

UNIVERSIDADE FEDERAL DO RIO GRANDE
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
MESTRADO EM ENGENHARIA DE COMPUTAÇÃO

**Análise da robustez de votadores majoritários na presença
de falhas transientes em arquiteturas TMR e ATMR**

Ingrid Fortes Vasconcelos Oliveira

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Computação.

Orientador: Prof. Dr. Paulo Francisco Butzen

Rio Grande, 2020

Ficha Catalográfica

O48a Oliveira, Ingrid Fortes Vasconcelos.

Análise da robustez de votadores majoritários na presença de falhas transientes em arquiteturas TMR e ATMR / Ingrid Fortes Vasconcelos Oliveira. – 2020.

87 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande – FURG, Programa de Pós-Graduação em Computação, Rio Grande/RS, 2020.

Orientador: Dr. Paulo Francisco Butzen.

1. Microeletrônica 2. Confiabilidade 3. Tolerância a falhas
4. Redundância Modular Tripla 5. Redundância Modular Tripla Aproximada 6. Votadores Majoritários 7. Falha Transiente I. Butzen, Paulo Francisco II. Título.

CDU 004

Catálogo na Fonte: Bibliotecário José Paulo dos Santos CRB 10/2344

ATA DE SESSÃO DE DEFESA DE DISSERTAÇÃO DE MESTRADO

Ata nº 10/2020

Na data de 24 de agosto de 2020, às 13 horas e 15 minutos, ocorreu a Sessão de Defesa de Dissertação de Mestrado de Ingrid Fortes Vasconcelos Oliveira, que apresentou a dissertação intitulada “Análise da robustez de votadores majoritários na presença de falhas transientes em arquiteturas TMR e ATMR”, realizada sob a orientação do Prof. Dr. Paulo Francisco Butzen. A banca examinadora foi constituída pela Profs. Dra. Fernanda Gusmão de Lima Kastensmidt (UFRGS), Dr. Ricardo Augusto da Luz Reis (UFRGS) e Dr. Ewerson Luiz de Souza Carvalho (FURG), sob a presidência do orientador. Após a apresentação do trabalho, a banca arguiu a candidata e, a seguir, deliberou pela

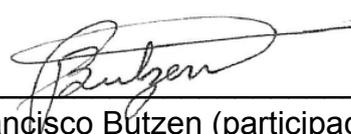
- (X) aprovação da Dissertação
- () aprovação da Dissertação, sugerindo modificações no texto
- () reprovação da Dissertação

Rio Grande, 24 de agosto de 2020.



Prof. Dra. Fernanda Gusmão de Lima Kastensmidt (participação remota)

Prof. Dr. Ricardo Augusto da Luz Reis (participação remota)

Prof. Dr. Ewerson Luiz de Souza Carvalho (participação remota)

Prof. Dr. Paulo Francisco Butzen (participação remota)
Orientador

*“Transforme as pedras que você tropeça
nas pedras de sua escada.”*

(Sócrates)

