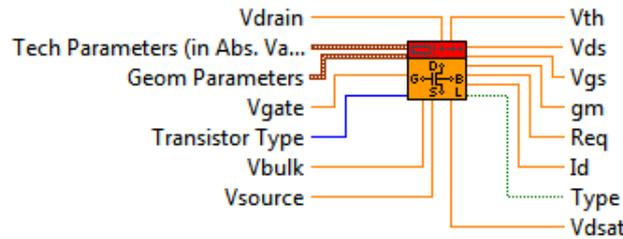


Figura 3.1: VI do Transistor MOSFET.vi.

As entradas **Vdrain**, **Vgate**, **Vbulk** e **Vsource** são as voltagens de dreno, porta, corpo e fonte do transistor respectivamente. A entrada **Transistor Type** serve para indicar se o transistor é PMOS ou NMOS. A entrada **Tech Parameters** é um *cluster* que serve para especificar os parâmetros físicos da tecnologia a serem usados (como os relatados na tabela 3.3) e devem especificar-se no valor absoluto. A entrada **Geom Parameters** é um *cluster* que serve para indicar os parâmetros geométricos do transistor como a largura da porta W , o comprimento da porta L , a espessura do óxido t_{ox} e a longitude de sobreposição x_d (*overlapping*). Na figura 3.2 se mostra estes parâmetros geométricos sob um transistor.

De acordo com (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a) e dependendo do tipo de transistor especificado no **Transistor Type**, a tensão de limiar do transistor V_{th} é dada pela seguinte equação:

$$V_{th} = V_{th0} + \gamma(\sqrt{|(-2)\phi_F + V_{SB}|} - \sqrt{|2\phi_F|}) \quad (3.1)$$

Agora, com os dados fornecidos por todas as entradas e o V_{th} calculado pela equação 3.1 se deve calcular a corrente do dreno I_D em duas regiões de operação: a região Resistiva (ou Tríodo) e a região de Saturação. Escolhe-se entre estes dois valores dependendo se o $V_{DS} = V_D - V_S$ passa ou não o limite da tensão de saturação $V_{GT} = V_{GS} - V_{th}$. Para um transistor NMOS, se $V_{DS} \geq V_{GT}$ então $I_D = I_{Dsat}$, se não $I_D = I_{Dres}$. Para um transistor PMOS, se $V_{DS} \leq V_{GT}$ então $I_D = I_{DSAT}$, se não $I_D = I_{Dres}$.

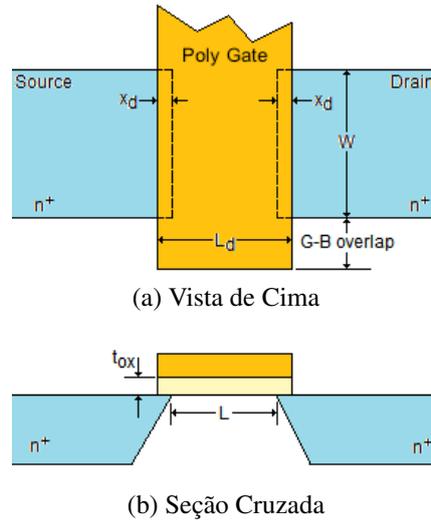


Figura 3.2: Parâmetros Geométricos do Transistor MOSFET (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a).

$$I_{Dres} = \frac{\mu_n \varepsilon_{ox}}{t_{ox}} \frac{W}{L} \left[V_{GT} V_{DS} - \frac{V_{DS}^2}{2} \right] \quad (3.2a)$$

$$I_{Dsat} = \frac{\mu_n \varepsilon_{ox}}{2t_{ox}} \frac{W}{L} V_{GT}^2 (1 + \lambda V_{DS}) \quad (3.2b)$$

O conjunto de equações 3.2 são dadas para o transistor NMOS. A mobilidade μ_n é calculada usando a equação 2.25a. Para obter a corrente de saturação do transistor I_{Dsat} deve-se calcular primeiro o valor da tensão de saturação V_{Dsat} e substituir este na equação 3.2a por V_{DS} (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a):

$$V_{Dsat} = \kappa V_{GT} \quad (3.3)$$

O valor κ mede o grau de velocidade de saturação e é definido a seguir:

$$\kappa = \frac{1}{1 + \frac{V_{GT}}{\xi L}} \quad (3.4)$$

onde $\xi = 2v_{sat}/\mu_n$ é o valor do campo elétrico do canal do transistor, o qual se volta crítico e máximo ao alcançar o valor de v_{sat} .

A tensões V_{th} , V_{DS} , V_{GS} , V_{Dsat} e a corrente I_D são fornecidas pelas saídas do VI **Vth**, **Vds**, **Vgs**, **Vdsat** e **Id** respectivamente. A saída **Req** e o valor R_{eq} que depende da corrente $I_{DSAT} = I_{Dres}(V_{DS} = V_{Dsat})$ é o transistor, este ligado completamente (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a):

$$R_{eq} = \frac{3}{4} \frac{V_{DS}}{I_{DSAT}} \left[1 - \frac{5}{6} \lambda V_{DS} \right] \quad (3.5)$$

A saída **gm** representa a transcondutância g_m do transistor que é definida da seguinte maneira a partir da equação 3.2b (CAVERLY, 2007):

$$g_m = \frac{\partial I_D}{\partial V_{GS}} = \frac{\mu_n \epsilon_{ox}}{t_{ox}} \frac{W}{L} V_{GT} \quad (3.6)$$

A saída **Type** só fornece informação do tipo de transistor de ser preciso. Se **Type=FALSE**, o transistor é NMOS, e se **Type=TRUE** é PMOS.

3.1.2 Histerese, Carga Capacitiva e Faixa Proibida

Criaram-se algumas outras ferramentas para ajudar a simular portas lógicas com mais realismo. Estas ferramentas são os VIs *Hysteresis.vi*, *Capacitive Load.vi* e *Logic Level.vi*.

3.1.2.1 *Hysteresis.vi*

Esta ferramenta simula um *Buffer* com Histerese. Este VI se baseia no princípio do cruzamento das faixas de tensão abaixo de $1/3$ de V_{DD} e acima de $1/3$ de V_{SS} (GND). Quando o sinal de entrada vai subindo e cruza a faixa de $2(V_{DD} - V_{SS})/3$, a saída fica em “1” lógico. Se o sinal de entrada vai descendo e cruza a faixa de $(V_{DD} - V_{SS})/3$, a saída fica em “0” lógico. Uma vez não acontecendo qualquer uma das condições anteriores, o valor fica inalterado. Deve-se sempre considerar que $V_{DD} > V_{SS}$. A figura 3.3 mostra o VI e seu BD correspondente. As entradas **VDD** e **VSS** fixam as tensões de alimentação e a entrada **Vin** serve para o ingresso do dado a avaliar. A saída **Logic Level** indica se a saída é um “1” lógico (*TRUE*) ou “0” lógico (*FALSE*). Por sua natureza de manejar só valores escalares, pode ser considerado como VIs *Point-By-Point*.

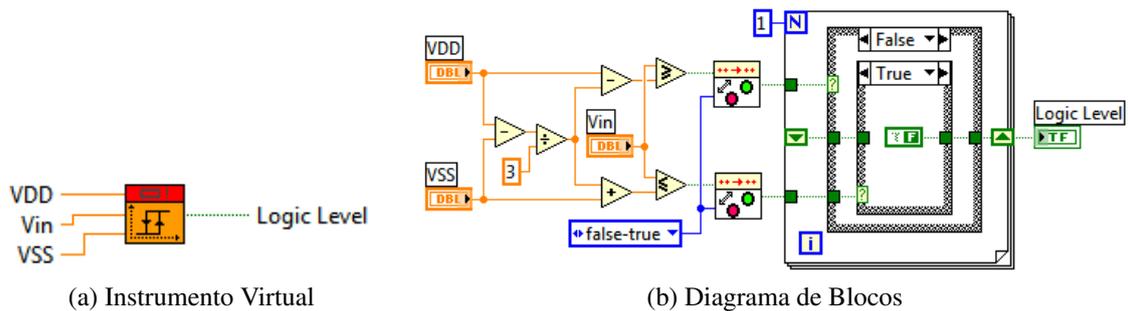


Figura 3.3: *Hysteresis.vi*

3.1.2.2 *Logic Level.vi*

Esta ferramenta serve para a simulação de uma banda proibida nos circuitos digitais. Quando o sinal de entrada está entre V_{SS} e $(V_{DD} - V_{SS})/3$ então se considera um “0” lógico. Se o sinal está entre $2(V_{DD} - V_{SS})/3$ e V_{DD} então se considera como um “1” lógico. Se o sinal fica dentro da banda proibida (i.e. entre $(V_{DD} - V_{SS})/3$ e $2(V_{DD} - V_{SS})/3$) então se escolhe aleatoriamente o nível lógico usando uma distribuição uniforme. Deve-se sempre considerar que $V_{DD} > V_{SS}$. Na figura 3.4 mostra o VI e seu BD correspondente. As entradas e saídas deste VI são iguais que as de *Hysteresis.vi*. Pode ser também considerado um VI *Point-By-Point*.

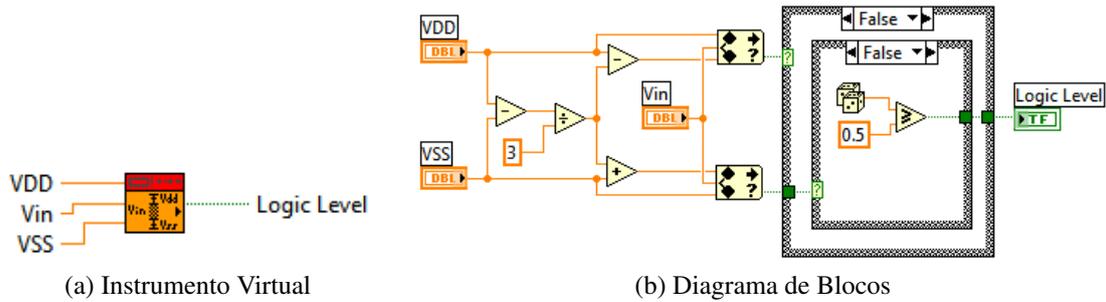


Figura 3.4: Logic Level.vi

3.1.2.3 Capacitive Load.vi

Este VI simula uma carga capacitiva aterrada que detecta as mudanças da tensão e impedância. Este VI se baseia em funções *Point-By-Point* que detecta as mudanças no *array* de entrada **Vin(V)** para poder reiniciar um contador que controla estas mudanças. A entrada **C(F)** serve para definir o valor da capacitância, a **Vo(V)** define a tensão inicial que possui a capacitância e o *array* **Req (Ohm)** serve para definir o valor da resistência ao longo do tempo; cada elemento do **Req (Ohm)** acontece no tempo **dt (s)**. A tensão da a capacitância no tempo é definida pelas equações a seguir:

$$V_C(t) = (V_i - V_f)e^{-t/RC} + V_f \quad \text{Descarrega} \quad (3.7a)$$

$$V_C(t) = (V_f - V_i)(1 - e^{-t/RC}) + V_i \quad \text{Carrega} \quad (3.7b)$$

onde V_f é a tensão final da capacitância, V_i é a tensão inicial e o valor $-t/RC$ é obtido ao pegar, elemento por elemento, o valor do seguinte *array* resultante: **-dt (s)/C(f) Req (Ohm)**. Esta operação se vale do polimorfismo para obter seu resultado (TRAVIS; KRING, 2006). Quando o VI detecta que a tensão de entrada **Vin (V)** vai de baixo para cima ($V_f > V_i$) se usa a equação (3.7b), e quando vai de cima para baixo ($V_f \leq V_i$) se usa a equação (3.7a). O valor $V_C(t)$ é entregue em forma de *array* na saída **Vc (V)**. Na figura 3.5 se mostra o VI desta ferramenta e na figura 3.6 seu correspondente BD.

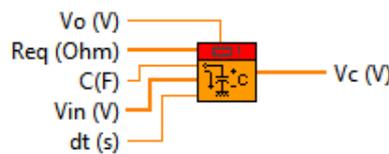


Figura 3.5: Capacitive Load.vi

3.1.3 Portas Lógicas

As portas lógicas utilizadas foram modeladas com a ajuda do *LabVIEW* usando as equações básicas de modelagem de transistores para este fim (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a), onde, cada porta lógica possui pinos de entrada (o inversor só tem a entrada A) nomeados de **VinA**, **VinB**, **CLK** ou **D**, uma entrada de corrente de ruído **Inoise**, uma saída de tensão do resultado da operação **Vout** e uma saída da resistência equivalente **Req**, como se mostra na figura 3.7. Estas portas usam a tecnologia CMOS de 250nm. Na tabela 3.4 se mostram as dimensões dos transistores simulados para as portas lógicas.

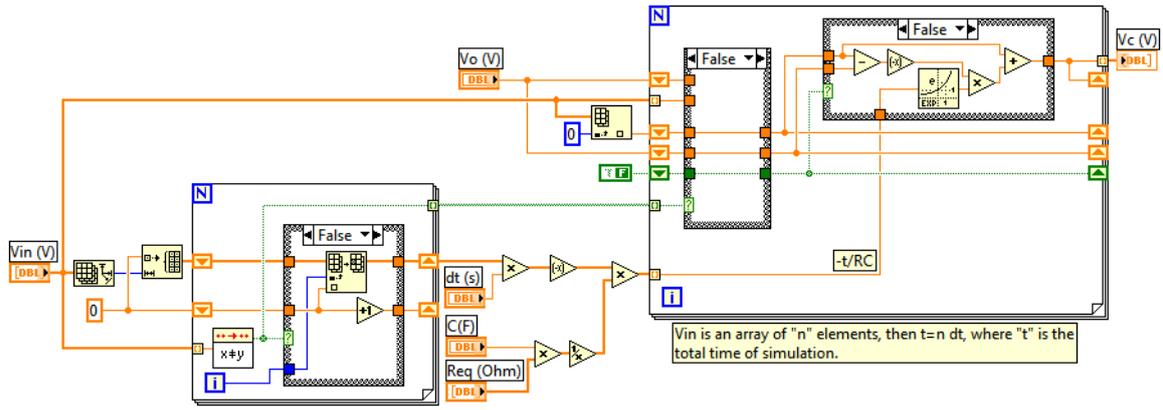


Figura 3.6: BD do VI Capacitive Load.vi

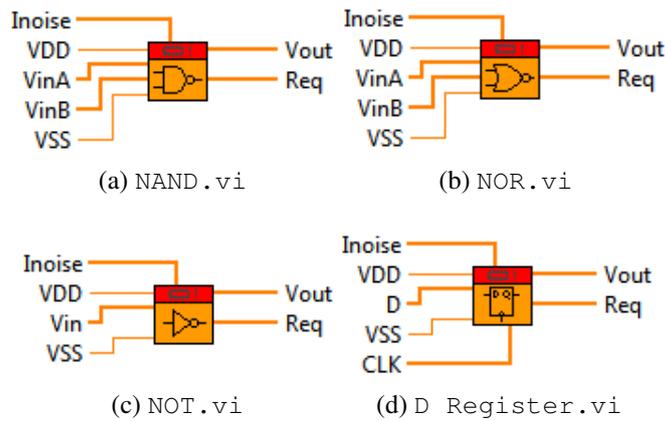


Figura 3.7: VIs das Portas Lógicas simuladas.

Na figura 3.8 mostra o BD do VI NOR.vi da figura 3.7b. Para simular o funcionamento das portas se usa o modelo de disjuntor-resistência mostrado em (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a). Desta forma se pode trabalhar com a tensão de saída **Vout** e a resistência equivalente **Req** que vê a carga no pino de saída da porta. A corrente **Inoise** é inserida em forma de um *array* de números de dupla precisão que representa a onda no tempo e serve para injetar a falha na porta lógica.

Os dois transistores fora da estrutura *FOR* servem para definir a resistência equivalente destes quando estão completamente ligados. Além disso, se tem um mecanismo que discerne se o *array* **Inoise** é vazio ou não. Dentro da estrutura *FOR* se encontram os VIs dos transistores da porta e se procede a comparar a corrente **Id** da cada um com a corrente **Inoise**. Se a corrente da falha é menor ou inexistente, então a porta funciona normalmente, mas se a corrente de falha é maior então acontece uma descida na tensão de saída e uma mudança significativa na impedância de saída. Na figura 3.10 se demonstra estas variações supondo uma tensão de alimentação $VDD = V_{DD}$, fazendo $VSS = V_{SS} = 0V$ (*GND*) e supondo que a resistência dreno-fonte do transistor NMOS é R_n e do transistor PMOS é R_p . Para o caso da figura 3.10a se tem que:

$$V_{out} = 0 \quad (3.8a)$$

$$R_{eq} = \frac{R_n}{2} \quad (3.8b)$$

VI	W_{PMOS} (nm)	W_{NMOS} (nm)
NAND.vi	900	600
NOR.vi	1800	300
NOT.vi	900	300
D Register.vi	1800	600

Tabela 3.4: Larguras das portas dos Transistores Simulados para cada VI criado. O valor da largura da porta é fixo à $L = 250\text{nm}$ e comprimento do oxido de $t_{ox} = 5.60\text{nm}$.

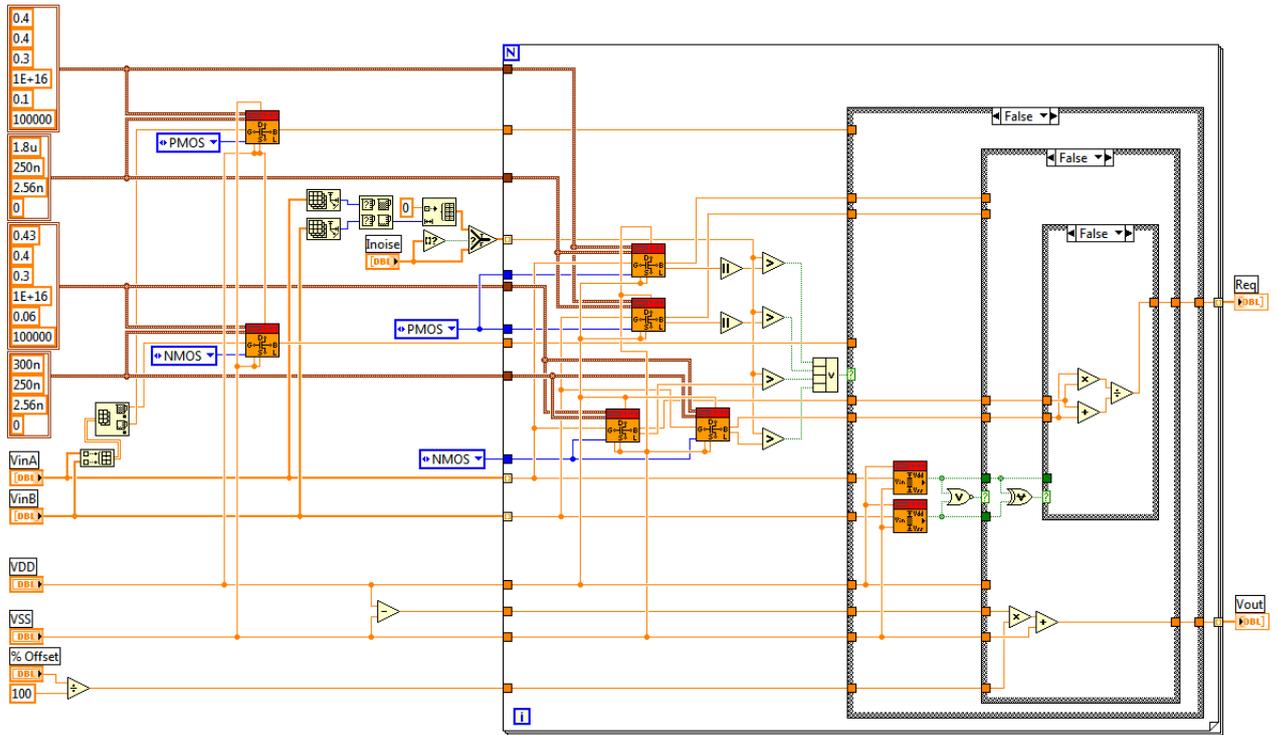


Figura 3.8: BD do VI NOR.vi.

Para o caso que $\mathbf{VinA} = V_{DD}$ e $\mathbf{VinB} = V_{SS}$ ou $\mathbf{VinA} = V_{SS}$ e $\mathbf{VinB} = V_{DD}$ (figuras 3.10b e 3.10c) se tem o seguinte:

$$V_{out} = 0 \quad (3.9a)$$

$$R_{eq} = R_n \quad (3.9b)$$

E para o caso que $\mathbf{VinA} = \mathbf{VinB} = V_{SS}$ mostrado na figura 3.10d se tem que:

$$V_{out} = V_{DD} \quad (3.10a)$$

$$R_{eq} = 2R_p \quad (3.10b)$$

Agora o que acontece numa falha causada pelo SET. O transiente de corrente $I_p(t)$ deve ser suficiente para ligar pelo menos um dos transistores que deveriam estar desligados para essa combinação de entradas, alterando a impedância de saída R_{eq} e também a magnitude da tensão de saída V_{out} . Em geral, para uma porta lógica CMOS de m entradas a quantidade de combinações de ligações de seus transistores não válidas que podem

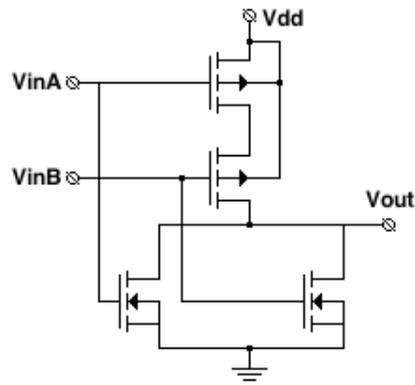


Figura 3.9: Circuito da porta NOR CMOS.

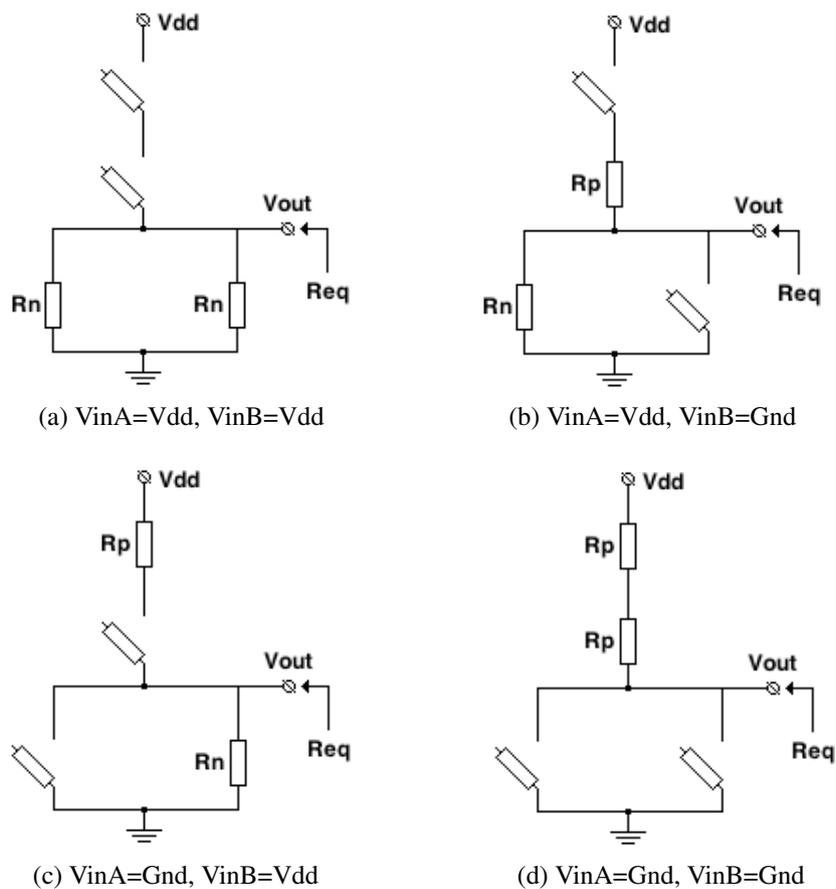


Figura 3.10: Equivalência da porta NOR CMOS em resistências quando é aplicada a seus entradas distintos estímulos válidos.

ser considerados causados por um pulso transiente são $C_{SET}(m) = 3^m - 2^m$, então para nosso caso de $m = 2$ se tem que $C_{SET}(2) = 5$. O pior caso dessas 5 combinações é que todos os transistores desligados passem a estar ligados, fazendo que no circuito aconteça o pior caso de falha. Este é mostrado na figura 3.11 e os valores R_{eq} e V_{out} são afetados da seguinte maneira:

$$V_{out} = \frac{R_n}{4R_p + R_n} (V_{DD} \pm 2R_p I_p(t)) \quad (3.11a)$$

$$R_{eq} = \frac{2R_p R_n}{4R_p + R_n} \quad (3.11b)$$

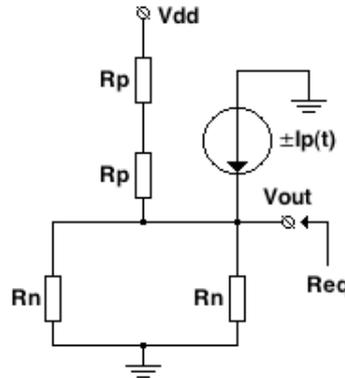


Figura 3.11: Circuito Equivalente da porta NOR CMOS com o pior caso de falha.

Pode-se então perceber a descida da tensão da saída da porta e uma relativa descida da impedância de saída. Estas duas características podem afetar as cargas conectadas na saída da porta lógica. Este mesmo jeito é aplicado nas demais portas desenvolvidas para este trabalho com distintas equações para cada uma delas. Na tabela 3.5 são mostradas as equações de V_{out} e R_{eq} com falha em função de $I_p(t)$, R_p , R_n e V_{DD} . O sinal do $I_p(t)$ é “+” se o SET acontece na rede *Pull-Up* da porta CMOS e “-” se a falha acontece na rede *Pull-Down* (LIN; KIM; LOMBARDI, 2011). Para o Registrador D, a falha acontece na última porta inversora do circuito.