Resumo

O avanço da tecnologia possibilitou o aprimoramento dos dispositivos eletrônicos em relação a desempenho e funcionalidade. Como resultado, os dispositivos eletrônicos estão cada vez mais inseridos no cotidiano das pessoas. Esse grande avanço foi possível devido a miniaturização dos transistores, componentes base dos circuitos integrados. Entretanto, a redução das dimensões dos transistores tornou os circuitos mais sensíveis a falhas, principalmente oriundas da incidência de radiação. A suscetibilidade a essas falhas está diretamente relacionada a redução da tensão de alimentação e aumento da frequência de operação dos circuitos. Como resultado dessas reduções manter a operacionalidade dos circuitos mesmo na presença de falhas é de extrema importância, principalmente quando falamos de aplicações críticas. Técnicas de redundância de hardware são comumente utilizadas para tolerar esse tipo de falhas. Dentre essas técnicas, a mais conhecida e geralmente adotada é a técnica de Redundância Modular Tripla. A ideia por trás da técnica é que um módulo, que esteja propagando uma falha, seja mascarado pelos outros dois módulos livres de falhas, garantindo assim o mascaramento total de uma falha única. Porém, trata-se de uma técnica altamente custosa necessitando de um aumento de mais de 200% da área do circuito. A técnica de Redundância Modular Tripla Aproximada é utilizada em aplicações que buscam uma melhor relação entre área e cobertura de falhas com a finalidade de diminuir os custos de projeto através da combinação da técnica de redundância modular tripla e computação aproximada aplicada nos módulos da arquitetura. Ambas técnicas possuem um circuito votador que é responsável pela correta seleção da saída da arquitetura, sendo esse o ponto crítico de ambas arquiteturas. Caso uma falha incida no circuito votador, poderá ser observado um erro na saída da estrutura. Diferentes topologias foram propostas na literatura com o intuito de melhorar a robustez do bloco. Observa-se então a importância de realizar pesquisas com o intuito de melhorar a confiabilidade do circuito votador. Portanto, o objetivo principal dessa dissertação é investigar a robustez à falha transiente de diferentes topologias de votadores majoritários considerando aspectos relacionados às técnicas TMR e ATMR. Para isso foram selecionadas e modificadas diferentes implementações da função majoritária propostas na literatura, resultando num conjunto de catorze circuitos votadores. A análise divide-se em duas partes para cada arquitetura, a primeira consiste em uma análise da robustez à falha transiente a nível de leiaute, onde as regiões PN inversamente polarizadas críticas são identificadas, e a segunda consiste em uma análise elétrica do comportamento da falha transiente nas regiões PN críticas previamente obtidas, onde informações são extraídas a respeito da quantidade de energia necessária para que ocorra uma mudança no estado lógico da saída do circuito. Desta forma, essa dissertação discute as melhores opções de votadores majoritários para ambas arquiteturas, visto que a melhor opção de votador para uma arquitetura nem sempre terá o mesmo desempenho em outra. Esse trabalho busca otimizar a escolha do votador para arquiteturas aproximativas provendo um estudo dos vetores de entrada e seu impacto na robustez das implementações utilizadas. A metodologia de estimativa de suscetibilidade proposta nessa dissertação identifica as regiões ativas críticas do circuito e estima a robustez a uma falha transiente única no circuito votador. Além disso, essa metodologia não se limita somente na avaliação de votadores, podendo ser aplicada a qualquer outro circuito combinacional de forma a avaliar a sua robustez.

Palavras-chaves: microeletrônica, confiabilidade, tolerância a falhas, redundância modular tripla, redundância modular tripla aproximada, votadores majoritários, falha transiente.

Abstract

As a consequence of the technology scaling, the amount of energy necessary to store information and the increased operating frequency directly impacts the circuits reliability. Due to transistor shrinking effects, the field of fault tolerance has the challenge of maintaining acceptable levels of system operability even under the influence of faults. To deal with the radiation effects, especially considering a critical application, hardware redundancy techniques are the most used ones, such as the Triple Modular Redundancy. The technique guarantees 100% fault coverage if one of the architecture modules is faulty. However, it is a highly costly technique requiring an increase of more than 200% of the circuit area. The Approximate Triple Modular Redundancy technique is used in applications seeking a better trade-off between area overhead and fault coverage to reduce design costs through the combination of the triple modular redundancy technique and approximate computing applied to the modules. Both techniques have a voter circuit responsible for the correct selection of the architecture output. The voter is the critical point of both architectures. If a fault strikes the majority voter, there is no guarantee that the output will be correct. Several topologies have been proposed in the literature to mitigate this problem and increase the robustness of the block, showing the importance of analyzing their expected behavior under the presence of faults. Therefore, this master thesis aims to investigate the robustness of different topologies of majority voters to a transient fault considering aspects related to TMR and ATMR techniques. Different implementations of the majority function proposed in the literature were selected and modified, resulting in a set of fourteen voter circuits. The analysis is divided into two parts for each architecture. The first part consists of an analysis of the robustness to a transient fault at the layout-level, where the critical reverse-biased PN junctions are identified. The second part consists of an electrical analysis of a transient fault behavior in the previously obtained critical PN junctions. Information is extracted regarding the amount of energy necessary for a change in the logic state of the circuit output to occur. Thus, this master thesis discusses the best options for majority voters for both architectures, since the best voter option for one architecture will not always have the same performance in another. This work seeks to optimize the voter's choice for approximate architectures by providing a study of the input vectors and their impact on the robustness of the implementations used. The susceptibility estimation methodology proposed in this master thesis identifies the critical diffusion areas of the circuit and estimates the robustness to a single transient fault in the voter circuit. Besides, this methodology is not limited to the assessment of voters and can be applied to any other combinational circuit to assess its robustness.

Keywords: microelectronics, reliability, fault tolerance, triple modular redundancy, approximate triple modular redundancy, majority voters, single event transient.

Lista de ilustrações

Figura 1 – Transistores (a) NMOS e (b) PMOS.	24
Figura 2 – Simbologia do funcionamento dos transistores NMOS e PMOS utilizando chaves lógicas.	25
Figura 3 – Esquemático de transistor das portas lógicas: (a) inversor, (b) NAND2 e (c) NOR2	26
Figura 4 – Esquemático <i>transmission gate</i>	26
Figura 5 – Representações da porta lógica NAND2 à nível (a) lógico, (b) transistor, (c) diagrama <i>stick</i> e (d) leiaute (SCHVITZ, 2020)	27
Figura 6 – Representação dos estágios da incidência de uma partícula em uma região PN	30
Figura 7 – Exemplificação do mascaramento lógico.	31
Figura 8 – Exemplificação do mascaramento elétrico.	31
Figura 9 – Exemplificação do mascaramento por janela de amostragem.	32
Figura 10 – Exemplificação da arquitetura de um sistema TMR.	33
Figura 11 – Exemplificação da arquitetura de um sistema ATMR.	34
Figura 12 – Representação da relação de mintermos entre as funções F , G e H	35
Figura 13 – Esquemático de transistores da expressão lógica $A * (B + C)$	35
Figura 14 – Esquemático extraído da expressão booleana referente à função majoritária.	39
Figura 15 – Implementações do Votador Clássico: (a) NANDS e (b) NORS.	40
Figura 16 – Esquemático a nível de bloco lógico do Votador Kshirsagar proposto na literatura.	41
Figura 17 – Esquemático a nível de bloco lógico do Votador Ban proposto na literatura.	41
Figura 18 – Implementações do Votador Ban utilizando combinações de 3 diferentes implementações de multiplexadores para a porta lógica XOR utilizando lógica estática CMOS: (a) Ban (LIEBL, 2016)(OLIVEIRA; SCHVITZ; BUTZEN, 2018), (b) Ban 2, (c) Ban 3; e para a porta lógica XOR utilizando lógica PTL: (d) Ban 4, (e) Ban 5 e (f) Ban 6.	42
Figura 19 – Implementações do Votador CMOS: (a) votador CMOS (OLIVEIRA; SCHVITZ; BUTZEN, 2019) e (b) votador CMOS 2.	43
Figura 20 – Esquemático a nível de porta lógica do Votador Bala proposto na literatura.	44

Figura 21 – Implementações do Votador Bala: (a) votador Bala (OLIVEIRA; SCHVITZ; BUTZEN, 2019) e (b) votador Bala 2.	44
Figura 22 – Implementações do Votador Bala CMOS: (a) votador Bala CMOS (OLI- VEIRA; SCHVITZ; BUTZEN, 2019) e (b) votador Bala CMOS 2.	45
Figura 23 – Diagrama <i>stick</i> do Votador CMOS	48
Figura 24 – Metodologia adotada na análise a nível elétrico.	50
Figura 25 – Diagramas <i>sticks</i> dos circuitos votadores Clássico e CMOS.	73
Figura 26 – Diagramas <i>sticks</i> dos circuitos votadores Bala e Bala CMOS.	74
Figura 27 – Diagramas <i>sticks</i> dos circuitos votadores Ban.	75

Lista de tabelas

Tabela 1 – Tabela verdade da expressão lógica $A * (B + C)$ utilizando a técnica ATMR.	36
Tabela 2 – Comparativo das arquiteturas redundantes TMR e ATMR levando em consideração o exemplo da expressão lógica $A * (B + C)$.	36
Tabela 3 – Tabela comparativa das quatorze implementações apresentadas da função majoritária.	46
Tabela 4 – Resultado da análise da vulnerabilidade à falha SET das áreas ativas sensíveis (AAS) do Votador CMOS para o vetor de entrada "000".	49
Tabela 5 – Resultados obtidos para as topologias de votadores majoritários considerando a arquitetura TMR tradicional.	56
Tabela 6 – Simplificação da tabela de LET_{th} ($MeV.cm^{-2}/mg$) para arquitetura TMR considerando a tecnologia de $32nm$.	58
Tabela 7 – Quantidade de áreas ativas críticas obtidas pelo votador NANDS.	60
Tabela 8 – Quantidade de áreas ativas críticas obtidas pelo votador Ban 6.	60
Tabela 9 – Resultados sintetizados de áreas ativas críticas (AAC) classificados de acordo com a melhor ou pior ordem dos módulos para cada votador majoritário.	61
Tabela 10 – LET_{th} ($MeV.cm^{-2}/mg$) máximo e mínimo de todas as possíveis ordens de módulos para arquitetura ATMR considerando a tecnologia de $32nm$.	62
Tabela 11 – Resultados sintetizados de LET_{th} ($MeV.cm^{-2}/mg$) considerando a tecnologia de $32nm$.	63
Tabela 12 – Análise das áreas ativas sensíveis para o votador majoritário Clássico NANDS para todos os vetores de entrada.	77
Tabela 13 – Análise das áreas ativas sensíveis para o votador majoritário Clássico NORS para todos os vetores de entrada.	77
Tabela 14 – Análise das áreas ativas sensíveis para o votador majoritário CMOS para todos os vetores de entrada.	78
Tabela 15 – Análise das áreas ativas sensíveis para o votador majoritário CMOS 2 para todos os vetores de entrada.	78