Porta	Tensão de Saída	Resistência Equivalente
NAND	$V_{out} = \frac{4R_n}{R_p + 4R_n} (V_{DD} \pm \frac{R_p}{2} I_p(t))$	$R_{eq} = \frac{2R_p R_n}{R_p + 4R_n}$
NOR	$V_{out} = \frac{R_n}{4R_p + R_n} (V_{DD} \pm 2R_p I_p(t))$	$R_{eq} = \frac{2R_p R_n}{4R_p + R_n}$
NOT	$V_{out} = \frac{R_n}{R_p + R_n} (V_{DD} \pm R_p I_p(t))$	$R_{eq} = \frac{R_p R_n}{R_p + R_n}$
Registrador D	$V_{out} = \frac{R_n}{R_p + R_n} (V_{DD} \pm R_p I_p(t))$	$R_{eq} = \frac{R_p R_n}{R_p + R_n}$

Tabela 3.5: Equações para V_{out} e R_{eq} para modelar as portas desenvolvidas com falhas SEU/SET.

3.1.4 Geradores de Falhas

Para a inserção de falhas nas portas desenvolvidas teve-se que fazer alguns VIs geradores de falhas SET baseados no conjunto de equações (2.1): `SET.vi`, `SETxn.vi` e `RndSETxn.vi`.

3.1.4.1 SET.vi

Esta ferramenta se baseia exclusivamente no conjunto de equações (2.1). Na figura 3.12 se mostra o VI e seu BD. As entradas de dados I_0 (A), t_F (s) e t_R (s) são I_0 , τ_F e τ_R respectivamente. A entrada **Eval. Time** (s) é o tempo máximo para gerar o SET. A entrada **Shifting** serve para deslocar o SET. Este deslocamento l deve estar entre 0 e a quantidade máxima de elementos do *array* de saída do SET $I_p(t)$ (A). Esta quantidade de elementos N depende do tempo máximo de avaliação T (dado pela entrada **Eval. Time** (s)) e do tempo mínimo de cada elemento do *array* δt (dado pela entrada **dt** (s)): $N = T/\delta t$, então $l \in [0, N]$. A carga equivalente do pulso é obtida por uma integração da curva em $I_p(t)$ (A) e usando o valor fornecido em **dt** (s), o resultado é mostrado na saída **Charge** (C). Na figura 3.13 é mostrado o FP desta ferramenta para a geração de um pulso SET para $I_0 = 350\mu\text{A}$, $\tau_R = 10\text{ps}$ e $\tau_F = 100\text{ps}$ em $1\mu\text{s}$ com um $\delta t = 1\text{ps}$. A carga equivalente do pulso é 31.5fC .

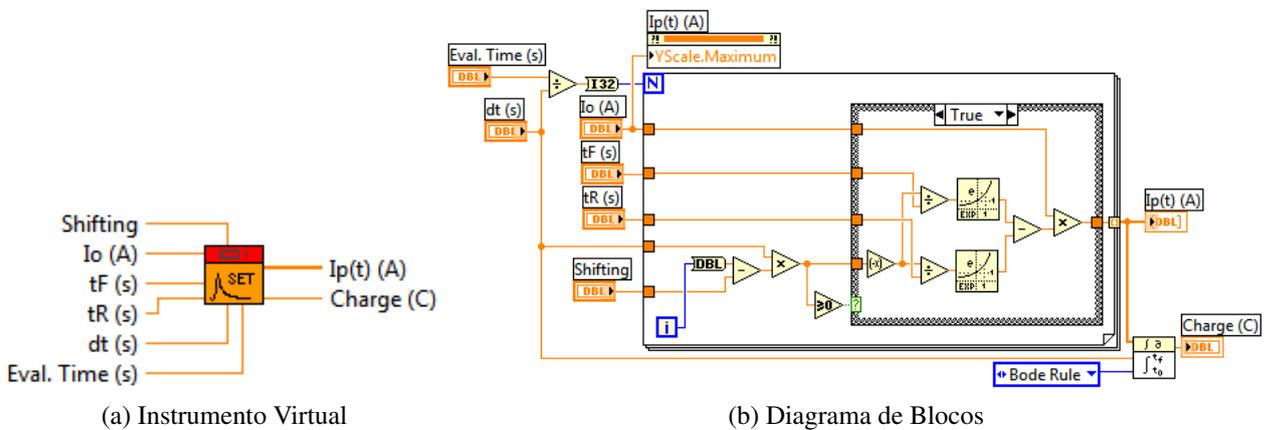


Figura 3.12: SET.vi

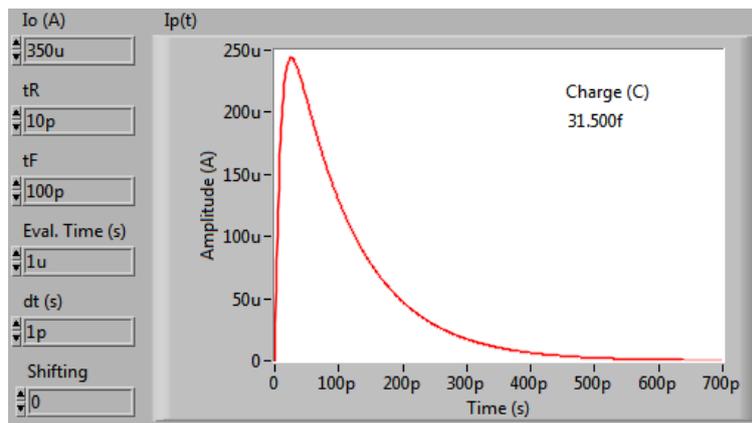


Figura 3.13: FP do SET.vi

3.1.4.2 SETxn.vi

Este VI serve para gerar um conjunto de pulsos SET iguais e espaçados num determinado tempo. Na figura 3.14 se mostram o VI e o BD desta ferramenta a qual se baseia no VI SET.vi da seção 3.1.4.1. A maioria das entradas deste VI são as mesmas que do

SET.vi e cumprem as mesmas funções. A entrada **Delay** insere o atraso de tempo em segundos entre os pulsos e **# Pulses** define quantos pulsos SET se quer gerar. A entrada **Shifting** recebe o deslocamento do primeiro pulso gerado em segundos. A saída **Pulse Train** mostra os pulsos dentro de um *array* de tempo onde cada elemento acontece em um tempo **dt** (s). Na figura 3.15 se mostra o FP desta ferramenta, onde são definidos $I_0 = 350\mu\text{A}$, $\tau_R = 1\text{ps}$, $\tau_F = 15\text{ps}$, um $\delta t = 1\text{ps}$, uma quantidade de pulsos SET de 10 dentro de um tempo total de geração de 10ns, com um deslocamento de 2ns e um tempo entre pulsos de 750ps. Os pulsos podem ser vistos na tela de visualização.

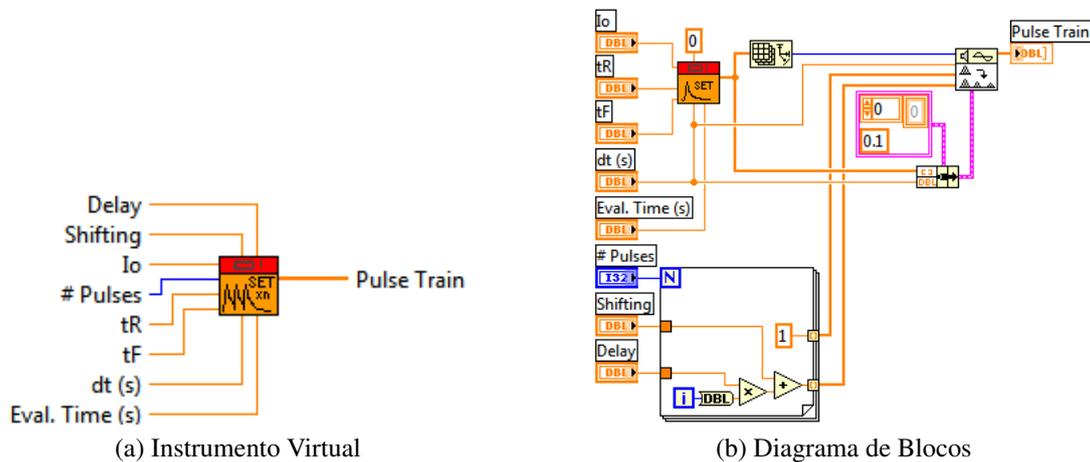


Figura 3.14: SETxn.vi

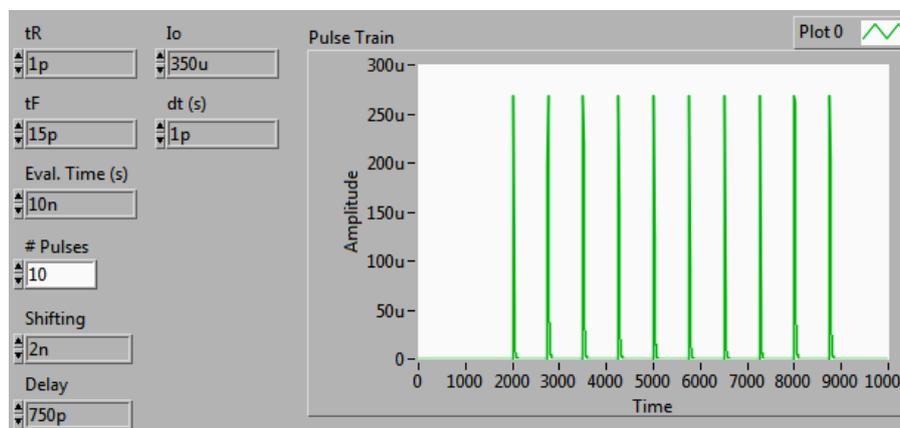


Figura 3.15: FP do SETxn.vi

3.1.4.3 RndSETxn.vi

Esta ferramenta gera um número de pulsos SET determinado pela entrada **# Pulses** de forma aleatória com a distribuição uniforme dentro de um tempo de simulação definido na entrada **Eval. Time** (s). Na figura 3.16 se mostram o VI e o BD desta ferramenta a qual se baseia no VI SET.vi da seção 3.1.4.1. Cada pulso SET está determinado pelas entradas **Io**, **tF** e **tR** e o resultado é mostrado no *array* de saída **Pulse Train** com uma resolução definida pela entrada numérica **dt** (s). É possível também fazer que a amplitude dos pulsos sejam variáveis aleatoriamente com a distribuição uniforme usando o controle booleano **Random Amplitude?** que por padrão está configurado em *FALSE*. Na figura 3.17 se

mostra o FP desta ferramenta, a qual tem definida os mesmos parâmetros de pulso SET mostrados na figura 3.15; o controle booleano **Random Amplitude?** está em *FALSE*.

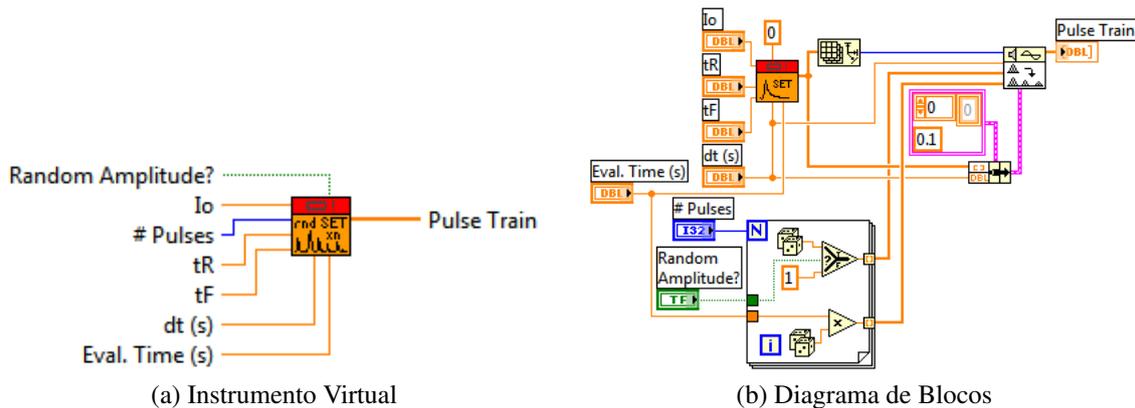


Figura 3.16: RndSETxn.vi

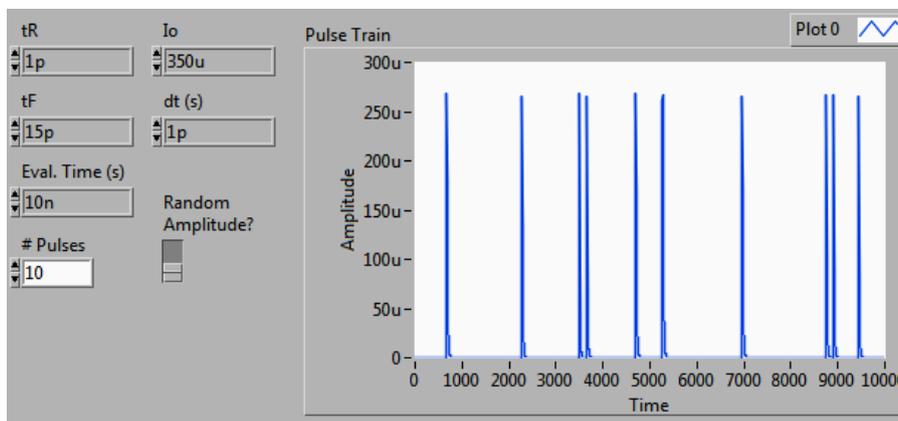


Figura 3.17: FP do RndSETxn.vi

3.1.5 Geradores de Dados e Sinais de Relógio

O pacote de simulação de falhas também possui VIs para gerar dados e sinais de relógio. Estes se baseiam em ferramentas de análise que o LabVIEW fornece.

3.1.5.1 Clock Generator.vi

Este VI é um gerador de relógio ideal (i.e. os tempos de subida e descida são zero). Deve-se indicar o valor da frequência do relógio na entrada **Frequency (Hz)**, além das tensões máxima (entrada **VDD (V)**) e mínima (entrada **VSS (V)**). Com a entrada booleana **Init (F: '0')** se pode controlar se o sinal de relógio inicia com o deslocamento de 0 graus (*FALSE*) ou de 180 graus (*TRUE*). O *cluster* **Info. Sampling** serve para indicar a frequência de amostragem do sinal e o número de amostras. Na figura 3.18 mostra este VI, o qual se baseia em uma das funções de análise que fornece o LabVIEW. As saídas deste VI são o sinal de relógio resultante em formato *waveform* **Clock (V)** no formato de *array* de booleanos **Clock (T/F)**, o tempo total do sinal gerado **Signal Total Time**, a quantidade de amostras geradas **Sample Quantity** (que é o mesmo numero de elementos dos *arrays* de saída) e o tempo de cada amostra **dt**.

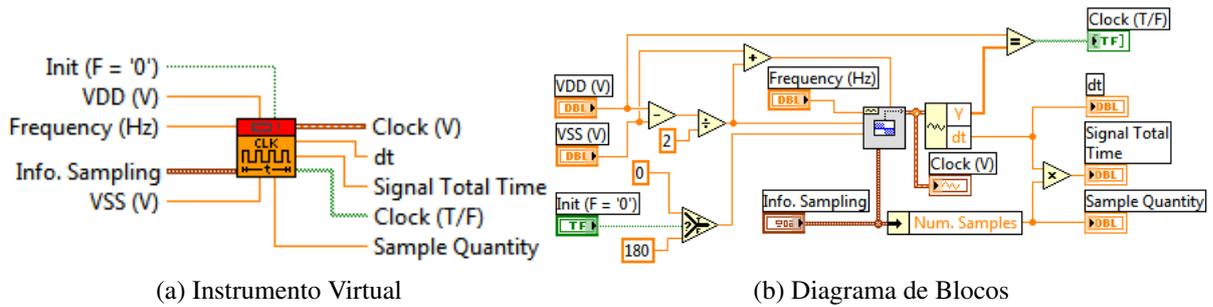


Figura 3.18: Clock Generator.vi

3.1.5.2 Complementary Clock Generator.vi

Esta ferramenta é muito parecida ao VI da seção 3.1.5.1, mas esta gera, além do sinal de relógio normal, um deslocado 180 graus nas saídas **Complementary Clock (V)** e **Complementary Clock (T/F)**. Outra diferença é a entrada **Duty Cycle (%)** que serve para definir o ciclo de trabalho dos relógios gerados. Este VI é mostrado na figura 3.19.

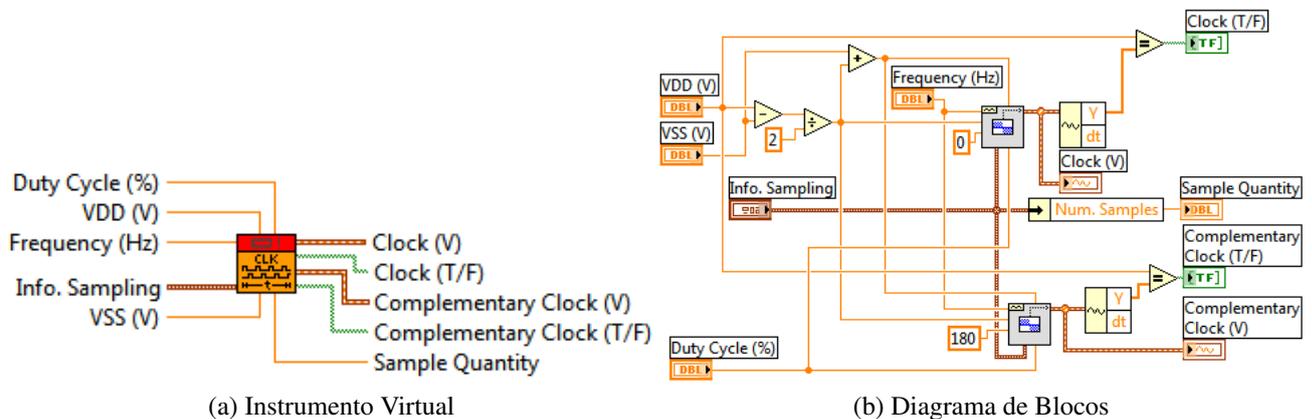


Figura 3.19: Complementary Clock Generator.vi

3.1.5.3 Delta t Clock Generator.vi

Este VI gera três sinais do relógio ideal, mas estão deslocadas um número de graus Δt . Este VI é mostrado na figura 3.20 e é muito parecido com os VIs mostrados nas seções 3.1.5.1 e 3.1.5.2. A entrada **Phase** serve para indicar a quantidade de deslocamento em graus Δt dos sinais gerados. Na figura 3.21 se mostra a maneira do exemplo que entrega este VI se o deslocamento é de 90° para um sinal de $f = 100\text{Hz}$. A figura 3.21a mostra o sinal do relógio gerado sem deslocamento, as outras duas tem deslocamento de $\Delta t = 90^\circ$ e $2\Delta t = 180^\circ$ respectivamente.

3.1.5.4 Real Clock Generator.vi

Este VI gera um sinal de relógio com tempos de subida (t_{rise}), descida (t_{fall}) e tempos de valor máximo (T_h) e mínimo (T_l) como se mostra na figura 3.22. O período do sinal é igual a $T = f^{-1} = T_h + T_l + t_{rise} + t_{fall}$. O sinal gerado tem uma excursão do sinal de V_{DD} até V_{SS} . Este VI é mostrado na figura 3.23. Para gerar este sinal deve-se usar as entradas **Frequency (Hz)**, **trise (s)**, **tfall (s)** e **Th (s)** para defini-lo. Também é necessário indicar

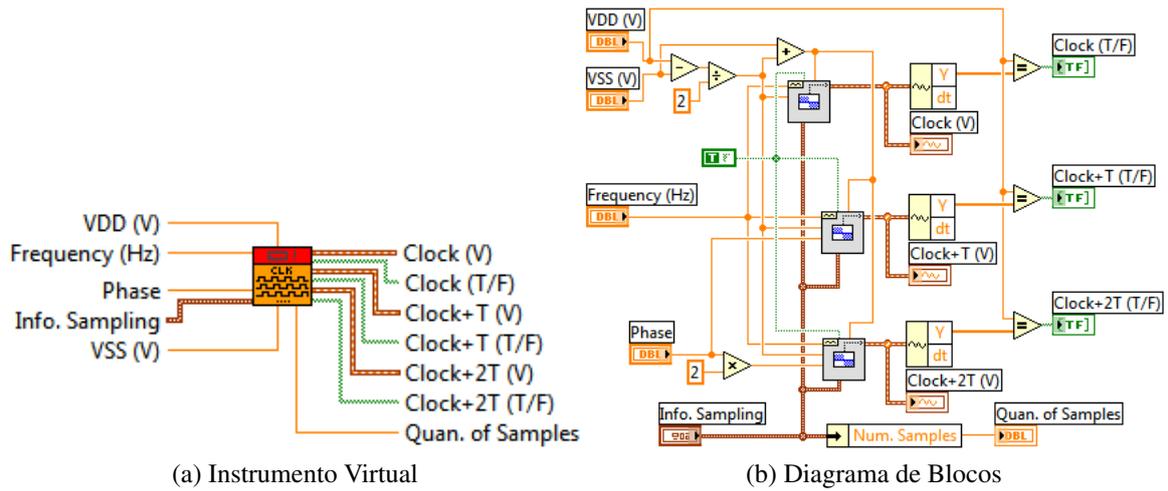


Figura 3.20: Delta t Clock Generator.vi

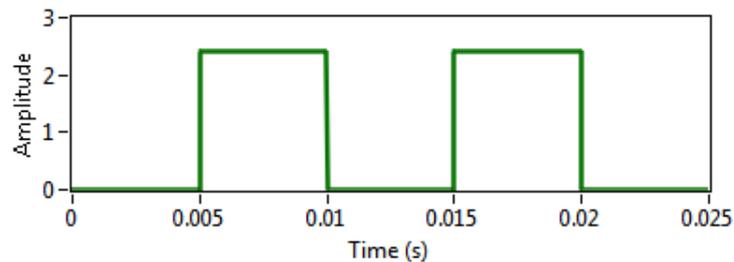
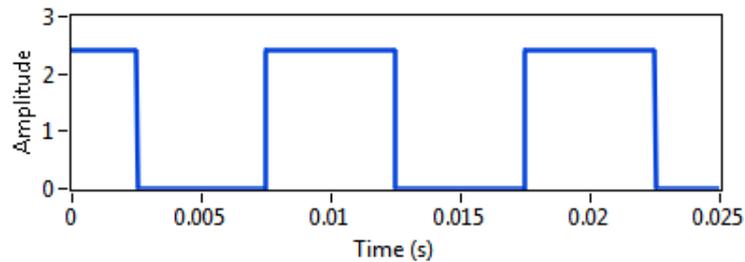
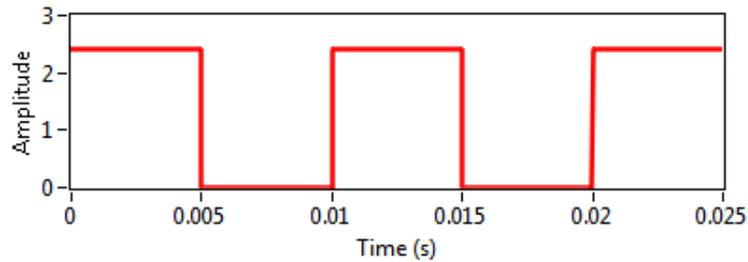


Figura 3.21: Sinais geradas pelo VI Delta t Clock Generator.vi para $f = 100\text{Hz}$.

o tempo total de simulação na entrada **Signal Total Time** e o tempo de cada amostra em **dt**. A entrada booleana **Init (F='0')** serve para inverter o sinal deste relógio respeitando

os tempos definidos nas entradas **trise** (s) e **tfall** (s) para os tempos de subida e descida; o efeito desta entrada é mostrado na figura 3.24.

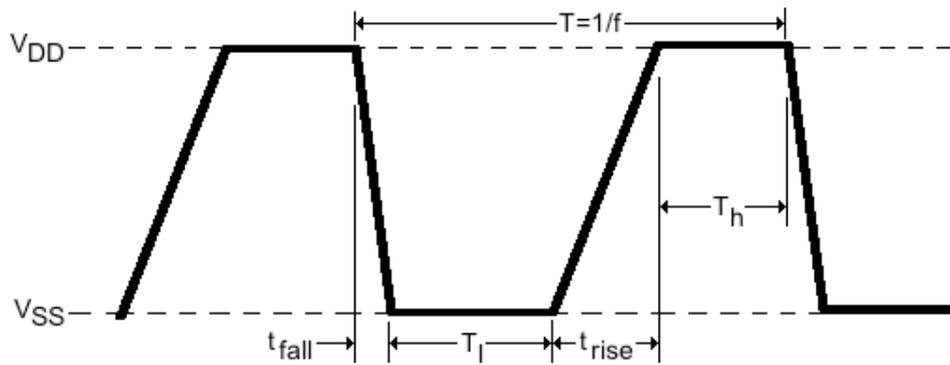
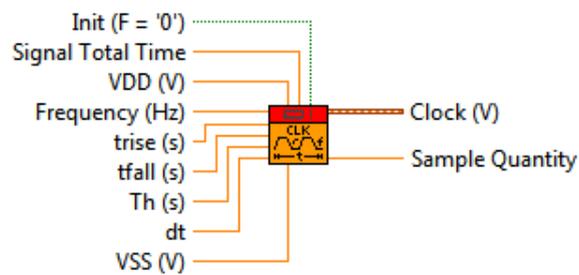
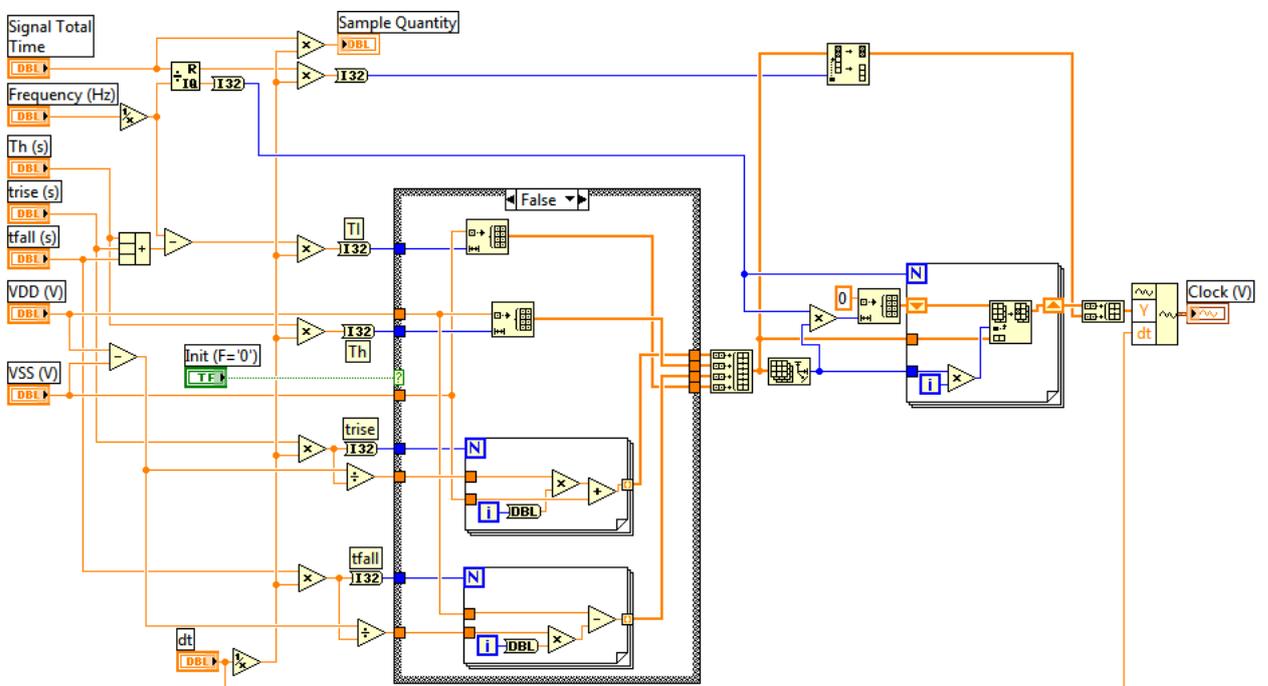


Figura 3.22: Forma de onda de um sinal de relógio real.