Tabela 16 – Análise das áreas ativas sensíveis para o votador majoritário Bala para todos os vetores de entrada	78
Tabela 17 – Análise das áreas ativas sensíveis para o votador majoritário Bala 2 para todos os vetores de entrada	79
Tabela 18 – Análise das áreas ativas sensíveis para o votador majoritário Bala CMOS para todos os vetores de entrada	79
Tabela 19 – Análise das áreas ativas sensíveis para o votador majoritário Bala CMOS 2 para todos os vetores de entrada	79
Tabela 20 – Análise das áreas ativas sensíveis para o votador majoritário Ban para todos os vetores de entrada	80
Tabela 21 – Análise das áreas ativas sensíveis para o votador majoritário Ban 2 para todos os vetores de entrada	80
Tabela 22 – Análise das áreas ativas sensíveis para o votador majoritário Ban 3 para todos os vetores de entrada	80
Tabela 23 – Análise das áreas ativas sensíveis para o votador majoritário Ban 4 para todos os vetores de entrada	81
Tabela 24 – Análise das áreas ativas sensíveis para o votador majoritário Ban 5 para todos os vetores de entrada	81
Tabela 25 – Análise das áreas ativas sensíveis para o votador majoritário Ban 6 para todos os vetores de entrada	81
Tabela 26 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Clássico considerando a tecnologia de $32nm$	82
Tabela 27 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores CMOS considerando a tecnologia de $32nm$	83
Tabela 28 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Bala considerando a tecnologia de $32nm$	84
Tabela 29 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Bala CMOS considerando a tecnologia de $32nm$	85
Tabela 30 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Ban considerando a tecnologia de $32nm$ (parte I).	86
Tabela 31 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Ban considerando a tecnologia de $32nm$ (parte II).	87

Lista de abreviaturas e siglas

AAC	Área ativa crítica
AAS	Área ativa sensível
ATMR	<i>Approximate Triple Modular Redundancy</i>
CI	Circuito Integrado
CMOS	<i>Complementary Metal-Oxide-Semiconductor</i>
DTMR	<i>Diversity Triple Modular Redundancy</i>
FATMR	<i>Full Aproximate Triple Modular Redundancy</i>
FMR	<i>Fault Masking Ratio</i>
GND	Terminal de terra do circuito
LET	<i>Linear Energy Transfer</i>
LET _{th}	<i>Threshold LET</i>
MOS	<i>Metal Oxide Semiconductor</i>
MOSFET	<i>Metal Oxide Semiconductor Field Effect Transistors</i>
NMOS	Transistor MOS do tipo N
PMOS	Transistor MOS do tipo P
PLC	<i>Porta lógica complexa</i>
PTL	<i>Pass Transistor Logic</i>
SEB	<i>Single Event Burnout</i>
SEE	<i>Single Event Effects</i>
SEL	<i>Single Event Latchup</i>
SET	<i>Single Event Transient</i>

SEU	<i>Single Event Upset</i>
SHE	<i>Single Hard Error</i>
SPICE	<i>Simulation Program with Integrated Circuit Emphasis</i>
STMR	<i>Resettable Stateful Triple Modular Redundancy</i>
TMR	<i>Triple Modular Redundancy</i>
TP	Transistor de passagem
VDD	Terminal de alimentação do circuito
W	Largura do canal do transistor (<i>Width</i>)