(a) Instrumento Virtual



(b) Diagrama de Blocos

Figura 3.23: Real Clock Generator.vi

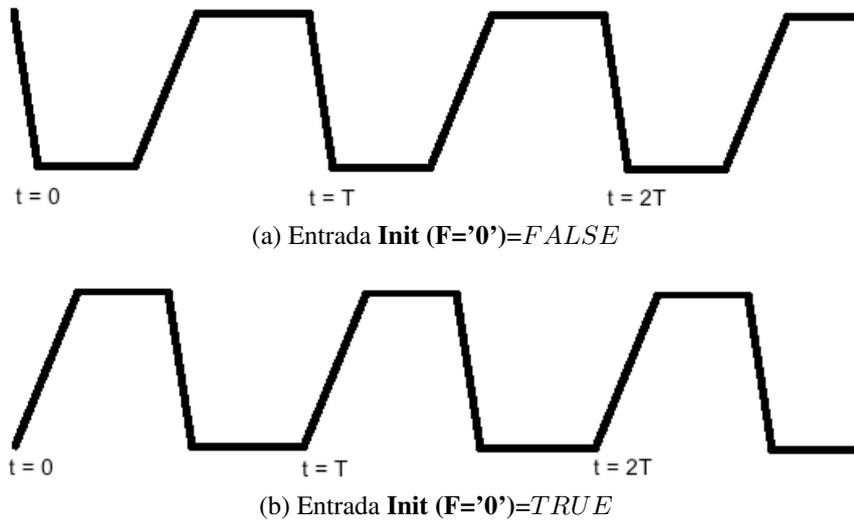
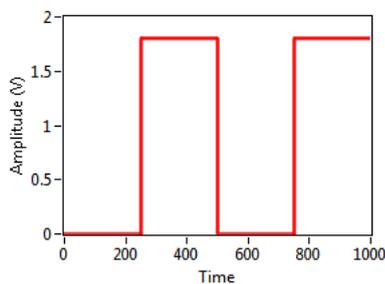


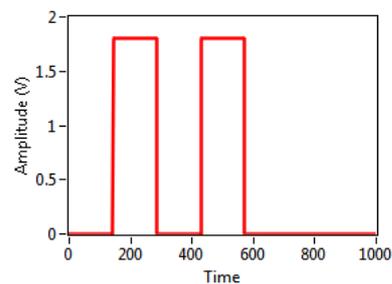
Figura 3.24: Efeito da entrada booleana **Init (F='0')** no Instrumento Virtual Real Clock Generator.vi

3.1.5.5 Digital Pattern.vi

O Digital Pattern.vi é um VI que pega um número inteiro sem sinal de 32 bits em sua entrada **Data** e o converte em um sinal que representa esse número em formato binário nas saídas do *array* **Digital Pattern (V)** (saída de ponto flutuante) e **Digital Pattern (T/F)** (saída booleana). A quantidade de elementos destas *arrays* são definidos pela divisão do valor de entrada **Eval. Time (s)** e o valor da entrada **dt (s)**. Uma vez obtendo a quantidade de bits do número é realizada a divisão do espaço de simulação (definido pela quantidade na entrada **Eval. Time (s)**) pela quantidade de bits do número. O bit menos significativo se localiza à esquerda do espaço de simulação. Às vezes se precisa de mais bits para representar um número concatenando bits “0” na esquerda; esta função é permitida colocando a entrada booleana **Zeros at Left (F: NO)** para *TRUE* e determinando o número de bits zero que se quer agregar na entrada **Num. of Zeros at Left**. Na figura 3.25 pode-se ver o uso deste recurso. A figura 3.26 mostra este VI, onde pode-se notar que ele contém uma saída do tipo *array* booleano e uma saída de tipo *array* de ponto flutuante.



(a) **Data=10** ($1010_{(2)}$) com **Num. of Zeros at Left=0**



(b) **Data=10** ($1010_{(2)}$) com **Num. of Zeros at Left=2**

Figura 3.25: Efeito de Zeros na esquerda de um dado '10' decimal no VI Digital Pattern.vi

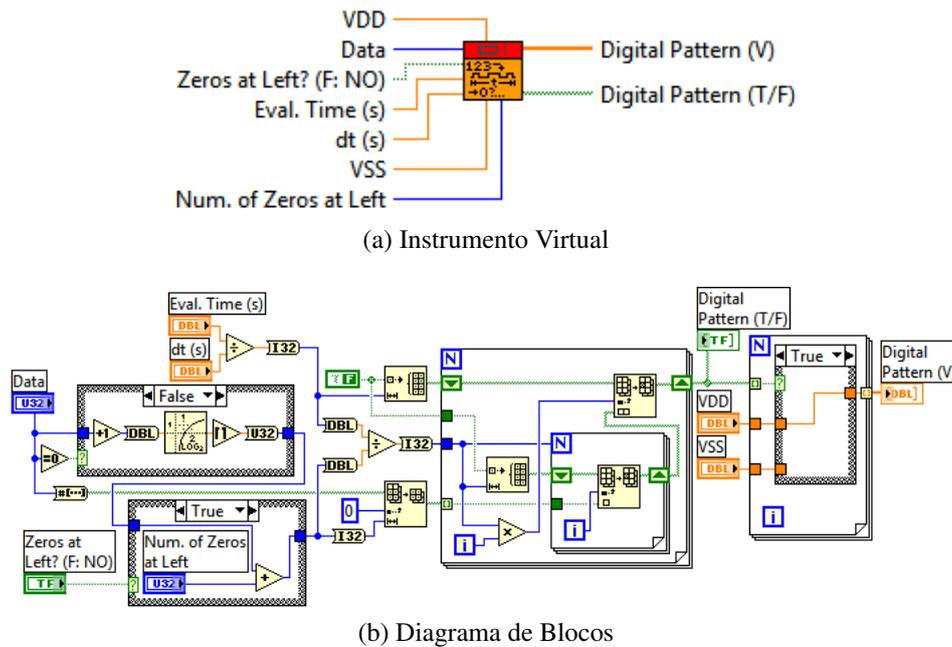


Figura 3.26: Digital Pattern.vi

3.2 Simulação em Lógica Booleana

Este conjunto de VIs é baseado na utilização de *arrays* de booleanos (possuindo apenas os valores de *TRUE* ou *FALSE*). Esta simulação tem por objetivo a análise comportamental de técnicas de correção de falhas e sua evolução enquanto se variam seus parâmetros, como é a frequência ou deslocamento de relógio. Alguns destes VIs precisam do apoio de estruturas contidas no *LabVIEW*, como estruturas do tipo *FOR* e *WHILE*, que possuem registradores de deslocamento (*Shift-Register*) (JOHNSON; JENNINGS, 2006; TRAVIS; KRING, 2006). Estes registradores são mostrados na figura 3.27. Cada Registrador de Deslocamento consta de uma entrada e uma saída e servem para realimentar dados gerados anteriormente. Na tabela 3.6 é mostrado o resumo dos VIs que formam este conjunto de ferramentas.

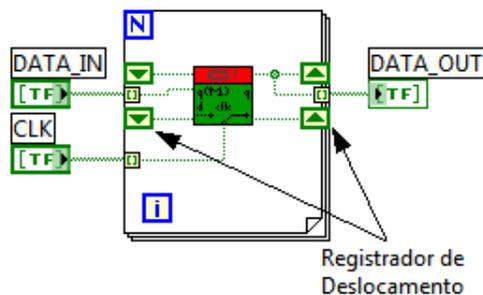


Figura 3.27: Registradores de Deslocamento em uma estrutura *FOR*.

Ícone do VI	Nome do VI	Comentário
	Incomplete Register.vi	Simula um Registrador de 1 bit.
	Incomplete Register.vi	Simula um Registrador de “n” bits como um número inteiro.
	Random Digital Pattern.vi	Simula um padrão pseudo-aleatório Gauss ou Poisson.
	Bit Flip.vi	Simula uma troca de valor no bit de dados.
	Voter.vi	Simula uma função de Votador pela Maioria.

Tabela 3.6: Resumo de Funções do pacote de Simulação em Lógica Booleana.

3.2.1 Registradores

Estes VIs simulam o funcionamento dos registradores de forma comportamental. Eles são *Incomplete Register.vi* e *Incomplete Bus Register.vi*. Estes precisam usar *Shift Registers* para “completar” suas funções.

3.2.1.1 *Incomplete Register.vi*

Este VI serve para simular o Registrador tipo D de um único bit. Na figura 3.28 se mostra esta ferramenta tanto seu ícone VI representativo como seu BD. É possível configurar este registrador para que funcione como *Flip-Flop* (detecção da borda do sinal do relógio) ou como *Latch* (detecção do nível do relógio), para isto se deve usar a entrada booleana **Edge/Level Selector (Edge: F)**, se a entrada é *FALSE* o registrador vai a estar configurado como *Flip-Flop*, se e *TRUE* o registrador se comportará como um *Latch*. As entradas **D** e **CLK** servem para os dados e o relógio respectivamente. Os conjuntos de entrada/saída **CLK(t-1)/CLK(t)** e **Q(t-1)/Q(t)** devem ser usadas com registradores de deslocamento dentro de estruturas *FOR* e *WHILE* como é mostrado na figura 3.27, onde a saída e tomada de **Q(t)**.

3.2.1.2 *Incomplete Bus Register.vi*

Esta ferramenta é muito parecida a aquela mostrada na seção 3.2.1.1. A diferença é o tipo de dados para **D** e o conjunto **Q(t-1)/Q(t)**, os quais são do tipo inteiro sem sinal de 32 bits que representa o bus de dados. Na figura 3.29 se mostra o VI.

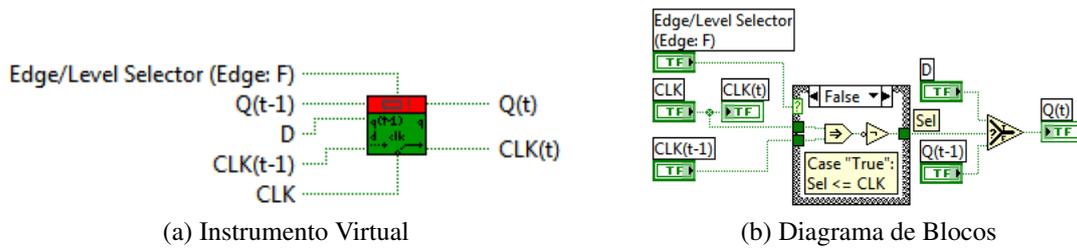


Figura 3.28: Incomplete Register.vi

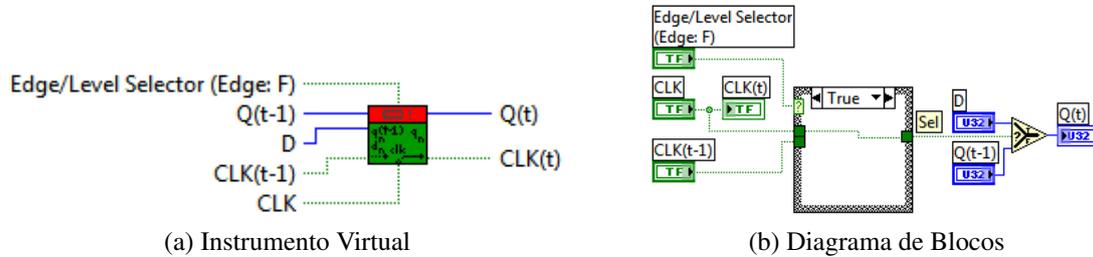


Figura 3.29: Incomplete Bus Register.vi

3.2.2 Outras Funções

Há também alguns outros VIs úteis para a simulação de circuitos digitais tolerantes a falhas de forma comportamental.

3.2.2.1 Random Digital Pattern.vi

Neste gerador de ruído aleatório precisa de ser indicado o número de amostras a gerar, o tipo de ruído e o parâmetro crítico da distribuição de probabilidade. Na figura 3.30 é mostrado este VI. A entrada **Num. of Samples** serve para inserir o numero de amostras do sinal de ruído. **Noise Type** é uma entrada de tipo enumerado que serve para escolher entre um padrão de Ruído Branco Gaussiano ou um padrão de Ruído de Poisson, este último está programado como padrão (FAURE; VELAZCO; PERONNARD, 2005). **Distribution Critical Value** é uma entrada que permite dar o valor crítico para a geração do padrão de ruído; este valor é 0.5 por padrão. Para um Ruído com Distribuição de Gauss o valor crítico é o Desvio Padrão σ , por enquanto para a Distribuição de Poisson é a Média dos valores $\bar{\mu}$. A entrada booleana **Hysteresis (T: YES)** serve para comutar entre o uso do VI *Hysteresis.vi* (mostrado na seção 3.1.2.1) e o uso do ruído puro sem o *buffer* com histerese. O uso da histerese é importante só para poder enxergar melhor os resultados da simulação. A saída é o *array* booleano **Random Digital Pattern (T/F)**.

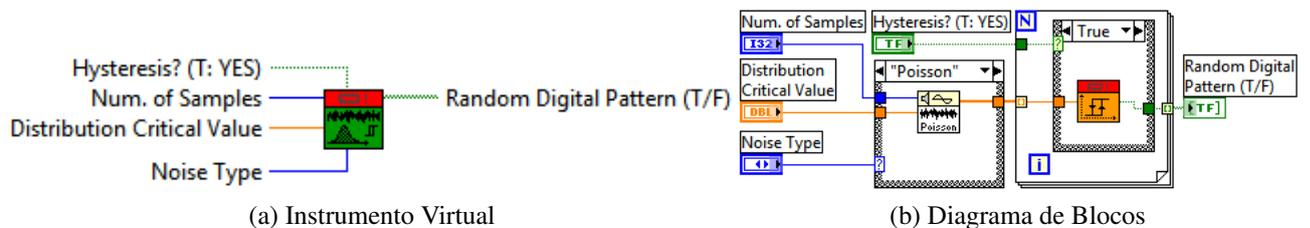


Figura 3.30: Random Digital Pattern.vi

3.2.2.2 *Bit Flip.vi*

Este VI, o qual é mostrado na figura 3.31, serve para mudar o valor booleano que é colocado na entrada $Q(t)$. Se a entrada booleana **Fault** é *FALSE*, então o valor da entrada passa à $Q(t)$ w. **Fault** sem alteração, mas se **Fault** é *TRUE* então o valor da entrada passa à $Q(t)$ w. **Fault** invertida.

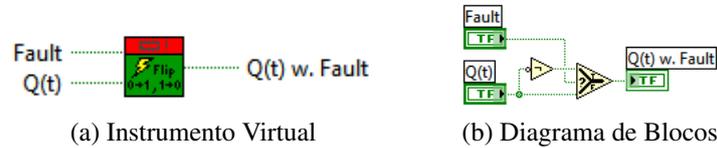


Figura 3.31: Bit Flip.vi

3.2.2.3 *Voter.vi*

Esta ferramenta só implementa a função lógica de “votador pela maioria”. Esta função tem por entradas **P1**, **P2** e **P3**; e por saída a **Out**. Este VI é mostrado na figura 3.32.

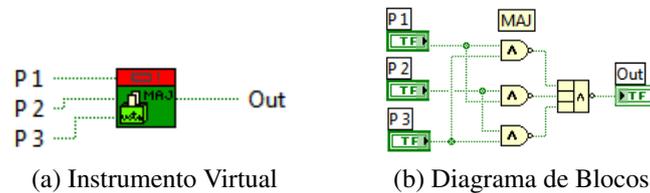


Figura 3.32: Voter.vi

3.2.2.4 *Outras Funções Lógicas*

O LabVIEW fornece muitas funções booleanas para implementar outros circuitos lógicos de maneira comportamental, como multiplexadores, codificadores, somadores e mais. É possível combinar estes com estruturas especiais e/ou funções polimórficas do LabVIEW para melhorar seu desempenho (TRAVIS; KRING, 2006; JOHNSON; JENNINGS, 2006).

3.3 Simulação Analógica

O pacote possui também alguns circuitos analógicos (Como espelhos de corrente, transistores em configurações especiais, etc.) para poder simular os efeitos das falhas transientes neles. Estes VIs se baseiam nas funções *Point-By-Point* (INSTRUMENTS, 2004), as quais são boas para simulação em tempo real e são desenvolvidas usando o conceito da Função Global Funcional (JOHNSON; JENNINGS, 2006). Esta parte do pacote ainda está em teste, motivo pelo qual se decidiu fazer ferramentas só para este tipo de simulação. Na tabela 3.7 mostra o resumo destes VIs.

Ícone do VI	Nome do VI	Comentário
	Transistor MOS PtByPt.vi	Simula um Transistor MOSFET ponto por ponto.
	Tank PtByPt.vi	Simula um Circuito Tanque <i>LC</i> ponto por ponto.
	NMOS Current Mirror PtByPt.vi	Simula um Espelho de Corrente NMOS.
	NMOS Cross Pair PtByPt.vi	Simula uma configuração de transistores NMOS em par cruzado para gerar uma resistência negativa.
	NMOS Active Resistor.vi	Simula um transistor NMOS em configuração de resistência ativa.
	PMOS Active Resistor.vi	Simula um transistor PMOS em configuração de resistência ativa.
	Capacitive Load.vi	Simula uma Capacitância aterrada ponto por ponto.

Tabela 3.7: Resumo de Funções do pacote de Simulação em Lógica Booleana.

3.3.1 Simulação do Transistor

A simulação do transistor neste pacote se baseia no modelo quadrático. Neste modelo a corrente do dreno I_D tem diferentes equações para modelar as diferentes regiões de operação. Estas equações são as seguintes (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a; CAVERLY, 2007):

$$I_{Dsth} = I_M \frac{W}{L} e^{\frac{V_{GS} - V_{th}}{n\phi_T}} \left[1 - e^{\frac{V_{DS}}{\phi_T}} \right] \quad (3.12a)$$

$$I_{Dres} = \frac{\mu_n \epsilon_{ox}}{t_{ox}} \frac{W}{L} \left[(V_{GS} - V_{th}) V_{DS} - \frac{V_{DS}^2}{2} \right] \quad (3.12b)$$

$$I_{Dsat} = \frac{\mu_n \epsilon_{ox}}{2t_{ox}} \frac{W}{L} (V_{GS} - V_{th})^2 (1 + \lambda V_{DS}) \quad (3.12c)$$

onde I_{Dsth} é a corrente de dreno na região sub-limiar, I_{Dres} é a corrente de dreno na região resistiva e I_{Dsat} é a corrente de dreno na região de saturação. Na equação 3.12a,

I_M é a corrente específica de sub-limiar, n é a constante de forma da tensão sub-limiar, $\phi_T = kT/q$ é a voltagem térmica ($T = 300$ K), μ_n é a mobilidade das cargas no canal (KANO, 1998). Nas equações 3.12b e 3.12c, W e L são a largura e comprimento do canal do transistor, ϵ_{ox} é a permissividade do óxido do silício, t_{ox} é a espessura do óxido, V_{th} é a tensão de limiar, V_{GS} é a tensão porta-fonte e V_{DS} é a tensão dreno-fonte. A tensão de limiar V_{th} é definida na equação 3.1.

O VI `Transistor MOS PtByPt.vi` é mostrado na figura 3.33. Este VI está baseado nas equações 3.12 e 3.1. Para configurar este VI se precisam de dois conjuntos de parâmetros: Os Parâmetros Geométricos (**Geom Parameters**) e os Parâmetros Tecnológicos (**Tech Parameters**), os quais são representados por entradas dos *clusters* no VI. Na tabela 3.8 pode-se ver os componentes do *cluster Tech Parameters* na ordem como devem ser inseridos no VI para tecnologia de $250nm$. Os parâmetros geométricos são os mesmos mostrados na figura 3.2 da seção 3.1.1.

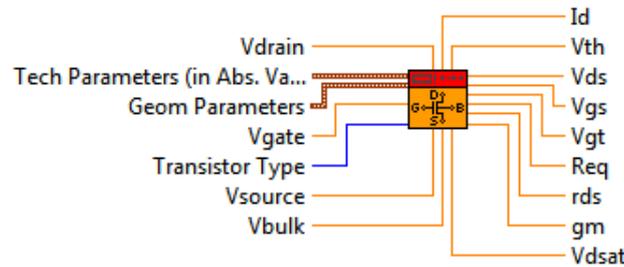


Figura 3.33: Instrumento Virtual Transistor MOS PtByPt.vi.

Parametro	Simbolo	PMOS	NMOS	Unidade
Voltagem de Limiar à $V_{GS} = 0$	V_{th0}	-0.40	0.43	V
Efeito Corpo de Transistor	γ	0.40	0.40	$V^{0.5}$
Potencial do Trabalho	ϕ_F	-0.30	0.30	V
Dopagem de Doadores	N_D	1×10^{16}	1×10^{16}	cm^{-3}
Dopagem de Aceitadores	N_A	1×10^{14}	1×10^{14}	cm^{-3}
Comprimento de Canal	λ	-0.10	0.06	V^{-1}
Velocidade da Saturação	v_{sat}	100000	100000	m/s
Densidade de Portadores no Si	n_i	1.5×10^{10}	1.5×10^{10}	cm^{-3}
Forma da Tensão Sub-Limiar	n	1.49	1.49	—

Tabela 3.8: Parâmetros Físicos da Tecnologia de $250nm$ para os transistores de Simulação Analógica (RABAEY; CHANDRAKASAN; NICOLIĆ, 2003a).

3.3.2 Circuitos Úteis

Estes VIs servem para criar outros circuitos analógicos. Fez-se isto para evitar a condição de código “*spaghetti*” (“bagunça” de fios de conexão) muito comum no código feito no LabVIEW (TRAVIS; KRING, 2006). Também se fez isto para criar código hierárquico e modular que melhore a leitura e confiabilidade.

3.3.2.1 *Tank PtByPt.vi*

Este VI simula um circuito tanque de maneira comportamental. Se baseia no circuito de indutância e capacitância (diodo varicap) em paralelo. Seu VI é mostrado na figura

3.34a. Para sintonizar o Tanque deve-se usar a entrada de dupla precisão **Vtune** com um número entre 1 e 1.8 V para definir a capacitância do Tanque entre 500 pF a 1 pF; este valor é definido pela seguinte equação:

$$C = 1123.75 - 623.75 V_{tune} \quad (3.13)$$

onde C é o valor da capacitância em picofarads e $V_{tune} \in [1, 1.8]$ V. Se V_{tune} está fora deste intervalo, então o valor é forçado para as cotas superior ou inferior do intervalo dependendo de que limite seja ultrapassado.

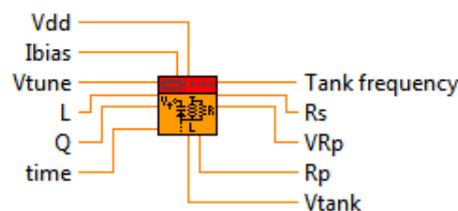
A frequência do circuito tanque f em Hertz e definida pela seguinte equação:

$$f = \frac{1}{2\pi\sqrt{LC}} \quad (3.14)$$

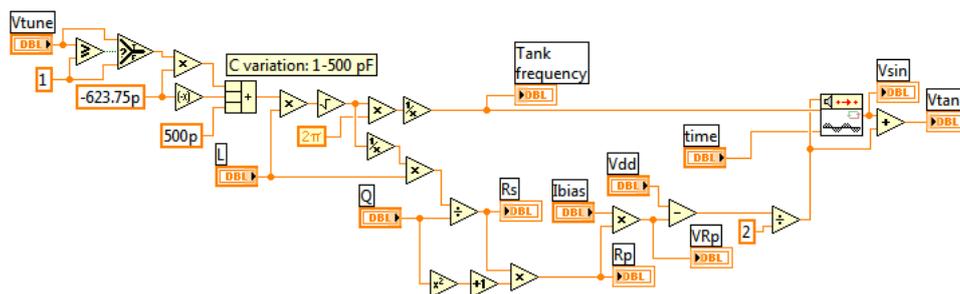
onde L é o valor da indutância na entrada **L** no VI. O valor f é possível de obter na saída **Tank Frequency** do VI e serve para gerar o sinal senoidal na saída **Vtank** com uma amplitude e deslocamento **Vdd**. A entrada **time** é necessária para gerar o sinal em **Vtank** devido a que o gerador precisa de um valor de tempo para a geração de um ponto do sinal em cada iteração. As saídas **Rs** e **Rp** servem para mostrar o valor da resistência serial R_s da indutância L e seu equivalente paralelo R_p (CAVERLY, 2007). Estas resistências dependem do fator do qualidade do Circuito Tanque Q definida na entrada **Q** do VI:

$$R_s = \frac{2\pi f L}{Q} \quad (3.15a)$$

$$R_p = R_s(1 + Q^2) \quad (3.15b)$$



(a) Instrumento Virtual



(b) Diagrama de Blocos

Figura 3.34: Tank PtByPt.vi

3.3.2.2 NMOS Current Mirror PtByPt.vi

Esta ferramenta simula um Espelho de Corrente NMOS com uma resistência ativa PMOS que o polariza ele (mostrada na seção 3.3.2.4). Seu VI e Esquemático é mostrado

na figura 3.35. A corrente **Iset**, a tensão **Vout** e um dimensionamento adequado dos transistores são necessários para fixar a corrente **Iout**. A saída **Vref** mostra a tensão porta-fonte dos transistores do espelho. A saída **Req** serve para ver a resistência de dreno-fonte r_{ds} dos transistores do espelho. Para fixar os parâmetros tecnológicos e geométricos dos transistores e necessário preencher os *clusters* para os transistores NMOS e para o transistor PMOS que serve de Resistência Ativa. Estes *clusters* são iguais aos expostos na seção 3.3.1.

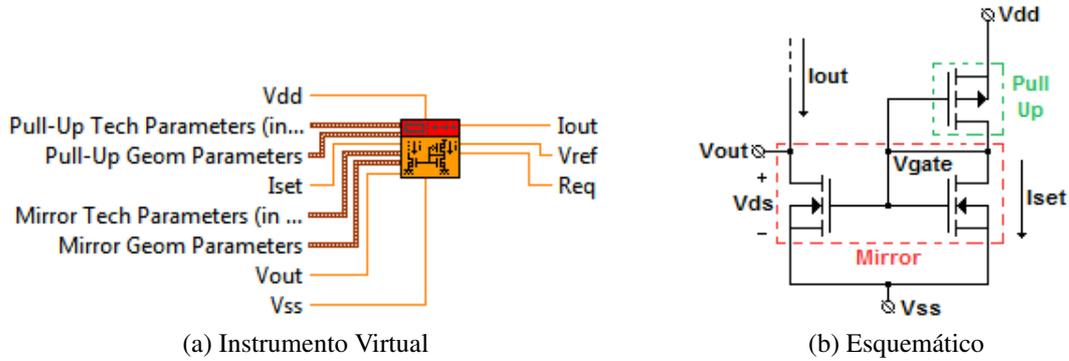


Figura 3.35: NMOS Current Mirror PtByPt.vi

3.3.2.3 NMOS Cross Pair PtByPt.vi

Esta ferramenta simula o comportamento de um par de transistores cruzados para gerar uma resistência negativa (CAVERLY, 2007). Esta arranjo de transistores é a base dos osciladores. Neste VI é possível inserir uma falha SET e olhar o comportamento de um circuito baseado neste arranjo (como os VCOs). Na figura 3.36 é mostrado o VI e o Esquemático desta ferramenta a qual é usado com dois VIs *Tank PtByPt.vi* relatados na seção 3.3.2.1. O funcionamento de um par cruzado de transistores como oscilador se baseia no Critério de Barkhausen (detalhado no anexo B). Para que o circuito oscile deve-se cumprir que:

$$\frac{1}{R_{p1}} + \frac{1}{R_{p2}} - g_{m1} - g_{m2} = 0 \quad (3.16)$$

onde R_{p1} e R_{p2} são as resistências paralelas dos tanques conectados acima e g_{m1} e g_{m2} são as transcondutâncias de cada transistor do par cruzado. Num princípio, estes circuitos são desenvolvidos para cumprir com este critério.

Para este caso em particular, a transcondutância de cada um dos transistores $g_m = g_{m1} = g_{m2}$ é igual a:

$$g_m = \sqrt{2K \frac{W}{L} I} \quad (3.17)$$

onde $K = \mu_n \varepsilon_{ox} / t_{ox}$ e I é a corrente do dreno dos transistores. Então, para que este circuito deixe de funcionar como deve quando se insere um transiente, não deve cumprir-se a igualdade da equação (3.16). Isto pode ser feito variando o valor da transcondutância de algum dos dois transistores do par cruzado. Como o SET é um transitório de corrente que afeta a corrente de dreno dos transistores e a sua quantidade total de portadores (da qual depende a mobilidade de acordo com o conjunto de equações (2.24)), então é possível escrever a equação (3.17) da seguinte forma:

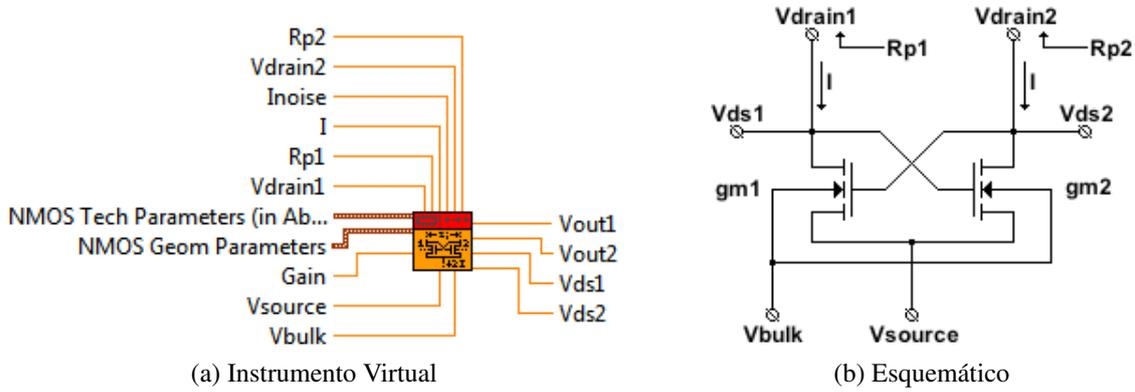


Figura 3.36: NMOS Cross Pair PtByPt.vi

$$g_m = \sqrt{2 \frac{\mu_n(N_{var})\epsilon_{ox}}{t_{ox}} \frac{W}{L} (I + I_p(t))} \quad (3.18)$$

onde $\mu_n(N_{var})$ é a mobilidade que depende de $N_{var} = N \cdot I / (I + I_p(t))$ e $I_p(t)$ é a corrente do SET. Se a corrente $I_p(t)$ é inexistente, então se volta para a equação (3.17) e o par cruzado funcionará normalmente.

Para este VI, só um transistor do par cruzado vai ser afetado pelo SET, neste caso será o transistor com transcondutância g_{m2} . Então, além das mudanças causadas pelo SET na mobilidade e na corrente do dreno, estas também afetarão as tensões de dreno de cada transistor da seguinte forma:

$$V_{dfi} = \frac{g_{m1} + g_{m2}}{g_{m1}^2 (R_{p1} + R_{p2})} \cdot V_{di} \quad (3.19)$$

onde V_{di} é a tensão de dreno sem falha do transistor i e V_{dfi} é a tensão do dreno com falha do transistor i . Esta formulação foi obtida de forma empírica usando os dados fornecidos pela simulação em Spectre do CMOS VCO mostrada em (CHEN et al., 2003). Esta condição só acontece se o Critério de Barkhausen mostrado na equação (3.16) não se cumprir.

No VI mostrado na figura 3.36a, **Rp1** e **Rp2** são as resistências em paralelo associados ao circuito tanque da seção 3.3.2.1, além de **Vdrain1** e **Vdrain2** que são as tensões de saída de cada tanque. A entrada **I** é a corrente de dreno de cada transistor e **Inoise** é a corrente transiente do SET $I_p(t)$. As entradas **Vsource** e **Vbulk** servem para definir as tensões de fonte e corpo dos transistores respectivamente. A entrada **Gain** serve para definir o fator do ganho do par cruzado (por padrão é 10). As saídas **Vds1** e **Vds2** mostram a voltagem entre dreno e fonte dos transistores e as saídas **Vout1** e **Vout2** mostram as voltagens anteriores, só que com um deslocamento indicado na entrada **Vsource**. Na figura 3.37 se mostra o BD deste VI.

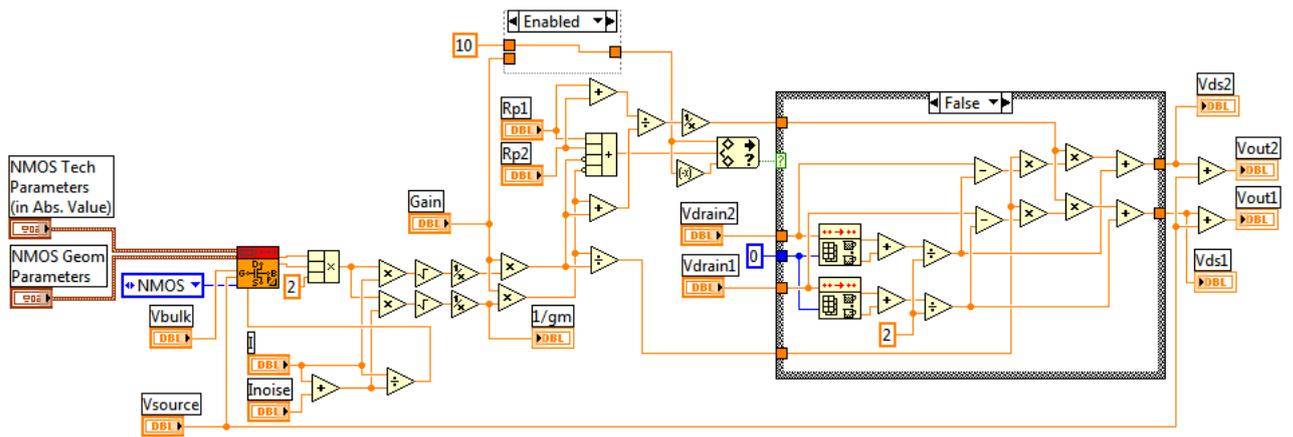


Figura 3.37: BD do VI NMOS Cross Pair PtByPt.vi

3.3.2.4 Resistências Ativas

Um importante motivo pelo qual não se usam resistores passivos num CI é a área que estes precisam no *wafers*. A alternativa é o uso de resistores ativos, os quais são feitos com transistores MOS (CAVERLY, 2007). Na figura 3.38 se mostra as configurações dos transistores NMOS e PMOS como cargas resistivas ativas, as quais se baseiam no uso dos transistores como fontes de corrente.

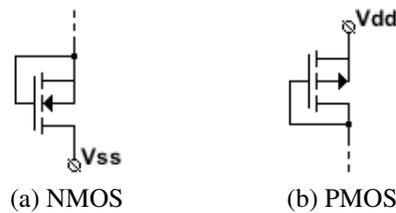


Figura 3.38: Configurações do Resistores Ativos com transistores MOS

Estes transistores permanecem na região de saturação descrita na equação (3.12c), só que esta muda ligeiramente pela configuração devido a que $V_{GS} = V_{DS}$ neste caso:

$$I_D = \frac{\mu_n \epsilon_{ox}}{2t_{ox}} \frac{W}{L} (V_{DS} - V_{th})^2 (1 + \lambda V_{DS}) \quad (3.20)$$

Considerando o valor de comprimento do canal λ suficientemente pequeno tal que $\lambda V_{DS} \rightarrow 0$, então o transistor MOS é similar a um diodo de característica I-V quadrática. Mas considerando λ como parte da equação, deve-se resolver a seguinte equação cubica para obter V_{DS} :

$$\lambda V_{DS}^3 + (1 - 2\lambda V_{th}) V_{DS}^2 + (\lambda V_{th}^2 - 2V_{th}) V_{DS} + (V_{th} - I_D \xi_{sat}^{-1}) = 0 \quad (3.21)$$

onde $\xi_{sat} = \frac{\mu_n \epsilon_{ox}}{2t_{ox}} \frac{W}{L}$ é a constante tecnológica do transistor em saturação. Para a resolução da equação 3.21 se usam funções do LabVIEW especializadas em resolução de equações polinomiais (INSTRUMENTS, 2004).

Na figura 3.39 se mostram os VIs para a simulação dos resistores ativos. A entrada **Input Value Type** define se o valor da entrada de definição **Input Value** é de corrente ou tensão. A entrada **Vsource** define o valor da fonte de alimentação (V_{ss} para NMOS

e V_{dd} para PMOS de acordo com a figura 3.38). A saída I_d informa sobre a corrente do dreno e V_{drain} sobre a tensão do terminal do dreno. V_r entrega o valor da tensão entre os terminais de fonte e dreno.

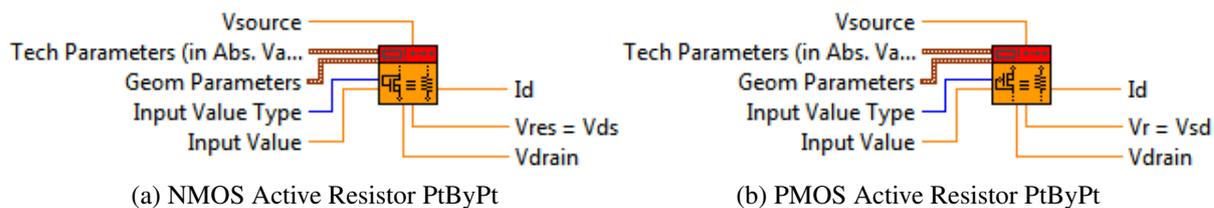


Figura 3.39: VIs do Resistores Ativos

3.3.2.5 Capacitive Load PtByPt.vi

Este VI é muito parecido a aquele exposto na seção 3.1.2.3. A diferença está nos tipos de dados que usam como entrada e o uso destes. Esta ferramenta usa entradas numéricas de dupla precisão, o qual o faz ideal para usa-se em tempo real devido a sua natureza de Global Funcional (JOHNSON; JENNINGS, 2006), típico de um VI *Point-By-Point*.

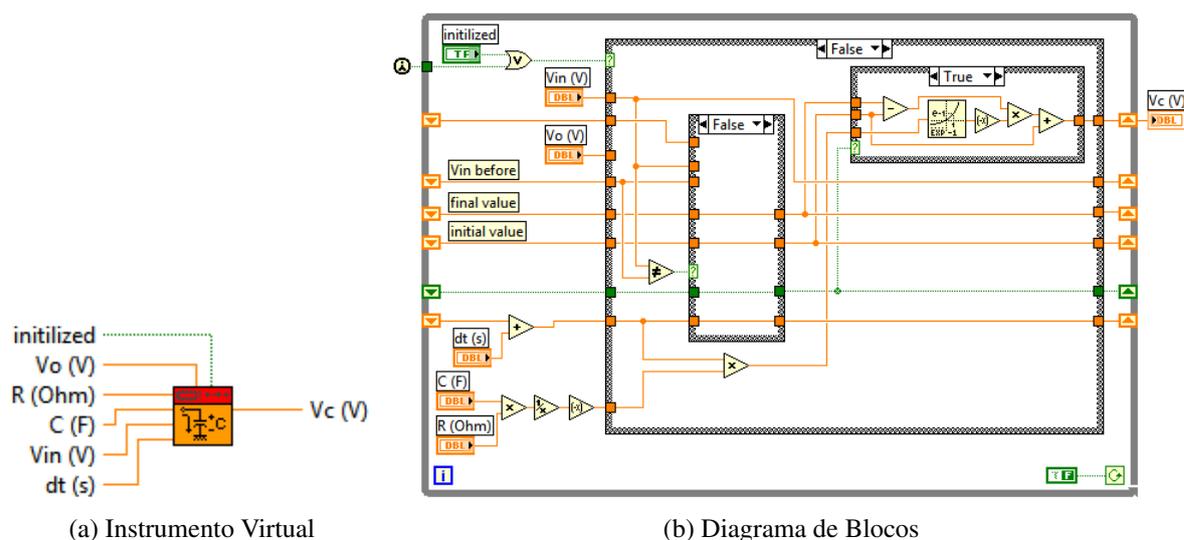


Figura 3.40: Capacitive Load PtByPt.vi

4 SIMULAÇÕES COM AS FERRAMENTAS DESENVOLVIDAS E RESULTADOS

O propósito de fazer simulações em LabVIEW é de criar um VI o mais similar a um circuito usando as ferramentas desenvolvidas neste pacote. As falhas são inseridas usando as ferramentas criadas para este fim. Os dados coletados são mostrados nos FP da correspondente simulação.

4.1 Simulação de um Inversor com SET

Na figura 4.1 pode-se ver um circuito formado por um inversor e uma carga capacitiva afetada pelo SET, o qual pode ser mal interpretado como um “1” quando deveria ser um “0” lógico. La fonte do sinal é uma onda quadrada ideal de $f = 1\text{MHz}$ e a tensão de alimentação e de 2.4V para este inversor de tecnologia de 250nm. A capacitância de carga é de 200fF que simula o *fan-out* do Inversor.

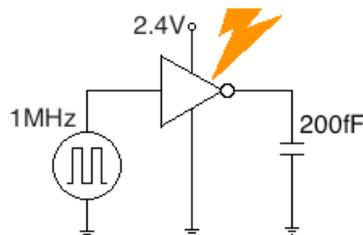


Figura 4.1: Circuito Inversor com SET.

Na figura 4.2 se mostra o BD desta simulação, onde o sinal da entrada é gerado por uma função *Square Waveform* própria do LabVIEW. Em esta simulação é usada uma taxa de amostragem de $F_s = 60$ Giga-amostras por segundo para gerar um sinal de $\#s=1$ Mega-amostras para uma simulação de $\#s/F_s = 16.7\mu s$ (lembre-se o critério do Nyquist: $f \leq F_s/2$) (TRAVIS; KRING, 2006; JOHNSON; JENNINGS, 2006; INSTRUMENTS, 2004).

A tensão resultante e a resistência na saída do inversor afeitam linearmente ao tensão da saída. O efeito pode-se ver na figura 4.3. O pulso SET pode-se interpretar como um “1” lógico se a carga capacitiva é trocada por uma porta lógica ou registrador.

Após realizar esta simulação se fez a mesma simulação usando o HSPICE e o Cosmoscope para fazer uma comparação entre os resultados no intervalo do tempo entre 2,9ns e 3,75ns. Na figura 4.4 pode-se ver esta comparação, a qual evidencia a falta da modelagem de capacitâncias parasitas neste trabalho, algo que o SPICE faz. Estas capacitâncias influenciam tanto na magnitude do sinal como em seu atraso total. A figura 4.5 mostra

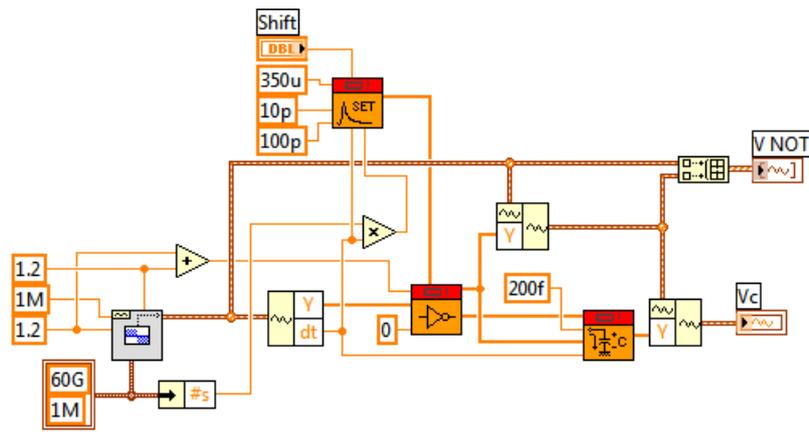
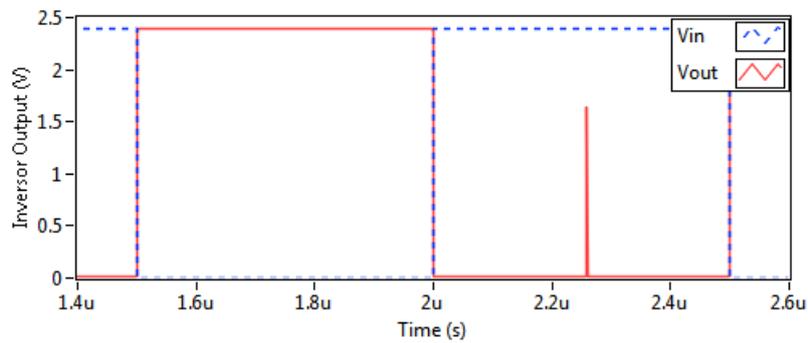
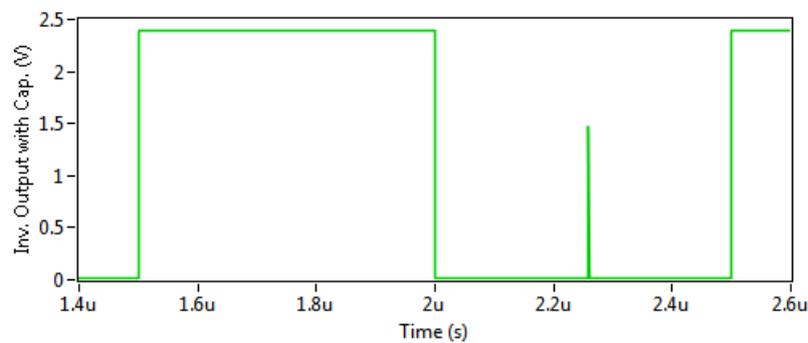


Figura 4.2: BD da Simulação do Circuito Inversor com SET.



(a) Entrada/Saída do Inversor com SET



(b) Saída do Inversor com o capacitor de $200fF$

Figura 4.3: Simulação no LabVIEW do Inversor com SET da figura 4.2.

o pulso na saída do circuito, o qual está magnificada para poder comparar os resultados das simulações. Para este caso se usou uma carga de $200fF$. No apêndice A se relatam a biblioteca usada e o *netlist* desta simulação.

4.2 Simulação de um Inversor com Múltiplos SETs

Esta simulação se fez usando como padrão a simulação mostrada na seção 4. Para este caso se usou um sinal de entrada do relógio ideal de $f = 10MHz$ com uma taxa de 100 Giga-amostras por segundo e uma quantidade de amostras de $\#s = 20000$, o suficiente para mostrar até dois períodos da onda quadrada. Para gerar os pulsos SET se usou a ferramenta `RndSETxn.vi` para que estes pulsos tenham amplitudes e locações

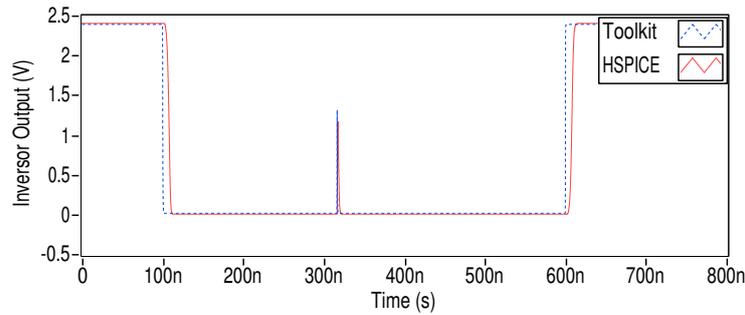


Figura 4.4: Comparação entre a Simulação no LabVIEW e a Simulação com HSPICE. Se mostra só uma parte da simulação com sua referência de tempo deslocada a zero.

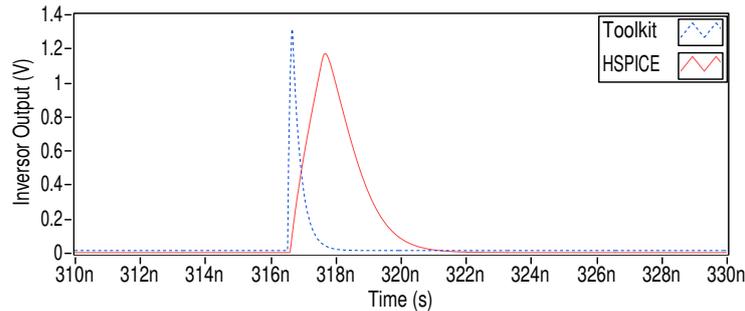


Figura 4.5: Comparação entre a Simulação no LabVIEW e a Simulação com HSPICE a maior escala do pulso resultante na saída do circuito.

no tempo aleatórias. A quantidade de pulsos se fixou em 20 e o protótipo de pulso SET se fixou em $I_0 = 350\mu\text{A}$, $\tau_R = 10\text{ps}$ e $\tau_F = 100\text{ps}$ de acordo com a equação (2.1a). Na figura 4.6 se mostra o BD desta simulação e na figura 4.7 pode-se ver estes pulsos SET gerados pelo VI RndSETxn.vi para esta simulação.

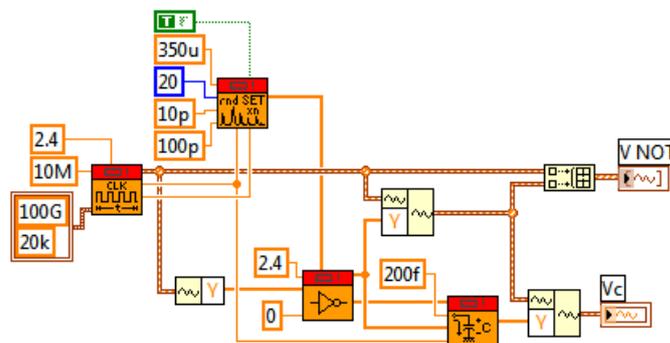


Figura 4.6: BD da Simulação de um Circuito Inversor com Múltiplos SETs.

Tal como se discutiu na seção 3.1.3, a variação da impedância da saída quando se tem falha acontecerá em ambos transistores do inversor para simular o pior caso de falha. Isto pode-se ver no resultado da simulação mostrada na figura 4.8. Os pulsos afetam tanto aos “1” como aos “0” de maneira similar (XUEYAN; LIYUN; JINMEI, 2011). O capacitor da saída de $200fF$ trata de filtrar os transientes, os quais tem distintos comprimentos ao longo do tempo.

Também se fez uma simulação com a fonte de relógio real relatada na seção 3.1.5.4. Para este caso se deveu mudar o BD da simulação para acomodar a ferramenta Real

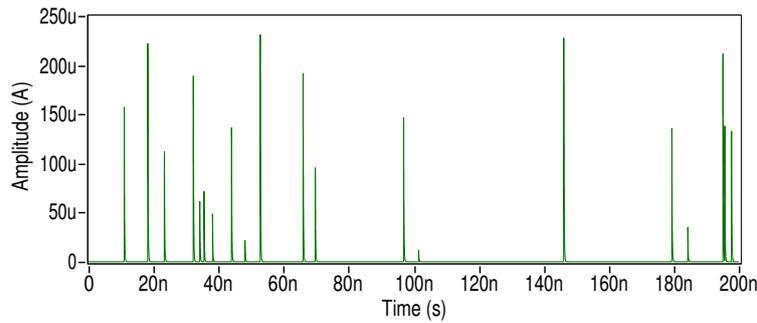


Figura 4.7: Pulsos SET gerados pelo VI `RndSETxn.vi` para esta simulação.