Sumário

1	Introdução	17
1.1	Motivação	18
1.2	Objetivo	19
1.2.1	Principais Contribuições	19
1.3	Organização da Dissertação	20
2	Referencial Teórico	23
2.1	Transistor MOS	23
2.1.1	<i>Static CMOS Logic</i>	24
2.1.2	<i>Pass Transistor Logic</i>	25
2.1.3	Representação de Portas Lógicas	26
2.2	Tolerância a Falhas	28
2.2.1	<i>Single Event Effects</i>	28
2.2.2	Tipos de Mascaramento	31
2.2.3	Redundância Modular	32
2.2.3.1	Redundância Modular Tripla	33
2.2.3.2	Redundância Modular Tripla Aproximada	34
3	Votadores Majoritários	39
3.1	Votador NANDS e Votador NORS	40
3.2	Votador Ban	41
3.3	Votador CMOS	43
3.4	Votador Bala	43
3.5	Votador Bala CMOS	45
3.6	Conclusões do Capítulo	45
4	Metodologia	47
4.1	Votadores na arquitetura TMR	47
4.1.1	Análise Lógica	47
4.1.2	Análise Elétrica	49

4.2	Votadores na arquitetura ATMR	51
4.2.1	Análise Lógica	52
4.2.2	Análise Elétrica	52
5	Resultados	55
5.1	Resultados considerando arquitetura TMR	55
5.1.1	Análise Lógica	55
5.1.2	Análise Elétrica	57
5.2	Resultados considerando arquitetura ATMR	59
5.2.1	Análise Lógica	59
5.2.2	Análise Elétrica	62
6	Considerações Finais	65
6.1	Produção Científica	66
	Referências	67
	Apêndices	71
	APÊNDICE A Diagramas <i>Sticks</i>	73
	APÊNDICE B Resultados	77

1 Introdução

Pesquisas e descobertas realizadas na área da microeletrônica possibilitaram o atual progresso tecnológico. Esse avanço possibilitou que os dispositivos eletrônicos abrangessem uma variedade maior de aplicações, tornando-os cada vez mais presentes no cotidiano da população. Esse aumento na presença dos dispositivos eletrônicos é uma consequência direta da miniaturização dos dispositivos, resultado da evolução da tecnologia de fabricação dos circuitos integrados (CIs) e da complexidade desses.

Além disso, os dispositivos eletrônicos estão constantemente sob a influência de radiação. Devido a redução das dimensões dos dispositivos, estes tornaram-se mais sensíveis a partículas de radiação. Um circuito atingido por uma partícula pode ter seu correto funcionamento comprometido temporariamente ou até mesmo permanentemente (FERLET-CAVROIS; MASSENGILL; GOUKER, 2013). Portanto a área de tolerância a falhas tem como principal desafio manter níveis aceitáveis de operacionalidade de um sistema, mesmo na presença de falhas (KOREN; KRISHNA, 2010). Desta forma, a habilidade de tolerar falhas desempenha um importante papel no projeto dos CIs, principalmente ao considerarmos aplicações que necessitam um alto nível de segurança, como por exemplo nas áreas aeroespacial e médica.

As falhas de efeito singulares, conhecidas como *Single Event Effects* (SEEs), são falhas oriundas da radiação, que ocorrem quando partículas subatômicas atingem regiões sensíveis de um circuito (DODD; MASSENGILL, 2003)(FRIEDBERG; COPELAND, 2011). Dentre essas falhas, existem as falhas transientes, mais conhecidas como *Single Event Transients* (SET). A falha SET é decorrente do impacto de uma partícula em uma região PN reversamente polarizada, considerada como a região sensível de um circuito combinacional (BAUMANN, 2005)(FERLET-CAVROIS; MASSENGILL; GOUKER, 2013)(AUTRAN et al., 2014)(SCHVITZ et al., 2019), causando uma perturbação na região atingida (DODD; MASSENGILL, 2003). Com o aumento da sensibilidade dos circuitos aos efeitos de radiação, decorrente da evolução da tecnologia, as pesquisas envolvendo falhas SETs aumentaram consideravelmente nas últimas décadas, tanto em relação ao estudo comportamental do efeito (BAUMANN, 2004)(FERLET-CAVROIS; MASSENGILL; GOUKER, 2013), quanto no desenvolvimento e análise de uma variedade de modelos e técnicas de simulação e mitigação (AGUIAR et al., 2018)(SCHVITZ et al., 2019).

Para aumentar a confiabilidade de um sistema aos efeitos das falhas de radiação, principalmente no caso de aplicações críticas, é comum a utilização de técnicas de redundância de hardware. Uma das técnicas mais conhecida e amplamente adotada pelos projetistas para aumentar a robustez dos circuitos é a técnica de Redundância Modular Tripla (TMR) (NEUMANN, 1956). A técnica TMR garante 100% de cobertura de uma falha em um dos módulos da arquitetura através da triplicação do módulo que se deseja proteger.