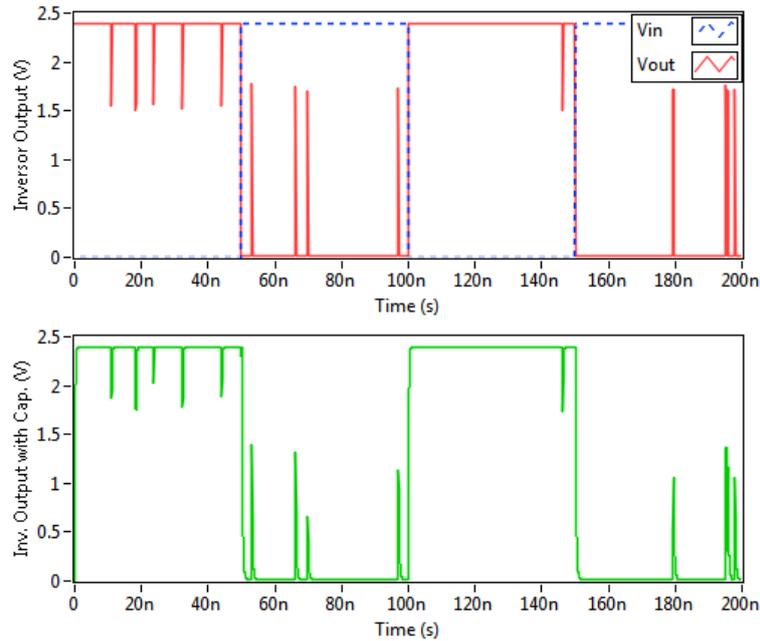


Figura 4.8: Simulação no LabVIEW do Inversor com Múltiplos SET e relógio ideal. O gráfico de cima mostra a entrada/saída do Inversor e o de baixo é a saída do inversor com o capacitor de 200fF.

`Clock Generator.vi`. Os parâmetros deste sinal são: tempo de subida $t_r = 5\text{ns}$, tempo de descida $t_f = 5\text{ns}$, tempo em nível alto $T_h = 45\text{ns}$, longitude da amostra $dt = 10\text{ps}$ e frequência do sinal $f = 10\text{MHz}$. Na figura 4.9 se mostra o BD desta simulação a qual usa o mesmo conjunto de pulsos SETs mostrados na figura 4.7.

Na figura 4.10 pode-se ver os resultados desta simulação. De fato este relógio com $t_r = t_s = 5\text{ns}$ é considerado pouco realista devido à sua lentidão. Este relógio também causa um fenômeno de *jitter* não desejado devido o natureza da modelagem do nível de ruído ou *bandgap* proibido pela ferramenta `Logic Level.vi`: esta modelagem é aleatória com uma distribuição uniforme entre $(V_{DD} - V_{SS})/3$ e $2(V_{DD} - V_{SS})/3$. A longitude deste *jitter* é de $t_{jr} = t_r/3$ para as bordas de subida e de $t_{jf} = t_f/3$ para a descida. Este *jitter* observado nas simulações deve ser pesquisado para ver se coincide com as modelagens de *jitter* conhecidas e seus possíveis efeitos na amostragem (SHIMANOUCHI, 2001; AZEREDO-LEME, 2011). Na figura 4.11 se mostra a mesma simulação, só que com um relógio mais real com $t_r = t_f = 1\text{ns}$ e $T_h = 49\text{ns}$ para simular o sinal de $f = 10\text{MHz}$. Neste caso o *jitter* gerado pelo VI `Logic Level.vi` é imperceptível,

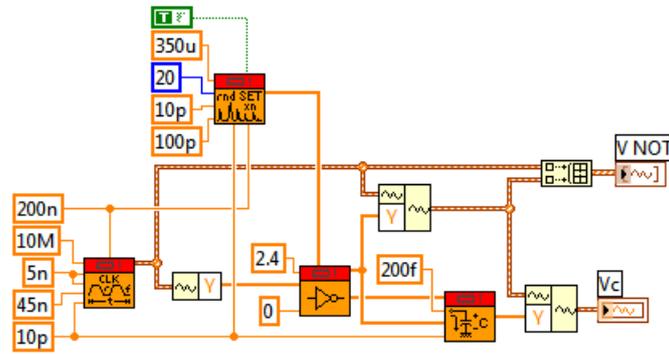


Figura 4.9: BD da Simulação de um Circuito Inversor com Múltiplos SETs e relógio real.

mas ainda existe. Isto é uma consequência de não modelar as capacitâncias parasitas dos transistores das portas lógicas, o qual também modela o atraso destas.

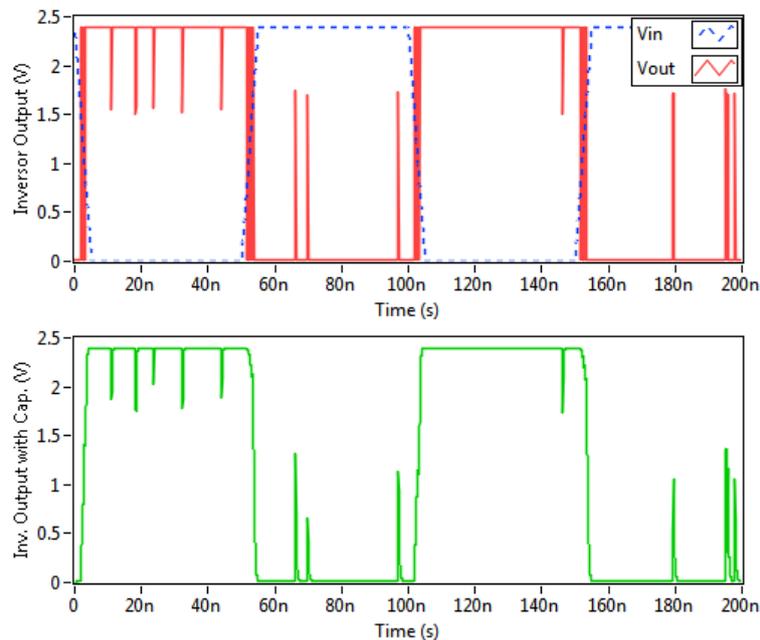


Figura 4.10: Simulação no LabVIEW do Inversor com Múltiplos SETs com um relógio real de $f = 10\text{MHz}$ com $t_r = t_f = 5\text{ns}$ e $T_h = 45\text{ns}$. O gráfico de cima mostra a entrada/saída do Inversor e o de baixo é a saída do inversor com o capacitor de 200fF .

4.3 Simulação de Circuito Lógico com SET

Na Figura 4.12 é apresentado um circuito combinacional onde uma de suas portas lógicas (neste caso uma *NAND*) é afetada pelo SET.

Na Figura 4.13 se mostra o diagrama de blocos deste circuito simulado. O relógio *CLK* é gerado pela ferramenta *Clock Generator.vi*; para este caso usou-se um relógio de 1MHz com uma taxa de amostragem de 60 Giga-amostras por segundo. Os dados de entrada dos circuitos são gerados usando o *Digital Pattern.vi*, um VI da biblioteca de ferramentas desenvolvidas, que pega um número inteiro e o converte para binário para gerar um sinal serial no tempo. O VI *SET.vi* gera um transitório de corrente modelada pela equação 2.1a.

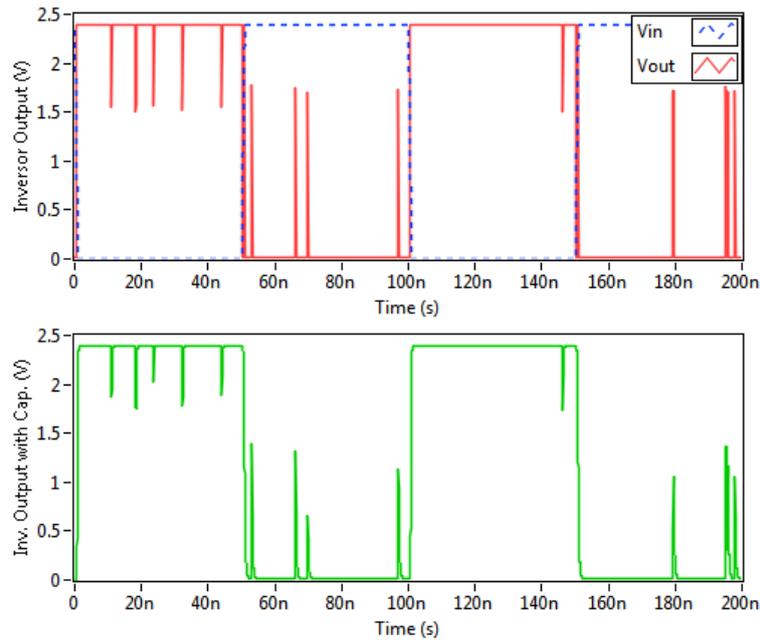


Figura 4.11: Simulação no LabVIEW do Inversor com Múltiplos SETs com um relógio real de $f = 10\text{MHz}$ com $t_r = t_f = 1\text{ns}$ e $T_h = 49\text{ns}$. O gráfico de cima mostra a entrada/saída do Inversor e o de baixo é a saída do inversor com o capacitor de 200fF .

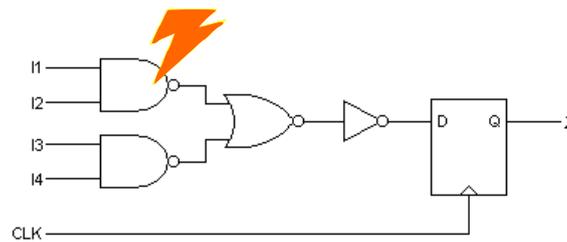


Figura 4.12: Circuito Combinacional afetado pelo SET.

O Painel Frontal da simulação é mostrado na Figura 4.14. Este painel contém dois gráficos de resposta da simulação, um com *NANDout+SET* e outro com *Combinational Logic* e um controle para deslocar o pulso SET (*SET Shift*). No gráfico *NANDout+SET*, pode-se ver o pulso SET sob o resultado da função lógica implementada antes do *Flip-Flop D*, do qual é localizado em 500ns . No gráfico 4.14b, pode-se ver o sinal do relógio e o resultado na saída do circuito, onde ele ficou alterado devido a presença do pulso SET inserido durante a execução da simulação. Quando o SET coincide com a borda de subida do relógio este é lido como um “1” lógico na entrada do registrador, gerando com isto um erro no sinal de saída.

4.4 Simulação do Bit-Flip em uma Célula de Memória Triplicada

O *Bit-Flip* pode acontecer em vários locais de um circuito, comumente encontrados em registradores e nos dados de entrada do circuito. Sua natureza é totalmente aleatória e pode acontecer a qualquer momento. Na figura 4.15 é mostrado o circuito que usa a técnica TMR explicada na seção 2.4.1. Cada um dos três registradores usa o mesmo relógio. Quando um e só um dos registradores é afetado pelo *Bit-Flip* o *Voter* procurará

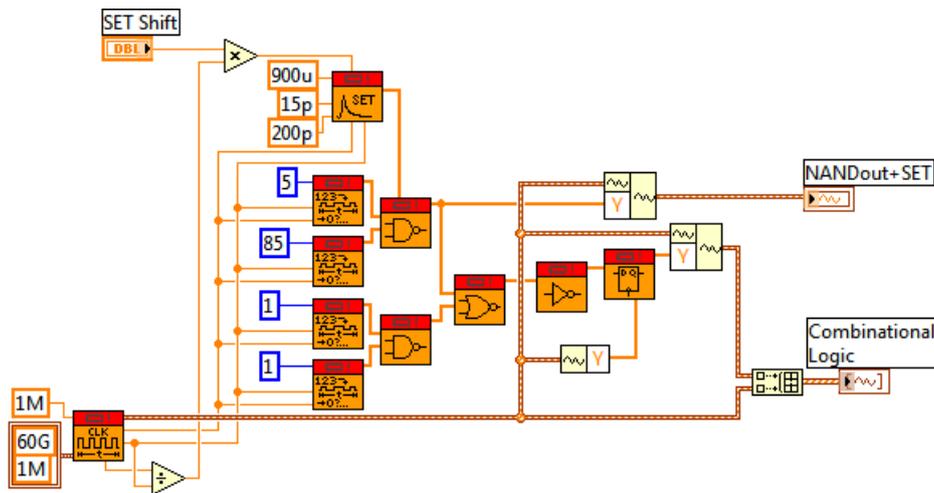
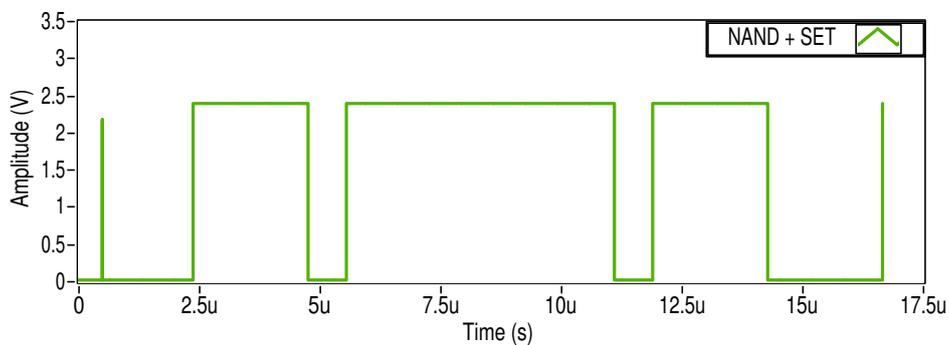
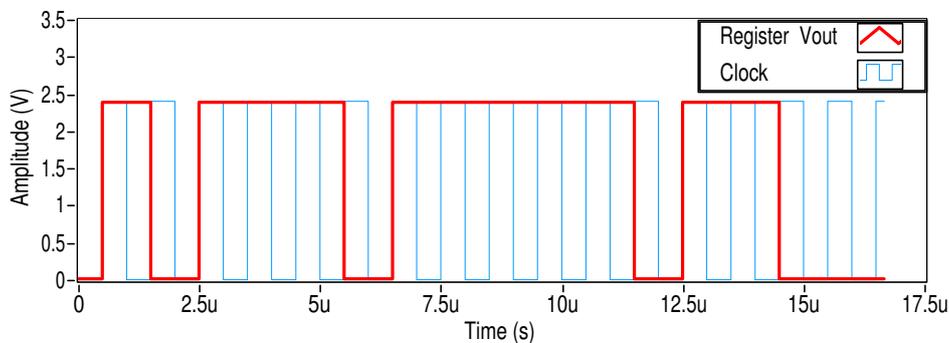


Figura 4.13: BD da Simulação do Circuito Combinacional da Figura 4.12.



(a) Saída da Porta NAND com SET



(b) Saída do Registrador do circuito

Figura 4.14: FP da simulação do Circuito Combinacional da figura 4.12.

entre os três valores e escolherá aquele que se repete mais vezes de entre eles.

O BD da simulação deste circuito é mostrado na figura 4.16. Para representar a aleatoriedade e a mudança do valor dos dados devido a um *Bit-Flip*, é utilizado um gerador de Ruído de Poisson (FAURE; VELAZCO; PERONNARD, 2005) junto a um *buffer* com histerese para evitar que o ruído venha a ser muito agressivo, com isso é possível observar melhor os efeitos do *Bit-Flip*. Pode-se notar que o gerador do *Bit-Flip* afeta a só um dos registradores que estão dentro da estrutura *FOR*.

Este VI de simulação possui um controle para inserir um dado serial e um controle de frequência de relógio f_{CLK} . A saída é um gráfico que mostra o Dado inserido, o sinal

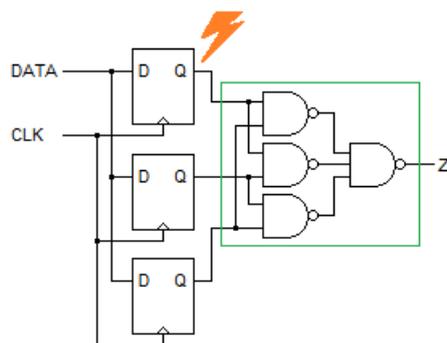


Figura 4.15: Circuito TMR afetado pelo *Bit-Flip*.

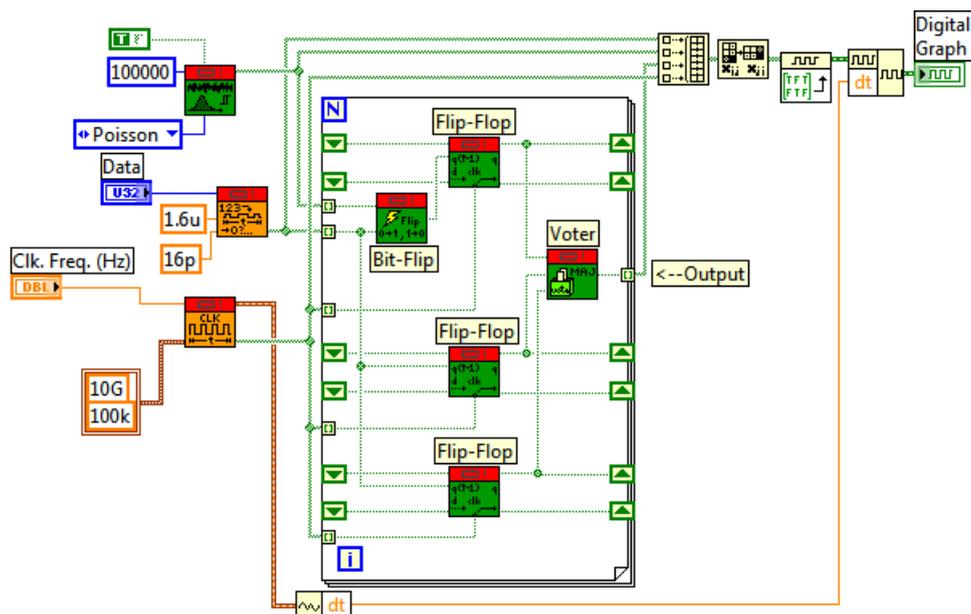


Figura 4.16: BD da Simulação do Circuito da figura 4.15.

que liga o *Bit-Flip*, a saída de dados da célula de memória simulada e o sinal do relógio. Como esta simulação foi feita respeitando as condições do TMR de que o *Bit-Flip* afeta só a uma única célula de memória das três que tem, então a falha é totalmente corrigida tal como se mostra na figura 4.20, sem importar a frequência do relógio.

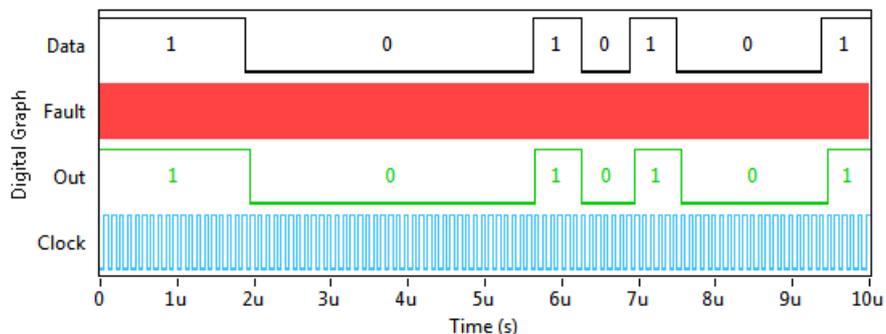


Figura 4.17: FP da simulação do TMR à $f_{CLK} = 10\text{MHz}$.

Agora se realizará a mesma simulação, só que serão duas células de memória afetadas

pelo mesmo *Bit-Flip*. O esquemático e o BD desta simulação são mostrados nas figuras 4.18 e 4.19. Neste caso já não se respeita a base regular de êxito do TMR. Os resultados a $f_{CLK} = 10\text{MHz}$ são mostrados na figura 4.20, onde agora pode-se perceber os erros causados pelo *Bit-Flip*.

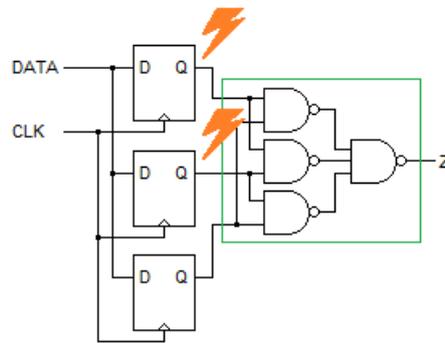


Figura 4.18: Circuito TMR afetado por dois *Bit-Flip*.

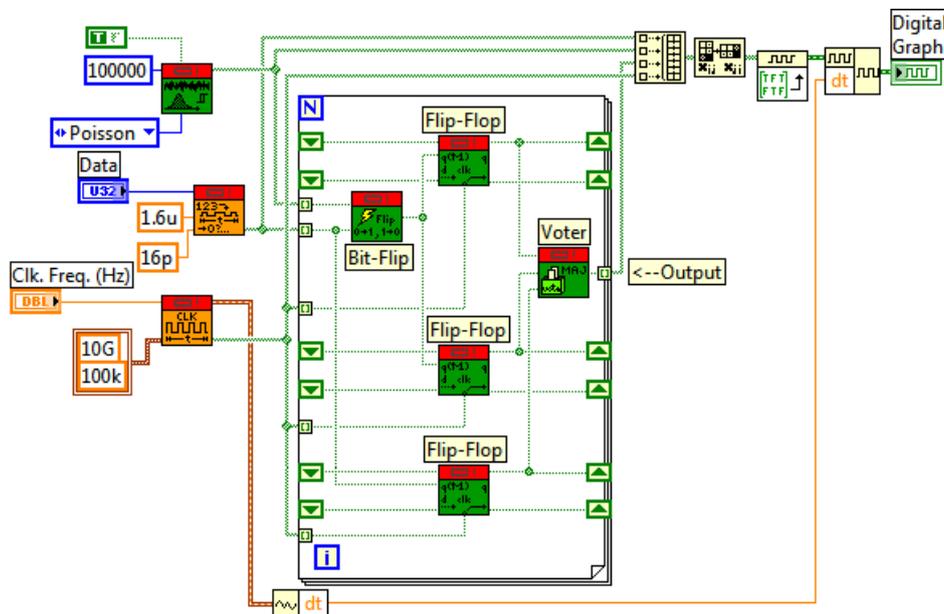


Figura 4.19: BD da Simulação do Circuito da figura 4.18 com Registradores afetados pela mesma fonte de *Bit-Flip*.

Se fez uma varredura de 0 a 275MHz para revisar o acontecido com esta taxa de erros com respeito na frequência. Cada simulação, numa determinada frequência do relógio f_{CLK} , foi rodada 100 vezes para obter dela um numero Meio de Erros. O resultado pode-se ver na figura 4.21. Desta figura se pode observar que a taxa de erros permanece quase constante a 12986 erros, onde o máximo valor de erros meios de 13400 erros a $f_{CLK} = 5\text{MHz}$. Isto significa que a técnica TMR não funciona se mais de um registrador é afetado pelo *Bit-Flip*.

Agora se realizará a mesma simulação do circuito mostrado na figura 4.18, só que neste caso as fontes do *Bit-Flip* e geradores de Ruído de Poisson serão diferentes, mas ambas com o mesmo valor de media $\bar{\mu} = 0.5$ (valor padrão) para esta distribuição. O BD desta simulação é mostrado na figura 4.22. A quantidade de erros diminui como visto

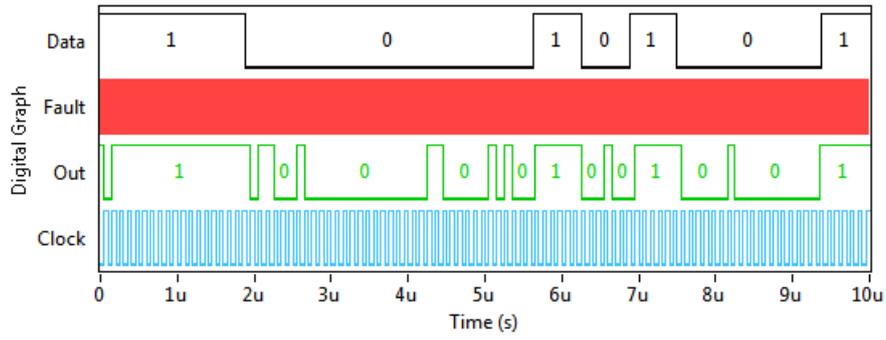


Figura 4.20: FP da simulação do TMR à $f_{CLK} = 10\text{MHz}$ do BD da figura 4.19.

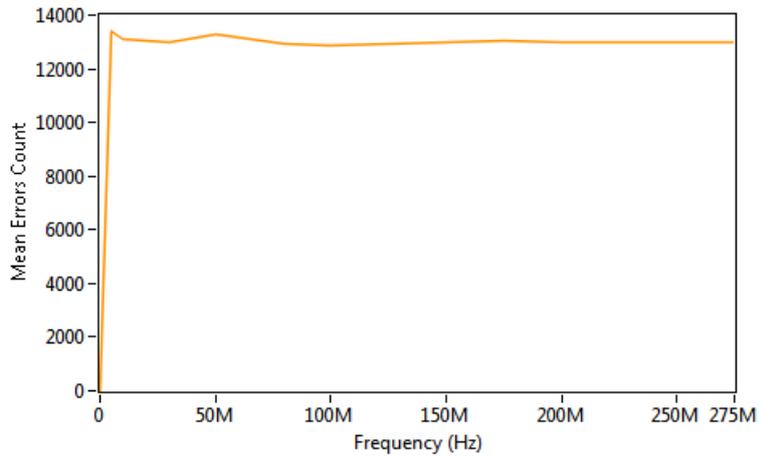


Figura 4.21: Resultados comparativos para diferentes frequências do Relógio.

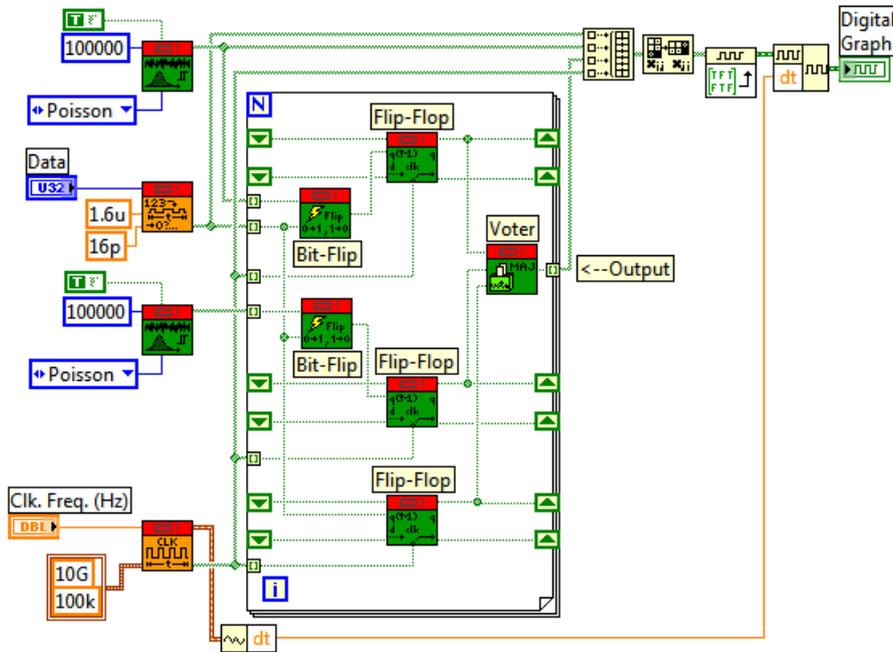


Figura 4.22: BD da Simulação do Circuito da figura 4.18 com dois fontes de *Bit-Flip* diferentes.

na figura 4.20 depois de ver o acontecido quando se realizou esta simulação a $f_{CLK} = 10\text{MHz}$. O FP mostrando estes resultados é mostrado na figura 4.23.

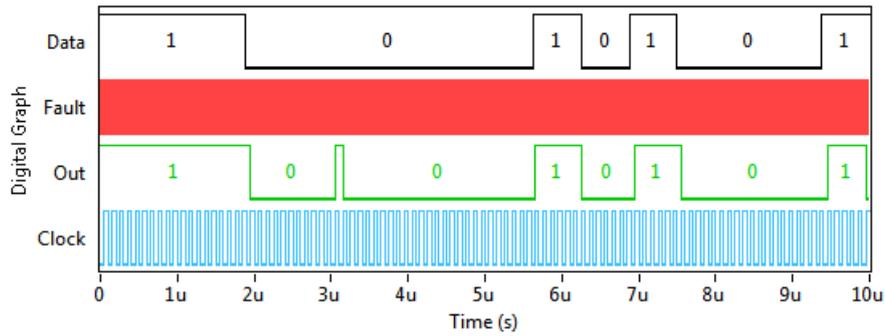


Figura 4.23: FP da simulação do TMR à $f_{CLK} = 10\text{MHz}$ do BD da figura 4.22.

Se fez uma varredura de 0 a 275MHz para revisar o que aconteceu com esta taxa de erros com respeito na frequência. Em cada simulação, a frequência do relógio f_{CLK} foi rodada 100 vezes para obter de ela um número Médio de Erros. O resultado pode-se ver na figura 4.24. Desta figura se pode observar que a taxa de erros permanece quase constante a 1657 erros, onde o máximo valor médio de erros é 1729 erros a $f_{CLK} = 200\text{MHz}$. Se comparamos estes resultados com os resultados da figura 4.21 se tem uma diferença de 87.24% em relação à da simulação anterior. Isto se deve a que alguns *Bit-Flips* de um registrador coincidem com os acontecidos do outro registrador é com os dados certos quando acontece a borda do relógio, além de que as sementes dos geradores aleatórios que geram os *Bit-Flips* são escolhidas pelo computador automaticamente.

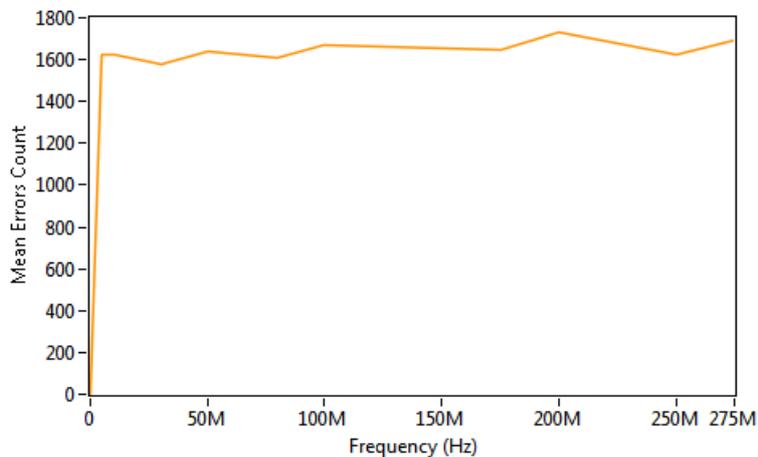


Figura 4.24: Resultados comparativos para diferentes frequências do Relógio.

4.5 Simulação do Bit-Flip em uma Célula de Memória com Redundância Triplicada no Tempo

Na Figura 4.25 é mostrado um circuito que usa a técnica TMR com redundância no tempo para tornar o circuito tolerante a falhas. Foram triplicados os registradores D e se adicionou um votador no final do circuito triplicado. Cada um dos três registradores usa um relógio da mesma frequência, mas cada relógio possui um deslocamento de tempo de uma certa quantidade de graus (este deslocamento é representado por Δt).

O BD da simulação do circuito que representa os *Flip-Flop* mais a estrutura do votador pode ser visto na Figura 4.26. Nesta imagem é possível ver os VIs responsáveis pela produção do Ruído de Poisson, o gerador de Dados e o gerador de relógio triplicado com defasagem na onda. Além disso, pode-se ver o Gerador de *Bit-Flip* que se encontra dentro de uma estrutura *For* junto com os três Registradores D e o *Voter*. Este gerador afeta a todas as entradas dos registradores por igual, simulando a falha nos dados de entrada.

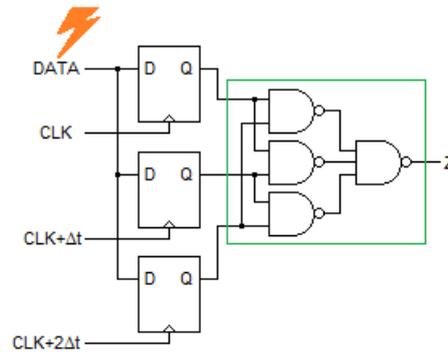


Figura 4.25: Circuito TMR com Redundância Temporal afetado pelo *Bit-Flip*.

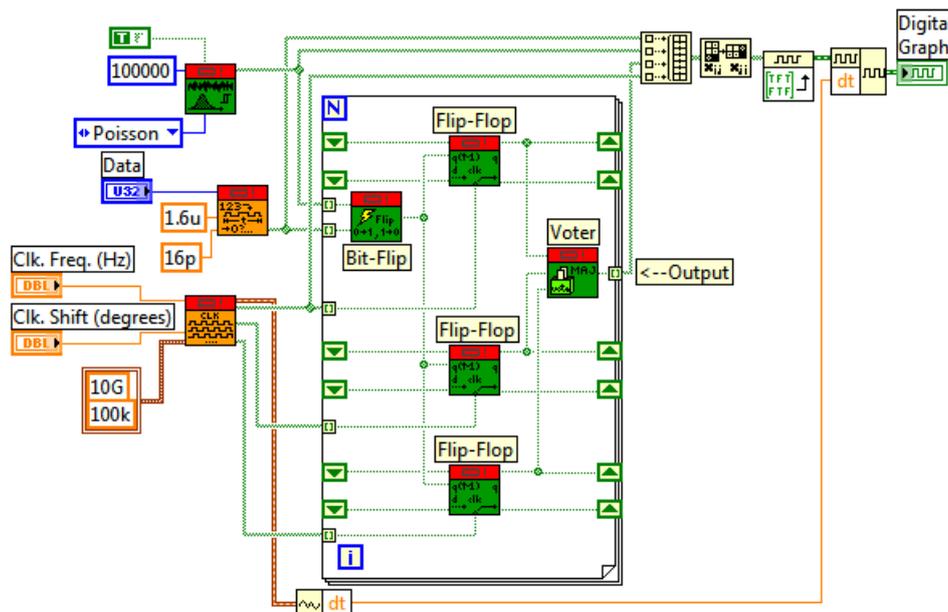


Figura 4.26: BD da Simulação do Circuito da figura 4.25.

Este VI possui um controle para inserir a frequência de relógio (*Clk. Freq. (Hz)*), um controle para colocar os dados (*Data*) e um controle para inserir a defasagem do relógio em graus (*Clock Shift (degrees)*). A saída é um gráfico mostrando quatro sinais: o dado inserido, o sinal que liga as falhas de *Bit-Flip*, a saída do circuito e o sinal do relógio. Este gráfico é mostrado na figura 4.27, onde o sinal de falha (*Fault*) não é visível devido ao uso de uma alta frequência do relógio. Após haver rodado esta simulação com valores diferentes de deslocamento de relógio Δt e frequência, fazendo uma comparação entre este circuito e um circuito sem *Bit-Flip*, pode-se observar uma tendência do número de erros médios vs. a frequência para valores distintos de defasagem do relógio e o deslocamento em graus dos relógios de cada *Flip-Flop*. Em cada simulação a frequência do relógio

f_{CLK} e Δt foi rodada 100 vezes para obter um número Médio de Erros. Os resultados são mostrados na figura 4.28, onde a quantidade mínima de erros médios é de 4433,5 erros à $f_{CLK} = 50\text{MHz}$ em $\Delta t = 45^\circ$ e máxima é de 13142,5 erros à $f_{CLK} = 80\text{MHz}$ em $\Delta t = 0^\circ$. Pode-se concluir que esta técnica perde confiabilidade em altas frequências, isto é devido a que durante os períodos de transição dos relógios é mais provável que dados errados passem através dos três *Flip-Flops*, enganando ao *voter*. Algo muito parecido acontece com a técnica TMR implementada em FPGAs, onde a quantidade de pontos sensíveis é dramaticamente incrementado quando acontece um período de transição do relógio (BERG et al., 2006).

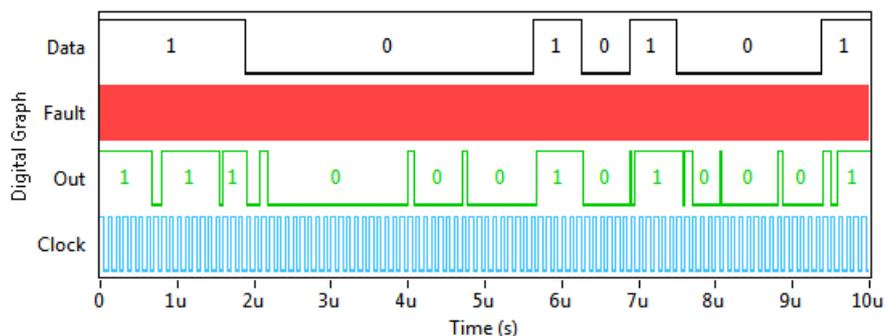


Figura 4.27: FP da simulação do TMR Com Redundância Temporal à $\Delta t = 90^\circ$ e $f_{CLK} = 10\text{MHz}$.

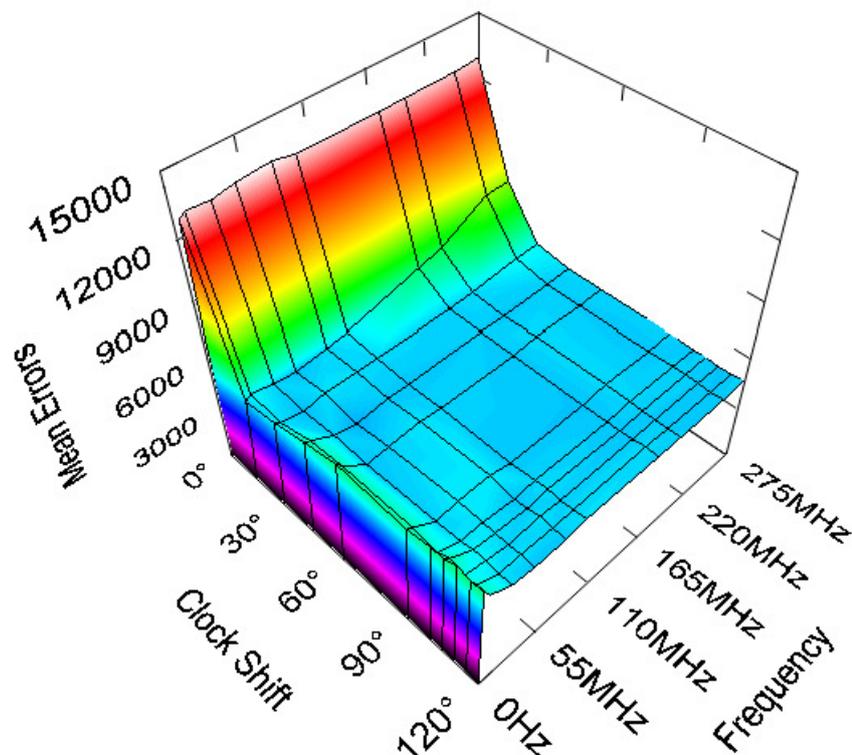


Figura 4.28: Resultados comparativos para diferentes Δt do Relógio em Graus.

Finalmente, pode-se perceber uma semelhança na taxa média de erros quando se tem $\Delta t = 0^\circ$ com aquela mostrada na figura 4.21. Isto faz pensar em que a Técnica TMR com

Redundância no Tempo para $\Delta t = 0^\circ$ é equivalente à Técnica de TMR que tem dois de seus registradores afetados pela mesma fonte de *Bit-Flip*.

4.6 Simulação de um VCO afetado por um Pulso LASER

Os circuitos analógicos também são afetados pelo SET, só que em eles deve-se de inserir um pulso SET de maior duração e carga. Este é o caso da simulação de um circuito VCO de 250nm afetado por um pulso SET com uma carga de 37,125pC como se mostra na figura 4.29. Este pulso foi simulado usando as equações do (2.1) tanto para o SET como para sua carga equivalente (neste caso $I_0 = 25\text{mA}$, $\tau_R = 15\text{ps}$ e $\tau_F = 1,5\text{ns}$). As partículas alfa que impactam sobre os circuitos analógicos contribuem com pouca carga (aproximadamente 50fC) a qual é interpretada como um ruído rosado de curta duração. Para poder perceber os efeitos de falhas nestes circuitos se precisa de simular pulsos de longa duração e de carga grande, perto do 40pC ou mais.

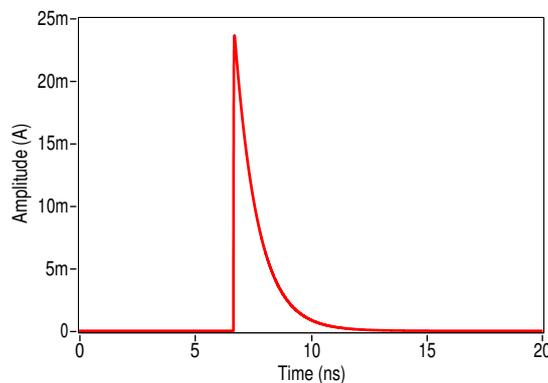


Figura 4.29: Pulso LASER Simulado. Sua carga equivalente é de 37,125pC.

Um VCO é um circuito analógico que gera um sinal senoidal cuja frequência depende do valor da tensão de sintonia (geralmente este valor é menor que o valor da tensão de alimentação do circuito). O circuito é mostrado na figura 4.30. Este circuito se baseia em dois circuitos: um circuito Tanque e um circuito de Par Cruzado do Transistores (formado pelos transistores **M1** e **M2**). Se usa, também, um transistor em configuração de resistor ativo (o transistor **M3**) e um circuito espelho de corrente (formado pelos transistores **M4**, **M5** e **M6**).

Para esta simulação se usou uma indutância de 10mH para cada circuito tanque e um controle de tensão perto dos 1,8V. Isto cria um sinal senoidal de 1,15GHz. Se simulará um pulso assíncrono do LASER usando o pulso SET mostrado na figura 4.29. Este pulso se aplicará no transistor **M2** da figura 4.30 (CHEN et al., 2003). O BD desta simulação é mostrado na figura 4.31. Na figura 4.32 se mostra o resultado desta simulação a qual é comparada com os resultados obtidos por (CHEN et al., 2003), os quais são mostrados na figuras 4.33.

Pode enxergar-se que a simulação feita no LabVIEW da figura 4.32 é muito parecida àquela feita experimentalmente por (CHEN et al., 2003) mostrada na figura 4.33a. Se pode notar que a mudança do valor de t_{ox} nos transistores varia a amplitude do sinal na simulação: uma diminuição de 19,6% do t_{ox} significa uma diminuição do 5% na amplitude da sinal. Também se fez um estudo do espectro de sinal com e sem o pulso LASER inserido para observar a Assinatura de Amortecimento. Mediante simulação, determinou-se que o espalhamento da frequência corresponde a uma irrupção temporal na saída do VCO

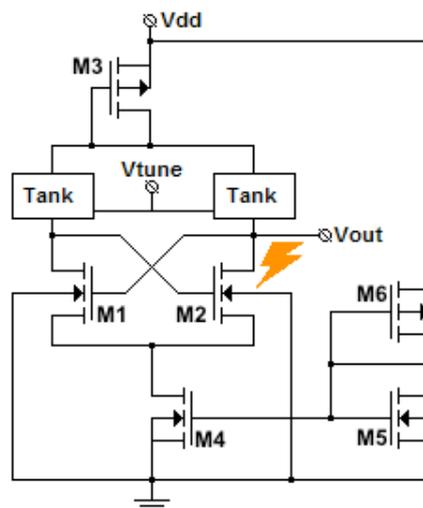


Figura 4.30: Esquemático do Circuito VCO. O raio indica o transistor ao qual se inserira a falha descrita na figura 4.29.

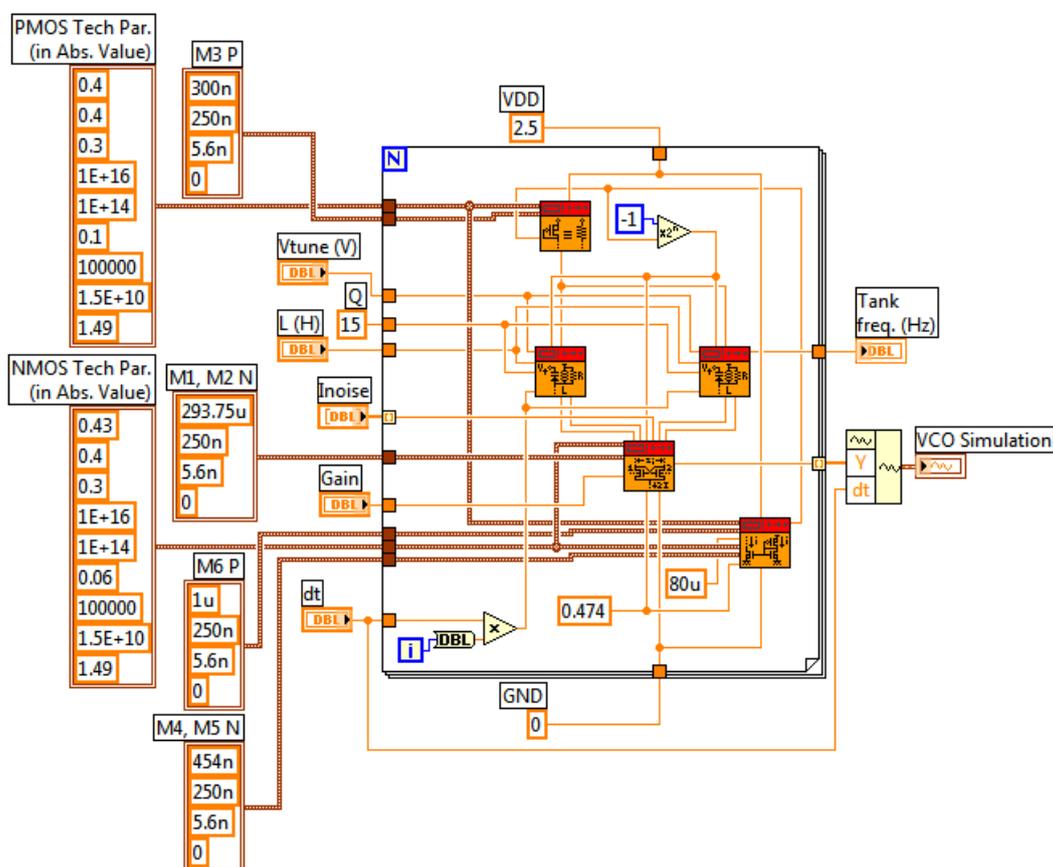


Figura 4.31: Diagrama de Blocos da Simulação de um VCO afeitado por um pulso LASER.

(CHEN et al., 2003). Por outro lado, as simulações indicam que isto correlaciona-se com a quantidade de tempo na qual a saída fica com falha. Na figura 4.34 se mostra o espectro deste VCO simulado antes e depois de inserir-se a falha. O incremento dos harmônicos na saída do VCO faz com que o espectro perto da frequência de oscilação se espalhe. Este fenômeno depende da carga do SET inserida no VCO. A figura 4.35 mostra o mesmo

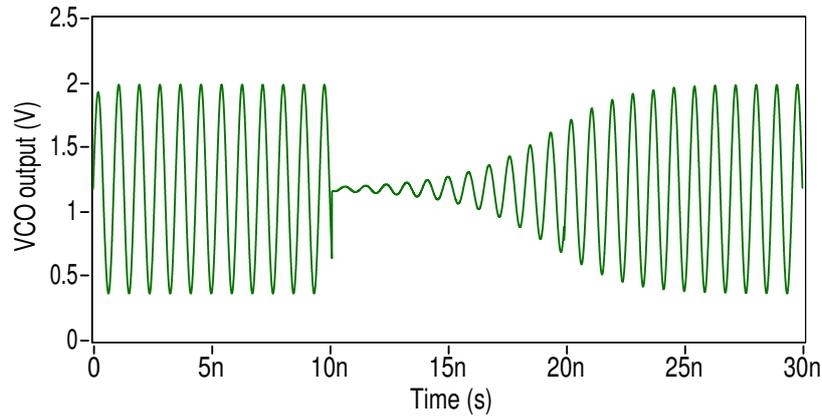


Figura 4.32: Simulação de um VCO afetado por um pulso LASER.

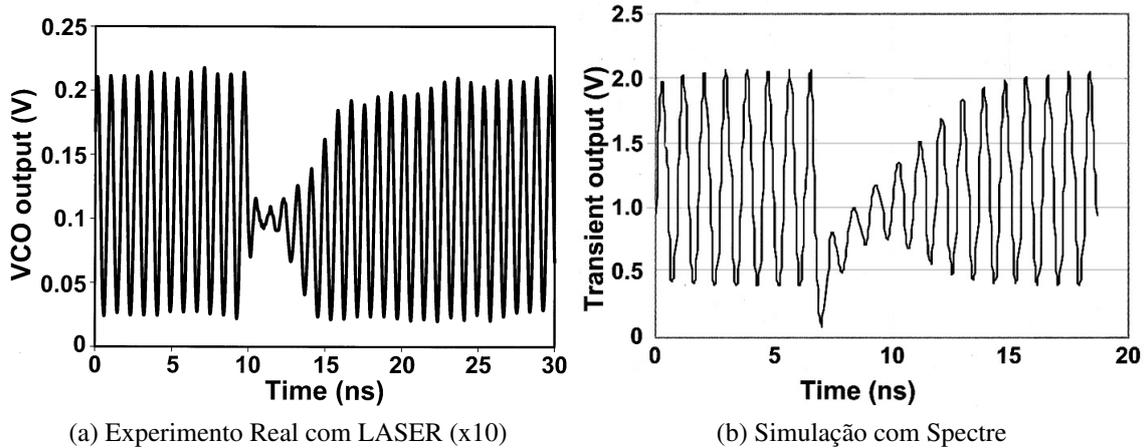


Figura 4.33: Resultados obtidos por (CHEN et al., 2003).

fenômeno, mostrando a influência da carga do pulso transitório sobre o VCO, desde 0 a 100pC. Pode-se ver que a falha no domínio da frequência é significativa a partir dos 90fC de carga inserida pelo SET de $\tau_R = 15\text{ps}$ e $\tau_F = 1,5\text{ns}$. Isto também pode ser enxergado com maior facilidade usando a Distorsão Harmônica Total mais Ruído (THD+N) (HUFFENUS et al., 2010; CHOI et al., 2011) da Assinatura de Amortecimento, que se mostra na figura 4.36, onde 90fC coincide com 10% do THD+N, que é a porcentagem típico de potência de ruído se é comparada com a potência de um sinal sem ruído.

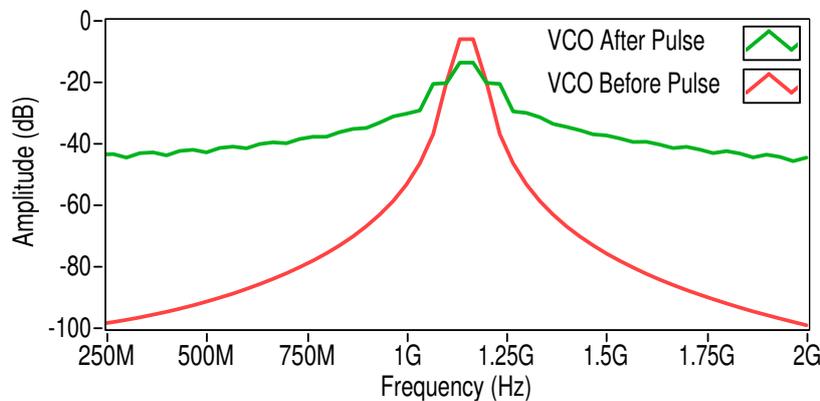


Figura 4.34: Assinatura de Amortecimento do VCO.

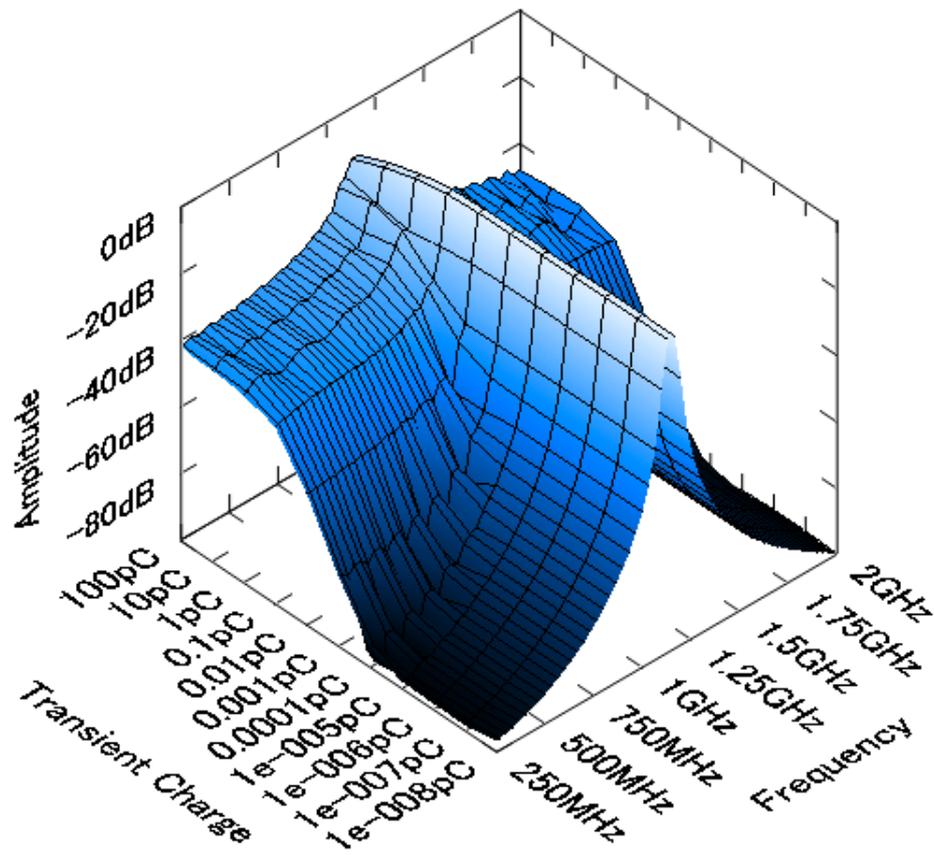


Figura 4.35: Varrimento 3D da Assinatura de Amortecimento do VCO.

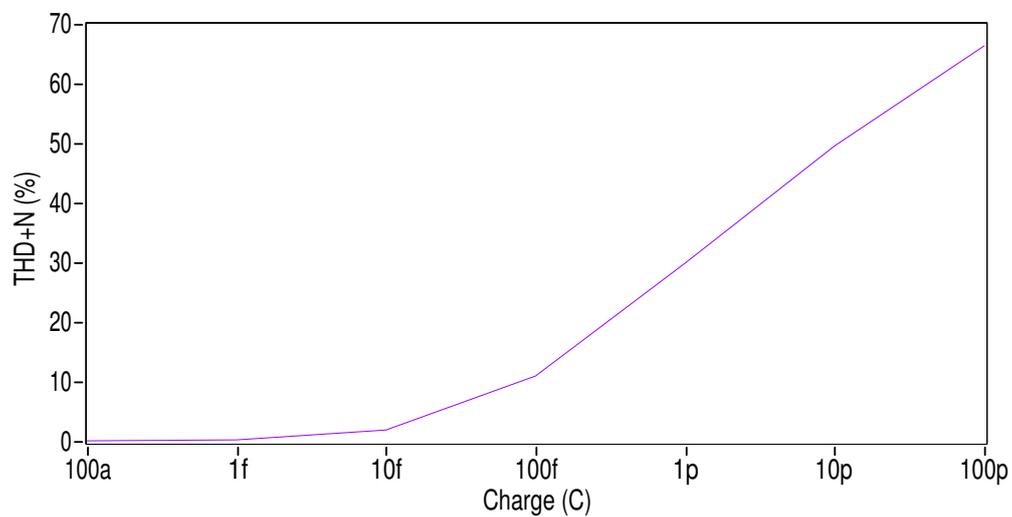


Figura 4.36: THD+N da Assinatura de Amortecimento do VCO.

5 CONCLUSÕES

Foram testados alguns dos métodos existentes de geração e mitigação de falhas em CI com êxito. Os resultados obtidos conferem com o exposto nos trabalhos presentes na literatura.

Os ajustes iniciais, quando se inicia uma simulação, devem ser precisos e iguais para todos os VIs que se usam, respeitando os critérios de tempo de simulação e tamanho de amostra. O conhecimento do Critério de Amostragem de Nyquist é necessário para poder definir a temporização da simulação.

As comparações dos resultados obtidos com o *Toolkit* e os obtidos por SPICE indicam a necessidade da simulação das capacitâncias parasitas das portas lógicas, tanto de entradas como de saída. Fazendo isto se poderia simular os atrasos e a mitigação de tensão devido à característica de “filtro” que tem as capacitâncias.

O uso de geradores de SET aleatórios tanto em localização temporal como em amplitude é uma das melhores formas de modelar o bombardeio de partículas sob um CI. Dependendo das condições ambientais e o tipo de partícula pode-se modelar estas situações de maneira adequada mudando a quantidade de carga de cada pulso transiente. Uma modelagem deste modo é possível ser feito no SPICE, mas isto requer uma análise prévia do tempo dos impactos e a mudança do sinal da fonte de corrente que modela cada um dos pulsos transientes dependendo do nível e que se encontra o sinal de saída.

O uso de uma fonte mais real é necessário para que as simulações fiquem mais perto da realidade, mas a escolha dos tempos tanto de subida como de descida é importante para criar um sinal de relógio real. Com respeito ao *Toolkit* feito, este tipo de sinal causa uma espécie de *jitter* devido ao modelagem do Nível de Ruído (*Band Gap*) das portas lógicas, fazendo ainda mais necessário a modelagem das capacitâncias parasitas de entrada.

Os SETs são um tema crítico nos sistemas que contém várias portas lógicas e registradores quando estes coincidem com a borda de relógio que permite o tráfego da informação da entrada de um *Flip-Flop* para sua saída. Isto significa corrupção da informação transmitida pela rede.

Se posso conferir que a técnica TMR só é válida quando o *Bit-flip* acontece em só um dos registradores do arranjo triplicado. Se acontece em mais de um destes, a técnica é ineficaz. De acordo com as simulações feitas para este trabalho é possível que a taxa de erros diminua se as fontes de *Bit-flip* que afetam a cada registrador do arranjo sejam de origens distintas, mas com as mesmas características de área transversal, isto é devido a que alguns *Bit-Flips* de um registrador não coincidem com os acontecidos do outro registrador e com o dado certo do último registrador, além de que sementes dos geradores aleatórios que geram os *Bit-Flips* são escolhidas pelo computador automaticamente. Também observou-se que as tendências das taxas de erro ficam quase constantes a altas frequências, isto significa uma estabilidade na taxa de erros que pode servir como uma

característica para qualificar o desempenho destas técnicas.

A técnica TMR com Redundância no Tempo tem uma taxa de erros alta quando a frequência e o deslocamento dos relógios aumenta, mas fica quase constante a partir de uma frequência de $f_{CLK} = 50\text{MHz}$ e de um deslocamento de $\Delta t = 30^\circ$, uma tendência observada nas simulações prévias com TMR que serve também como qualificação de desempenho desta técnica. Também se observou o parecido do desempenho desta técnica a $\Delta t = 0^\circ$ com o desempenho do TMR com dois registradores afetados pelo *Bit-flip* de uma mesma fonte de SEU. Isto pode significar que existe uma relação entre estas duas técnicas distintas com estas características.

A Assinatura de Amortecimento é uma característica que pode prever o comportamento de um circuito oscilador afetado pelo ASEE no Domínio da Frequência em função da carga que contém o pulso de corrente produzido. Outra forma melhor de saber quanta carga é necessária para considerar que o ASEE afeta de forma significativa ao VCO é usando o THD+N da Assinatura de Amortecimento, a qual pode mostrar mais claramente os efeitos considerando a corrente transiente como um ruído.

Após ter desenvolvido os VIs para a simulação de geração e tolerância a falhas, se pode demonstrar que o *LabVIEW* pode ser uma boa plataforma de prototipação, programação e simulação. Uma adequada hierarquização das funções e a otimização do cabeado entre elas aceleram o tempo de desenvolvimento das simulações feitas com este *toolkit*. Além disso, os métodos implementados serviram para melhorar o aprendizado destes temas tão importantes para as tecnologias modernas.

5.1 Trabalho Futuro

A modelagem do transistor deve ser mudado para poder simular tecnologias por abaixo dos 250nm. O modelo ACM seria ideal devido aos poucos parâmetros que precisa e a forma de tratar o transistor como um dispositivo simétrico (GALUP-MONTORO et al., 2003; MONTORO; SCHEIDER, 2007). Também é necessário a simulação das capacitâncias parasitas para uma correta modelagem dos atrasos nas Portas Lógicas. Com isto se melhorará a modelagem das falhas de acordo com as tecnologias atuais. Também se contempla a possibilidade de unificar os modelos dos transistores da parte lógica e analógica em um só.

Pretende-se também adicionar mais funções, tais como geradores de código Hamming, simulação de falhas e testes em circuitos de memória, EDAC e funções para análise de falhas em CIs. Existem alguns artigos que seriam interessantes de implementar num futuro, como o TMR baseado em *Scan-Chain* (EBRAHIMI; MIREMADI; ASADI, 2011) ou geração de Comparadores de Autocontrole Total (HUGHES; MCCLUSKEY; LU, 1984) e espalhamento dos pulsos SET através de cadeias de Portas Lógicas, por exemplo. Também se incrementaram as funções para a parte analógica com VIs para simular pares diferenciais, capacitores comutados, filtros analógicos e amplificadores Miller (ROMERO; PERETTI; MARQUÉS, 2001) e cascodo. Para isto deve-se desenvolver um tipo novo de dado com ajuda dos *clusters* do *LabVIEW* e evitar a condição de Código *spagetti* que acontece quando o BD está enchido de cabos, fazendo o seguimento do VI muito difícil (TRAVIS; KRING, 2006; JOHNSON; JENNINGS, 2006).

ANEXO A *NETLISTS* PARA HSPICE

A.1 Biblioteca `lib_250nmEE141.lib` da Berkeley University

```

1
2
3 * *****
4 *   PROCESS : 0.25um EE141      (2.5V) *
5 *   SPICE MODEL   BSIM3 - V3.1   *
6 * *****
7
8 * *****
9 *   THIS MODEL LIB CONTAINS :
10 *
11 *   1. MOSFET
12 *   LIB TT
13 *   SS
14 *   FF
15 *   SF
16 *   FS
17 *
18 * *****
19 *
20 * You can use this model library by including the following library reference
21 * in your SPICE file
22 *
23 *   .lib 'lib_path/lib_name' model_name
24 *
25 * EX: .lib '<ModelFile>' TT
26 *
27 *           for typical N,PMOS
28 *
29 * MAKE SURE TO CHANGE ALL REFERENCES to <MODELFILE> in this file to the actual location
30 * of the file on your computer.
31 *
32 * note:
33 *   corner_name
34 *   TT : typical model
35 *   SS : Slow NMOS Slow PMOS model
36 *   FF : Fast NMOS Fast PMOS model
37 *   SF : Slow NMOS Fast PMOS model
38 *   FS : Fast NMOS Slow PMOS model
39 *
40 * <ModelFile> = lib_250nmEE141.lib
41 * *****
42 *
43 *           2.5V NORMAL DEVICES LIB
44 *
45 * *****
46 * ***** CORNER_LIB OF TYPICAL MODEL *****
47 * LIB TT
48 * .param toxp = 5.8e-9 toxn = 5.8e-9
49 * +dxl = 0 dxw = 0
50 * +dvthn = 0 dvthp = 0
51 * +cjn = 2.024128E-3 cjp = 1.931092e-3
52 * +cjswn = 2.751528E-10 cjswp = 2.232277e-10
53 * +cgon = 3.11E-10 cgop = 2.68e-10
54 * +cjgaten = 2.135064E-10 cjgatep = 1.607088e-10
55 * +hdifn = 3.1e-07 hdifp = 3.1e-7
56 * .lib 'lib_250nmEE141.lib' MOS
57 * .ENDL TT
58 * ***** CORNER_LIB OF SNRP MODEL *****
59 * LIB SS
60 * .param toxp = 6.1e-9 toxn = 6.1e-9
61 * +dxl = 2.5e-8 dxw = -3e-8
62 * +dvthn = 0.06 dvthp = -0.06
63 * +cjn = 2.2265408E-3 cjp = 2.1242e-3
64 * +cjswn = 3.0266808E-10 cjswp = 2.4555e-10
65 * +cgon = 3.421E-10 cgop = 2.948e-10
66 * +cjgaten = 2.3485704E-10 cjgatep = 1.7678e-10
67 * +hdifn = 3.1e-07 hdifp = 3.1e-7
68 * .lib 'lib_250nmEE141.lib' MOS
69 * .ENDL SS
70 * ***** CORNER_LIB OF FNFP MODEL *****
71 * LIB FF

```

```

72 .param toxp = 5.5e-9 toxn = 5.5e-9
73 +dx1 = -2.5e-8 dxw = 3e-8
74 +dvthn = -0.06 dvthp = 0.06
75 +cjn = 1.8217152e-3 cjp = 1.738e-3
76 +cjswn = 2.4763752e-10 cjswp = 2.009e-10
77 +cgon = 2.799e-10 cgop = 2.412e-10
78 +cjpgaten = 1.9215576e-10 cjpgatep = 1.4464e-10
79 +hdifn = 3.1e-7 hdifp = 3.1e-7
80 .lib 'lib_250nmEE141.lib' MOS
81 .ENDL FF
82 ***** CORNER_LIB OF SNFP MODEL *****
83 .LIB SF
84 .param toxp = 5.8e-9 toxn = 5.8e-9
85 +dx1 = 0 dxw = 0
86 +dvthn = 0.06 dvthp = 0.06
87 +cjn = 2.2265408E-3 cjp = 1.738e-3
88 +cjswn = 3.0266808E-10 cjswp = 2.009e-10
89 +cgon = 3.11E-10 cgop = 2.68e-10
90 +cjpgaten = 2.3485704E-10 cjpgatep = 1.4464e-10
91 +hdifp = 3.1e-7 hdifn = 3.1e-07
92 .lib 'lib_250nmEE141.lib' MOS
93 .ENDL SF
94 ***** CORNER_LIB OF FNFP MODEL *****
95 .LIB FS
96 .param toxp = 5.8e-9 toxn = 5.8e-9
97 +dx1 = 0 dxw = 0
98 +dvthn = -0.06 dvthp = -0.06
99 +cjn = 1.8217152e-3 cjp = 2.1242e-3
100 +cjswn = 2.4763752e-10 cjswp = 2.4555e-10
101 +cgon = 3.11E-10 cgop = 2.68e-10
102 +cjpgaten = 1.9215576e-10 cjpgatep = 1.7678e-10
103 +hdifp = 3.1e-7 hdifn = 3.1e-07
104 .lib 'lib_250nmEE141.lib' MOS
105 .ENDL FS
106 *****
107 *
108 .LIB MOS
109
110 *****
111 * NMOS DEVICES MODEL *
112 *****
113
114
115 .MODEL nmos NMOS (
116 +LMIN = 2.4E-07 LMAX = '5.1E-07-dx1'
117 +LEVEL = 49 TNOM = 25 XL = '3E-8 + dx1'
118 +XW = '0 + dxw' VERSION = 3.1 TOX = toxn
119 +CALCACM = 1 SFVTFLAG = 0 VFBFLAG = 1
120 +XJ = 1E-07 NCH = 2.354946E+17 LLN = 1
121 +LWN = 1 WLN = 1 WVN = 1
122 +LINT = 1.76E-08 WINT = 6.75E-09 MOBMOD = 1
123 +BINUNIT = 2 DWG = 0 DWB = 0
124 +VTH0 = '0.4321336+dvthn' LVTH0 = 2.081814E-08 WVTH0 = -5.470342E-11
125 +PVTH0 = -6.721795E-16 K1 = 0.3281252 LK1 = 9.238362E-08
126 +WK1 = 2.878255E-08 PK1 = -2.426481E-14 K2 = 0.0402824
127 +LK2 = -3.208392E-08 WK2 = -1.154091E-08 PK2 = 9.192045E-15
128 +K3 = 0 DVT0 = 0 DVT1 = 0
129 +DVT2 = 0 DVTOW = 0 DVT1W = 0
130 +DVT2W = 0 NLX = 0 WO = 0
131 +K3B = 0 VSAT = 7.586954E+04 LVSAT = 3.094656E-03
132 +WVSAT = -1.747416E-03 PVSAT = 8.820956E-10 UA = 8.924498E-10
133 +LUA = -1.511745E-16 WUA = -3.509821E-17 PUA = -3.08778E-23
134 +UB = 8.928832E-21 LUB = -1.655745E-27 WUB = -2.03282E-27
135 +PUB = 3.4578E-34 UC = -1.364265E-11 LUC = 1.170473E-17
136 +WUC = -1.256705E-18 PUC = -6.249644E-24 RDSW = 447.8871
137 +PRWB = 0 PRWG = 0 WR = 0.99
138 +UO = 0.06005258 LUO = -6.31976E-09 WUO = -8.819531E-09
139 +PUO = 3.57209E-15 AO = -1.468837 LAO = 6.419548E-07
140 +WA0 = 5.512414E-07 PA0 = -9.222928E-14 KETA = -0.04922795
141 +LKETA = 2.360844E-08 WKETA = 1.560385E-08 PKETA = -5.98377E-15
142 +A1 = 0.02659908 LA1 = -6.511454E-09 A2 = 1
143 +AGS = -4.01637 LAGS = 1.090294E-06 WAGS = 1.162021E-06
144 +PAGS = -3.108579E-13 B0 = 0 B1 = 0
145 +VOFF = -0.1829426 LVOFF = 9.941631E-09 WVOFF = 1.568082E-08
146 +PVOFF = -2.832958E-15 NFACTOR = 0.6790636 LNFACTOR = 3.454948E-08
147 +WNFACTOR = 1.501016E-07 PNFACTOR = -2.955591E-14 CIT = 2.218499E-04
148 +LCIT = -1.076934E-10 WCIT = -3.286884E-10 PCIT = 1.658928E-16
149 +CDSC = 0 CDSCB = 0 CDSCD = 0
150 +ETA0 = 1.215578E-04 LETA0 = -1.037758E-11 WETA0 = -3.030225E-11
151 +PETA0 = 1.529658E-17 ETAB = 3.548681E-03 LETAB = -1.791374E-09
152 +WETAB = -6.897268E-10 PETAB = 3.481742E-16 DSUB = 0
153 +PCLM = 3.583838 LPCLM = -6.874146E-07 WPCLM = 5.664574E-08
154 +PPCLM = -1.33176E-15 PDIBLC1 = 0 PDIBLC2 = 5.379674E-03
155 +LPDIBLC2 = 7.808481E-09 WPDIBLC2 = 5.516945E-10 PPDIBLC2 = -2.784957E-16
156 +PDIBLCB = -0.1229374 LPDIBLCB = 4.956215E-08 WPDIBLCB = 3.299946E-08
157 +PPDIBLCB = -9.624918E-15 DROUT = 0 PSCBE1 = 4.472639E+08
158 +LPSCBE1 = 28.64041 WPSCBE1 = 15.7154 PPSCBE1 = -7.933138E-06
159 +PSCBE2 = 1.842585E-06 LPSCBE2 = 2.871008E-12 WPSCBE2 = 2.579183E-12
160 +PPSCBE2 = -1.301972E-18 PVAG = -2.015254E-03 LPVAG = 1.017757E-09
161 +WPVAG = 3.07622E-10 PPVAG = -1.55418E-16 DELTA = -0.02862256
162 +LDELTA = 1.492454E-08 WDELTA = -6.71663E-09 PDELTA = 3.407521E-15
163 +ALPHA0 = 0 BETA0 = 30 KT1 = -0.2579945
164 +LKT1 = -1.664895E-08 WKT1 = -1.633463E-08 PKT1 = 3.755864E-15
165 +KT2 = -0.05347481 LKT2 = 8.244731E-09 WKT2 = 1.13705E-09
166 +PKT2 = -1.240924E-15 AT = -1.132632E+04 LAT = 6.469047E-03
167 +WAT = 6.829220E-04 PAT = -4.154249E-10 UTE = -2.309089
168 +LUTE = 1.662427E-07 WUTE = 1.244801E-07 PUTE = -5.627924E-14
169 +UA1 = -3.461758E-10 LUA1 = 1.747495E-16 WUA1 = -1.42065E-16
170 +PUA1 = 7.171442E-23 UB1 = 0 UC1 = -2.38157E-12

```

```

171 +LUC1 = -2.895726E-18 WUC1 = -1.990052E-17 PUC1 = 1.004497E-23
172 +KT1L = 0 PRT = -1E-18 CJ = cjn
173 +MJ = 0.4960069 PB = 0.9173808 CJSW = cjsw
174 +MJSW = 0.443145 PBSW = 0.9173808 CJSWG = cjsgaten
175 +MJSWG = 0.443145 PBSWG = 0.9173808 HDIF = hdifn
176 +RS = 0 RD = 0
177 +ACM = 12 LDIF = 1.2E-07 RSH = 4.5
178 +CTA = 7.707813E-04 CTP = 5.512283E-04 PTA = 1.167715E-03
179 +PTP = 1.167715E-03 N = 1 XTI = 3
180 +CGDO = 'cgon' CGSO = 'cgon' CAPMOD = 0
181 +NQSMOD = 0 XPART = 1 CF = 0
182 +TLEV = 1 TLEVC = 1 JS = 1E-06
183 +JSW = 5E-11 )
184 *
185 *****
186 * PMOS DEVICES MODEL *
187 *****
188
189 .MODEL pmos PMOS ( LEVEL = 49
190 +VERSION = 3.1 LMIN = 2.4E-7 LMAX = '5.0E-7-dx1'
191 +XL = '3e-8+dx1'
192 +XW = '0+dxw' TNOM = 25 TOX = toxp
193 +CALCACM = 1 SFVTFLAG= 0 VFBFLAG = 1
194 +XJ = 1E-7 NCH = 4.1589E17
195 +LLN = 1 LWN = 1 WLN = 1
196 +WWN = 1 LINT = 1.2365E-8 WINT = 7.8E-9
197 +MOBMOD = 1 BINUNIT = 2 DWG = 0
198 +DWB = 0 VTH0 = 'dvtthp-0.6236538' LVTH0 = 2.649834E-8
199 +WVTH0 = 3.214189E-8 PVTH0 = -3.22268E-15 K1 = 0.4198155
200 +LK1 = 5.770498E-8 WK1 = 5.577151E-8 PK1 = -2.81684E-14
201 +K2 = 0.0429467 LK2 = -2.296405E-8 WK2 = -1.355302E-8
202 +PK2 = 6.848271E-15 K3 = 0 DVT0 = 0
203 +DVT1 = 0 DVT2 = 0 DVTOW = 0
204 +DVT1W = 0 DVT2W = 0 NLX = 0
205 +W0 = 0 K3B = 0 VSAT = 1.443912E5
206 +LVSAT = -7.688012E-4 WVSAT = -6.083648E-3 PVSAT = 2.186471E-10
207 +UA = 1.846811E-9 LUA = -3.27694E-16 WUA = -2.82106E-16
208 +PUA = 7.180233E-23 UB = -7.84535E-19 LUB = 4.772849E-25
209 +WUB = 2.599205E-25 PUB = -1.46530E-31 UC = -1.75560E-10
210 +LUC = 3.360832E-17 WUC = 1.504425E-17 PUC = -1.30556E-23
211 +RDSW = 1.03E3 PRWB = 0 PRWG = 0
212 +WR = 1 U0 = 0.0136443 LU0 = -7.22084E-10
213 +WU0 = -1.088554E-9 PU0 = 2.730854E-16 A0 = 0.1071803
214 +LA0 = 4.64252E-7 WA0 = 5.383179E-7 PA0 = -1.32033E-13
215 +KETA = -4.943762E-3 LKETA = -3.565304E-9 WKETA = -5.226247E-9
216 +PKETA = 2.640665E-15 A1 = 0 A2 = 0.4
217 +AGS = 0.1664005 LAGS = 1.19106E-7 WAGS = 5.29237E-8
218 +PAGS = -2.67304E-14 B0 = 0 B1 = 0
219 +VVOFF = -0.0592623 LVVOFF = -1.96686E-8 WVVOFF = -1.486398E-8
220 +PVVOFF = 7.510321E-15 NFACTOR = 0.8588103 LNFACTOR = -1.158881E-7
221 +WNFACTOR = 1.210664E-8 PNFACTOR = -6.11712E-15 CIT = 6.439495E-5
222 +LCIT = 2.916437E-10 WCIT = -3.11284E-11 PCIT = 1.572825E-17
223 +CDSC = 0 CDSB = 0 CDSCD = 0
224 +ETA0 = -3.819468E-3 LETA0 = 2.155422E-9 WETA0 = 8.235612E-10
225 +PETA0 = -4.16037E-16 ETAB = 1.334637E-3 LETAB = -7.93631E-10
226 +WETAB = 5.284657E-11 PETAB = -2.68353E-17 DSUB = 0
227 +PCLM = 0.1098002 LPCLM = 6.874263E-7 WPCLM = 6.724724E-7
228 +PPCLM = -1.97766E-13 PDIBLC1 = 0 PDIBLC2 = 5.801323E-3
229 +LPDIBLC2 = -1.81964E-9 WPDIBLC2 = -5.853396E-9 PPDIBLC2 = 2.957545E-15
230 +PDIBLCB = 0.1921199 DROUT = 0 PSCBE1 = 7.19E8
231 +PSCBE2 = 1E-20 PVAG = 0 DELTA = 0.01
232 +ALPHA0 = 0 BETA0 = 30 KT1 = -0.3248987
233 +LKT1 = -1.160393E-8 WKT1 = 4.153356E-8 PKT1 = -4.62347E-15
234 +KT2 = -0.0367632 AT = 1E4 UTE = -1.04
235 +UA1 = 3.992421E-10 UB1 = -9.23294E-19 LUB1 = -5.28718E-26
236 +WUB1 = -6.13069E-26 PUB1 = 1.503674E-32 UCL = -1.00699E-12
237 +KT1L = 0 PRT = 0 CJ = cjp
238 +MJ = 0.4812153 PB = 0.9134669 CJSW = cjswp
239 +MJSW = 0.3237595 PBSW = 0.9134669 CJSWG = cjsgatep
240 +MJSWG = 0.3237595 PBSWG = 0.9134669 HDIF = hdifp
241 +LDIF = 1.2E-7 ACM = 12 RS = 0
242 +RD = 0 RSH = 3.5 CTA = 8.3043E-4
243 +CTP = 4.30175E-4 PTA = 1.3004E-3 PTP = 1.3004E-3
244 +CGDO = cgop CGSO = cgop
245 +CAPMOD = 0 NQSMOD = 0 XPART = 1
246 +CF = 0 N = 1 XTI = 3
247 +TLEV = 1 TLEVC = 1 JS = 3E-7
248 +JSW = 5E-12 )
249 *
250 .ENDL MOS

```

A.2 Netlist Test .sp para a simulação de uma porta Inversora com SET no HSPICE

```
1
2 Inversor com SET
3
4 *****
5 *****Library*****
6 *****
7
8 .LIB 'lib_250nmEE141.lib' TT
9
10 *****
11 *****Sources*****
12 *****
13
14 VDDA VDD 0 DC 2.4V
15 VIN IN 0 AC PULSE(0 2.4V 0 10n 10n 490n 1u)
16 IP1 VDD OUT AC EXP(0 350u 3.2165231u 10p 3.2165231u 100p)
17
18 *****
19 *****Circuit*****
20 *****
21
22 M1 OUT IN VDD VDD PMOS L=250n W=900n
23 M2 OUT IN 0 0 NMOS L=250n W=300n
24 COUT OUT 0 200f
25
26 *****
27 *****Control*****
28 *****
29
30 .TRAN 1n 3.75u 2.9u 16.6667p
31
32 .END
```

ANEXO B O CRITERIO DE BARKHAUSEN

B.1 Critério de Oscilação

A figura B.1 mostra um sistema de realimentação positiva, o qual modela um oscilador qualquer. Nesta figura, x_i e x_o são os sinais de entrada e saída respectivamente, enquanto que x_r é o sinal de realimentação e x_e é o sinal de erro.

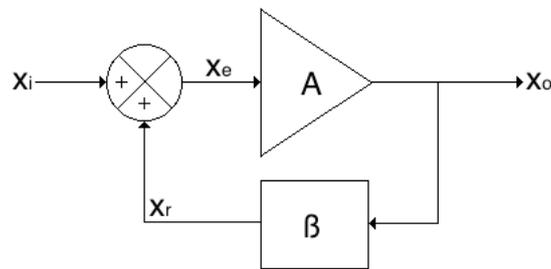


Figura B.1: Diagrama de Blocos de um circuito com Realimentação Positiva.

O amplificador tem um ganho inicial de $A = A(\omega)$ (conhecida como ganho de laço aberto), o $\beta = \beta(\omega)$ é o fator de realimentação e $A\beta = A(\omega)\beta(\omega)$ é o ganho do laço. Todos estes valores são números complexos que dependem da frequência angular ω . Então, o ganho do circuito é:

$$\frac{x_o}{x_i} = \frac{A}{1 - A\beta} \quad (\text{B.1})$$

É possível representar o valor $A\beta$ da seguinte forma:

$$A\beta = |A\beta|e^{j\phi_{A\beta}} \quad (\text{B.2})$$

onde $j = \sqrt{-1}$, $|A\beta|$ é o módulo e $\phi_{A\beta}$ é a fase do número complexo $A\beta$.

O comportamento deste circuito pode-se predecir a partir dos valores $|A\beta|$ e $\phi_{A\beta}$:

1. Se $|A\beta| < 1$, o circuito é estável sem importar o valor de $\phi_{A\beta}$.
2. Se $|A\beta| = 1$ e $\phi_{A\beta} = 0$ ($A\beta = 1$) para um valor de ω determinado, qualquer oscilação na entrada na mesma frequência $\phi_{A\beta}$ é mantida na mesma amplitude.
3. Se $|A\beta| > 1$ e $\phi_{A\beta} = 0$ ($A\beta > 1$) para um valor de ω determinado, qualquer oscilação na entrada do circuito é amplificada de forma indefinida até que a saturação do amplificador volte o circuito até a sua condição inicial.

Se o circuito tem $A\beta > 1$ podemos não usar o sinal de entrada devido a que o ruído, sempre presente, contem componentes de todas as frequências (ILLES BALEARS, 2007). A componente de ruído de frequência que cumpra esta condição (conhecido como **condição de arranque**) vai-se amplificar até a saturação do transistor ou até que um circuito auxiliar consiga que $A\beta = 1$ para esta frequência. Então a amplitude da oscilação é constante. Estas condições para que um circuito oscile são conhecidas como o “Critério de Barkhausen”.

B.2 Análise das Condições de Oscilação

O método de análise consiste primeiro em identificar o laço de realimentação e o sentido do laço (ILLES BALEARS, 2007). Depois o laço deve abrir-se num ponto qualquer, botar no início do laço aberto uma tensão auxiliar v_x , e no final do laço aberto uma impedância Z_{in} igual na impedância de entrada que se enxerga desde o início do laço aberto, tal como se mostra na figura B.2.

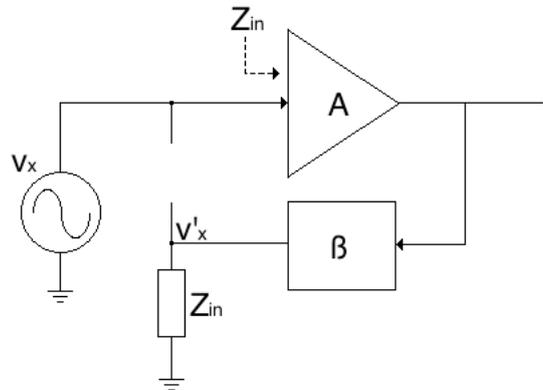


Figura B.2: Quebra do laço de realimentação para calcular o ganho do laço.

Se deve então calcular o sinal que chega até o final do laço v'_x . Então, o ganho do laço aberto é:

$$\frac{v'_x}{v_x} = A\beta \quad (\text{B.3})$$

Se aplica então o Critério de Barkhausen: $\phi_{A\beta} = 0$ e $|A\beta| > 1$ para obter a frequência de oscilação e a condição de arranque. O Critério de Barkhausen também pode ser aplicado se $A\beta > 1$, isto é $\Im(A\beta) = 0$ e $\Re(A\beta) > 1$.

O ganho de laço, $A\beta$, é independente do ponto onde o laço é quebrado, mas a dificuldade do cálculo não. Escolher um ponto ótimo no qual $Z_{in} = \infty$ pode simplificar muito os cálculos. Além disso, pode-se escolher um ponto onde a impedância de saída seja nula, e assim o valor de Z_{in} não importe.

REFERÊNCIAS

ADELL, P. et al. Analysis of single-event transients in analog circuits. **Nuclear Science, IEEE Transactions on**, [S.l.], v.47, n.6, p.2616 – 2623, dec 2000.

ALEXANDER, D. Transient ionizing radiation effects in devices and circuits. **Nuclear Science, IEEE Transactions on**, [S.l.], v.50, n.3, p.565 – 582, june 2003.

ANGHEL, L. et al. Multi-level Fault Effects Evaluation. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.69 – 88.

ARTOLA, L. et al. Collected Charge Analysis for a New Transient Model by TCAD Simulation in 90 nm Technology. **Nuclear Science, IEEE Transactions on**, [S.l.], v.57, n.4, p.1869 – 1875, aug. 2010.

AZEREDO-LEME, C. Clock Jitter Effects on Sampling: a tutorial. **Circuits and Systems, IEEE**, [S.l.], v.11, n.3, p.26 – 37, set. 2011.

BALASUBRAMANIAN, A. et al. Random Dopant Effect on Vt Variations Affecting the Soft-Error Rates of Nanoscale CMOS Memory Cells. In: RELIABILITY PHYSICS SYMPOSIUM, 2007. PROCEEDINGS. 45TH ANNUAL. IEEE INTERNATIONAL. **Anais...** [S.l.: s.n.], 2007. p.318 – 323.

BERG, M. et al. An Analysis of Single Event Upset Dependencies on High Frequency and Architectural Implementations within Actel RTAX-S Family Field Programmable Gate Arrays. **Nuclear Science, IEEE Transactions on**, [S.l.], v.53, n.6, p.3569 – 3574, dec 2006.

BOSIO, A.; NATALE, G. D. LIFTING: a flexible open-source fault simulator. In: ASIAN TEST SYMPOSIUM, 17. **Anais...** [S.l.: s.n.], 2008. p.35 – 40.

BOUDENOT, J.-C. Radiation Space Environment. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.1 – 9.

BUTT, N.; ALAM, M. Modeling single event upsets in Floating Gate memory cells. In: RELIABILITY PHYSICS SYMPOSIUM, 2008. IRPS 2008. IEEE INTERNATIONAL. **Anais...** [S.l.: s.n.], 2008. p.547 – 555.

CAVERLY, R. **CMOS RFIC Design Principles**. Norwood, Massachusetts 02062: Artech House, Inc., 2007.

CAVROIS, V. et al. Investigation of the Propagation Induced Pulse Broadening (PIPB) Effect on Single Event Transients in SOI and Bulk Inverter Chains. **Nuclear Science, IEEE Transactions on**, [S.l.], v.55, n.6, p.2842 – 2853, dec. 2008.

CHA, H. et al. A gate-level simulation environment for alpha-particle-induced transient faults. **Computers, IEEE Transactions on**, [S.l.], v.45, n.11, p.1248 – 1256, nov 1996.

CHA, H.; PATEL, J. A logic-level model for α -particle hits in CMOS circuits. In: COMPUTER DESIGN: VLSI IN COMPUTERS AND PROCESSORS, 1993. ICCD '93. PROCEEDINGS., 1993 IEEE INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 1993. p.538 – 542.

CHEN, C.-Y.; WU, C.-W. An adaptive code rate EDAC scheme for random access memory. In: DESIGN, AUTOMATION TEST IN EUROPE CONFERENCE EXHIBITION (DATE), 2010. **Anais...** [S.l.: s.n.], 2010. p.735 – 740.

CHEN, W. et al. Investigation of single-event transients in voltage-controlled oscillators. **Nuclear Science, IEEE Transactions on**, [S.l.], v.50, n.6, p.2081 – 2087, dec. 2003.

CHEN, W. et al. Radiation Hardened by Design RF Circuits Implemented in $0.13\mu\text{m}$ CMOS Technology. **Nuclear Science, IEEE Transactions on**, [S.l.], v.53, n.6, p.3449 – 3454, dec. 2006.

CHENG, K.-L. et al. RAMSES-FT: a fault simulator for flash memory testing and diagnostics. In: TWENTH IEEE VLSI TEST SYMPOSIUM. **Anais...** [S.l.: s.n.], 2002. p.281 – 286.

CHOI, S. et al. Experimental assessment of logic circuit performance variability with regular fabrics at 90nm technology node. In: SOLID-STATE CIRCUITS CONFERENCE, 2008. ESSCIRC 2008. 34TH EUROPEAN. **Anais...** [S.l.: s.n.], 2008. p.50 – 53.

CHOI, Y. et al. **A 0.018% THD+N, 88-dB PSRR PWM Class-D Amplifier for Direct Battery Hookup**. 2011. 1p. v.PP, n.99.

COLLADANT, T. et al. Study of transient current induced by heavy-ion in NMOS/SOI transistors. **Nuclear Science, IEEE Transactions on**, [S.l.], v.49, n.6, p.2957 – 2964, dec 2002.

CORREIA, V. P.; LUBASZEWSKI, M.; REIS, A. I. SIFU! - A Didactic Stuck-at Simulator. In: IEEE INTERNATIONAL CONFERENCE ON MICROELECTRONIC SYSTEMS EDUCATION. **Anais...** [S.l.: s.n.], 2003. p.93 – 94.

DAHLGREN, P.; LIDEN, P. A switch-level algorithm for simulation of transients in combinational logic. In: FAULT-TOLERANT COMPUTING, 1995. FTCS-25. DIGEST OF PAPERS., TWENTY-FIFTH INTERNATIONAL SYMPOSIUM ON. **Anais...** [S.l.: s.n.], 1995. p.207 – 216.

DODD, P. et al. Production and propagation of single-event transients in high-speed digital logic ICs. **Nuclear Science, IEEE Transactions on**, [S.l.], v.51, n.6, p.3278 – 3284, dec. 2004.

DUZELLIER, L.; BERGER, G. Test Facilities for SEE and Dose Testing. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.201 – 232.

EBRAHIMI, M.; MIREMADI, S.; ASADI, H. ScTMR: a scan chain-based error recovery technique for tmr systems in safety-critical applications. In: DESIGN, AUTOMATION TEST IN EUROPE CONFERENCE EXHIBITION (DATE), 2011. **Anais...** [S.l.: s.n.], 2011. p.1 – 4.

ECOFFET, R. In-flight Anomalies on Electronic Devices. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.31 – 68.

ENTRENA, L. et al. SET Emulation Considering Electrical Masking Effects. **Nuclear Science, IEEE Transactions on**, [S.l.], v.56, n.4, p.2021 – 2025, aug. 2009.

FACCIO, F. Design Hardening Methodologies for ASICs. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.143 – 160.

FAURE, F.; VELAZCO, R.; PERONNARD, P. Single-event-upset-like fault injection: a comprehensive framework. **Nuclear Science, IEEE Transactions on**, [S.l.], v.52, n.6, p.2205 – 2209, dec. 2005.

FOUCARD, G.; PERONNARD, P.; VELAZCO, R. Reliability limits of TMR implemented in a SRAM-based FPGA: heavy ion measures vs. fault injection predictions. In: TEST WORKSHOP (LATW), 2010 11TH LATIN AMERICAN. **Anais...** [S.l.: s.n.], 2010. p.1 – 5.

FOUILLAT, P. et al. Fundamentals of the Pulsed Laser Technique for Single-event Upset Testing. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.121 – 141.

GALLOWAY, K. F.; GAITAN, M.; RUSSELL, T. J. A Simple Model for Separating Interface and Oxide Charge Effects in MOS Device Characteristics. **Nuclear Science, IEEE Transactions on**, [S.l.], v.31, n.6, p.1497 – 1501, dec. 1984.

GALUP-MONTORO, C. et al. Theory, development, and applications of the Advanced Compact MOSFET (ACM) model. **Nanotech**, [S.l.], v.2, 2003.

GEREZ, S. H. **Algorithms for VLSI Design Automation**. West Sussex, Great Britain: John Wiley & Sons Limited, 1999.

GONZÁLES, I.; BERROJO, L. Supporting Fault tolerance in a industrial environment: the amatista approach. In: SEVENTH INTERNATIONAL ON-LINE TESTING WORKSHOP. **Anais...** [S.l.: s.n.], 2001. p.178 – 183.

GUSMÃO KASTENSMIDT, F. L. de; CARRO, L.; REIS, R. **Fault-Tolerance Techniques for SRAM-based FPGAS**. Dordratch, the Netherlands: Springer, 2006.

HO, C.-W.; RUEHLI, A.; BRENNAN, P. The modified nodal approach to network analysis. **Circuits and Systems, IEEE Transactions on**, [S.l.], v.22, n.6, p.504 – 509, jun 1975.

HSIEH, C. M.; MURLEY, P. C.; O'BRIEN, R. R. Dynamics of Charge Collection from Alpha-Particle Tracks in Integrated Circuits. In: RELIABILITY PHYSICS SYMPOSIUM, 1981. 19TH ANNUAL. **Anais...** [S.l.: s.n.], 1981. p.38 – 42.

HSIEH, C.; MURLEY, P.; O'BRIEN, R. A field-funneling effect on the collection of alpha-particle-generated carriers in silicon devices. **Electron Device Letters, IEEE**, [S.l.], v.2, n.4, p.103 – 105, april 1981.

HSING, Y.-T.; WU, S.-G.; WU, C.-W. RAMSES-D: dram fault simulator supporting weighted coupling fault. In: IEEE INTERNATIONAL WORKSHOP ON MEMORY TECHNOLOGY, DESIGN AND TESTING. **Anais...** [S.l.: s.n.], 2007. p.33 – 38.

HUFFENUS, A. et al. A phase-shift self-oscillating stereo class-D amplifier for battery-powered applications. In: CIRCUITS AND SYSTEMS (ISCAS), PROCEEDINGS OF 2010 IEEE INTERNATIONAL SYMPOSIUM ON. **Anais...** [S.l.: s.n.], 2010. p.769 – 772.

HUGHES, J.; MCCLUSKEY, E.; LU, D. Design of Totally Self-Checking Comparators with an Arbitrary Number of Inputs. **Computers, IEEE Transactions on**, [S.l.], v.C-33, n.6, p.546 – 550, june 1984.

ILLES BALEARS, U. de les. **Osciladores**. Disponivel em: <http://www.uib.es/depart/dfs/GTE/education/telematica/sis_ele_comunicacio/Apuntes/Capitulo%206.pdf>.

INSTRUMENTS, N. **LabVIEW Analysis Concepts**. Texas, USA: National Instruments Corporation, 2004.

INSTRUMENTS, N. **NI Vision IMAQ Vision Concepts Manual**. Texas, USA: National Instruments Corporation, 2005.

JEITLER, M.; DELVAI, M.; REICHÖR, S. FuSE - a hardware accelerated HDL fault injection tool. In: PROGRAMMABLE LOGIC, 2009. SPL. 5TH SOUTHERN CONFERENCE ON. **Anais...** [S.l.: s.n.], 2009. p.89 – 94.

JOHNSON, G. W.; JENNINGS, R. **LabVIEW Graphical Programming**. 4.ed. New York, USA: McGraw-Hill, 2006.

KAFKA, L.; NOVÁK, O. FPGA- based fault simulator. In: IEEE DESIGN AND DIAGNOSTICS OF ELECTRONIC CIRCUITS AND SYSTEMS, 2006. **Anais...** [S.l.: s.n.], 2006. p.272 – 276.

KANO, K. **Semiconductor Devices**. Upper Saddle River, New Jersey 07458: [s.n.], 1998.

KAWAMOTO, A.; CHO, K.; DUTTON, R. Perspectives paper: first principles modeling of high-k gate dielectrics. **Journal of Computer-Aided Materials Design**, [S.l.], v.8, p.39–57, 2001. 10.1023/A:1015011207910.

KRUCKMEYER, K.; BUCHNER, S.; DASGUPTA, S. Single Event Transient (SET) Response of National Semiconductor's ELDRS-Free LM139 Quad Comparator. In: RADIATION EFFECTS DATA WORKSHOP, 2009 IEEE. **Anais...** [S.l.: s.n.], 2009. p.65 – 70.

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

LIN, S.; KIM, Y.; LOMBARDI, F. **Analysis and Design of Nanoscale CMOS Storage Elements for Single Event Hardening with Multiple Node Upset**. 2011. 1p. v.PP, n.99.

LUCAL, H. M. Arithmetic Operations for Digital Computers Using a Modified Reflected Binary Code. **Electronic Computers, IRE Transactions on**, [S.l.], v.EC-8, n.4, p.449 – 458, dec. 1959.

MARTIN, S. et al. **Compendium of International Irradiation Test Facilities**. Sevilha, Espanha: RADECS 2011, 2011.

MCMICKELL, M. B. et al. Rapid development of space applications with responsive digital electronics board and LabVIEW FPGA. In: ADAPTIVE HARDWARE AND SYSTEMS (AHS), 2010 NASA/ESA CONFERENCE ON. **Anais...** [S.l.: s.n.], 2010. p.79 – 81.

MESSENGER, G. C. Collection of Charge on Junction Nodes from Ion Tracks. **IEEE Transactions of Nuclear Science**, [S.l.], v.NS-26, n.6, December 1982.

MISKOV-ZIVANOV, N.; MARCULESCU, D. Multiple Transient Faults in Combinational and Sequential Circuits: a systematic approach. **Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on**, [S.l.], v.29, n.10, p.1614 – 1627, oct. 2010.

MONTORO, C. G.; SCHEIDER, M. C. **Mosfet Modeling for Circuits Analysis and Design**. London, Great Britain: World Scientific, 2007.

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

MUSSEAU, O. Semi-empirical modelization of charge funneling in a np diode. In: RADIATION AND ITS EFFECTS ON DEVICES AND SYSTEMS, 1991. RADECS 91., FIRST EUROPEAN CONFERENCE ON. **Anais...** [S.l.: s.n.], 1991. p.429 – 432.

NENZI, P.; VOGT, H. **NGSPICE Users Manual Version 22**. [S.l.: s.n.], 2010.

PAUCAR, R.; SILVA, C. Extgracción del Parámetro N_{ot} para el Modelo del Ruido 1/f usando las expresiones del modelo ACM. In: XVII WORKSHOP IBERCHIP 2011. **Anais...** [S.l.: s.n.], 2011.

POIVEY, C. et al. Development of a test methodology for single-event transients (SETs) in linear devices. **Nuclear Science, IEEE Transactions on**, [S.l.], v.48, n.6, p.2180 – 2186, dec 2001.

RABAEY, J. M.; CHANDRAKASAN, A.; NICOLIĆ, B. **Digital Integrated Circuits, a Design Perspective**. 2.ed. Upper Saddle River, New Jersey 07458: Pearson Education, Inc., 2003.

RABAEY, J. M.; CHANDRAKASAN, A.; NICOLIĆ, B. **SPICE MODEL LEVEL 49 for 0.25 micron CMOS Process**. Disponivel em: <<http://bwrc.eecs.berkeley.edu/icbook/models.htm>>.

REED, I. A class of multiple-error-correcting codes and the decoding scheme. **Information Theory, IRE Professional Group on**, [S.l.], v.4, n.4, p.38 – 49, september 1954.

RIBEIRO, I.; WIRTH, G.; KASTENSMIDT, F. Modeling the effects of broadening and degradation of single event transient pulses in integrated circuits. In: RADIATION AND ITS EFFECTS ON COMPONENTS AND SYSTEMS (RADECS), 2008 EUROPEAN CONFERENCE ON. **Anais...** [S.l.: s.n.], 2008. p.192 –197.

ROMERO, E.; PERETTI, G.; MARQUÉS, C. Diseño y Simulación de un Amplificador Operacional Miller Orientado a Bajo Ruido y Bajo Offset. In: VII WORKSHOP IBER-CHIP 2001. **Anais...** [S.l.: s.n.], 2001.

SCHÖNE, H. et al. Time-resolved ion beam induced charge collection (TRIBICC) in micro-electronics. **Nuclear Science, IEEE Transactions on**, [S.l.], v.45, n.6, p.2544 – 2549, dec 1998.

SCHRIMPF, R. D. Radiation Effects in Microelectronics. In: VELAZCO, R.; FOUILLAT, P.; REIS, R. (Ed.). **Radiations Effects on Embedded Systems**. [S.l.]: Springer, 2007. p.11 – 29.

SHANFIELD, Z. et al. Angular Dependence of Charge Funneling in Si and GaAs Devices. **Nuclear Science, IEEE Transactions on**, [S.l.], v.34, n.6, p.1341 – 1346, dec. 1987.

SHIMANOUCI, M. An Approach to Consistent Jitter Modeling for various Jitter Aspects and Measurement Methods. In: TEST CONFERENCE, 2001. PROCEEDINGS. INTERNATIONAL. **Anais...** [S.l.: s.n.], 2001. p.848 – 857.

SINGH, C.; PODDAR, K. Implementation of a VI-based multi-axis motion control system for automated test and measurement applications. In: TENCON 2008 - 2008 IEEE REGION 10 CONFERENCE. **Anais...** [S.l.: s.n.], 2008. p.1 – 6.

SOLOMON, G. Self-synchronizing Reed-Solomon codes (Corresp.). **Information Theory, IEEE Transactions on**, [S.l.], v.14, n.4, p.608 – 609, jul 1968.

SRINIVASAN, G.; TANG, H.; MURLEY, P. Parameter-free, predictive modeling of single event upsets due to protons, neutrons, and pions in terrestrial cosmic rays. **Nuclear Science, IEEE Transactions on**, [S.l.], v.41, n.6, p.2063 – 2070, dec. 1994.

STERPONE, L. et al. An Analytical Model of the Propagation Induced Pulse Broadening (PIPB) Effects on Single Event Transient in Flash-Based FPGAs. **Nuclear Science, IEEE Transactions on**, [S.l.], v.58, n.5, p.2333 – 2340, oct. 2011.

SUR-KOLAY, S. et al. Fsimac: a fault simulator for asynchronous sequential circuits. In: ASIAN TEST SYMPOSIUM, 9. **Anais...** [S.l.: s.n.], 2000. p.114 – 119.

SYNOPSYS. **HSPICE Applications Manual**. USA: [s.n.], 2005.

TERVO, R.; MWAKATAPANYA, S. Improved error recovery in an X.25 packet receiver. In: COMMUNICATIONS, COMPUTERS AND SIGNAL PROCESSING, 1993., IEEE PACIFIC RIM CONFERENCE ON. **Anais...** [S.l.: s.n.], 1993. v.2, p.423 –426 vol.2.

TRAVIS, J.; KRING, J. **LabVIEW for Everyone** : graphical programming made easy and fun. 3.ed. [S.l.]: Prentice-Hall PTR, 2006.

TURFLINGER, T. Single-event effects in analog and mixed-signal integrated circuits. **Nuclear Science, IEEE Transactions on**, [S.l.], v.43, n.2, p.594 – 602, apr 1996.

TUROWSKI, M. et al. Single event upset modeling with nuclear reactions in nanoscale electronics. In: MIXED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, 2008. MIXDES 2008. 15TH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2008. p.443 – 448.

VALDERAS, M. et al. SET Emulation Under a Quantized Delay Model. In: DEFECT AND FAULT-TOLERANCE IN VLSI SYSTEMS, 2007. DFT '07. 22ND IEEE INTERNATIONAL SYMPOSIUM ON. **Anais...** [S.l.: s.n.], 2007. p.68 – 78.

VELAZCO, R.; FOUILLAT, P.; REIS, R. **Radiation Effects on Embeded Systems**. Dordratch, the Netherlands: Springer, 2007.

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

WROBEL, F. et al. Radioactive Nuclei Induced Soft Errors at Ground Level. **Nuclear Science, IEEE Transactions on**, [S.l.], v.56, n.6, p.3437 – 3441, dec. 2009.

WU, C.-F.; HUANG, C.-T.; WU, C.-W. RAMSES: a fast memory fault simulator. In: INTERNATIONAL SYMPOSIUM ON DEFECT AND FAULT TOLERANCE IN VLSI SYSTEMS. **Anais...** [S.l.: s.n.], 1999. p.165 – 173.

XUEYAN, D.; LIYUN, W.; JINMEI, L. Effects of Charge Sharing on the Single Event Transient response of CMOS Logic Gates. **Chinese Journal of Semiconductors**, [S.l.], v.32, n.9, september 2011.

ZHOU, Q.; CHOUDHURY, M.; MOHANRAM, K. Design optimization for robustness to single-event upsets. In: VLSI TEST SYMPOSIUM, 2006. PROCEEDINGS. 24TH IEEE. **Anais...** [S.l.: s.n.], 2006. p.6 pp. – 207.