1.1 Motivação

Apesar do TMR ser uma técnica eficaz para o que propõe, a arquitetura possui um ponto sensível que deve ser analisado. O circuito votador é o ponto crítico do TMR. Trata-se do bloco responsável por definir, a partir das saídas dos módulos, a saída correta do sistema. Caso uma falha incida no votador não há garantias de que o resultado da arquitetura está correto. Diversas topologias foram propostas na literatura com o objetivo de mitigar esse problema e aumentar a robustez do bloco (KSHIRSAGAR; PATRIKAR, 2009)(BAN; NAVINER, 2010)(BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016)(OLIVEIRA; SCHVITZ; BUTZEN, 2018)(OLIVEIRA; SCHVITZ; BUTZEN, 2019).

Visto que o TMR consiste basicamente em triplicar um módulo e utilizar um circuito votador para votar a saída majoritariamente, a técnica é altamente custosa necessitando de um aumento de mais de 200% da área do circuito. Esse custo é considerado tolerável em aplicações críticas, porém se torna um agravante para demais aplicações. A utilização da técnica de Redundância Modular Tripla Aproximada (ATMR) (SIERAWSKI; BHUVA; MASSENGILL, 2006) aparece para suprir a necessidade de aplicações onde é possível abrir mão de parte da cobertura de falhas em troca de um menor custo em área (SANCHEZ-CLEMENTE et al., 2016) (GOMES, 2018) através da combinação da técnica TMR e computação aproximada.

A técnica ATMR é composta por três módulos e um circuito votador. O módulo principal que implementa a função original (G) e dois módulos que implementam funções lógicas diferentes da função original, funções aproximadas (F e H). Isto é, existirão mintermos que irão divergir da função original. Para garantir que a técnica ATMR funcione corretamente é necessário seguir a seguinte regra: $F \subseteq G \subseteq H$. Essa regra assegura que a saída do módulo H só será 0 lógico quando a saída de G o for. O mesmo vale para o módulo F , mas para o 1 lógico.

Apesar da diferença existente na implementação dos módulos entre as arquiteturas re-

dundantes, onde no TMR é visado a cobertura de falhas e no ATMR a redução em área, conforme mencionado anteriormente, ambas possuem um circuito votador que define a saída. Visto que o circuito votador é totalmente responsável pela correta seleção da saída do sistema redundante, observa-se a importância de realizar pesquisas com o intuito de melhorar a confiabilidade desse circuito.

1.2 Objetivo

Desta forma, o objetivo principal dessa dissertação é investigar a robustez à falha SET de diferentes topologias de votadores majoritários considerando aspectos relacionados as técnicas TMR e ATMR. Foram selecionadas e modificadas diferentes implementações de votadores majoritários propostas na literatura (BAN; NAVINER, 2010)(BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016)(OLIVEIRA; SCHVITZ; BUTZEN, 2019), resultando na seleção de catorze circuitos votadores. Para atingir o objetivo principal, o trabalho foi dividido em duas etapas:

- Votadores na arquitetura TMR;
- Votadores na arquitetura ATMR.

Cada etapa é realizada em duas partes, onde a primeira parte consiste em realizar um análise da robustez à falha SET a nível de leiaute. Através dessa análise é possível identificar as regiões PN inversamente polarizadas sensíveis a falhas SET e as críticas, que com a incidência da falha um erro pode ser observado na saída do circuito votador.

A segunda parte consiste na análise elétrica do comportamento da falha SET nas regiões PN críticas, onde extraímos as informações de *LET threshold*. *LET threshold* trata-se da quantidade mínima de energia transferida por uma partícula para que ocorra uma mudança no estado lógico da saída do circuito. Através dos resultados obtidos das duas etapas é possível reunir as informações necessárias para escolher a melhor opção de um circuito votador de acordo com a arquitetura redundante a ser aplicada.

1.2.1 Principais Contribuições

A partir do objetivo previamente discutido, prover um estudo detalhado da robustez do circuito votador é a principal contribuição dessa dissertação. Para o desenvolvimento desse

trabalho diversos objetivos específicos podem ser evidenciados, possibilitando assim a geração de diversas contribuições.

- **Estudo e geração de diferentes arranjos de transistores para a função majoritária:** De acordo com o conhecimento da autora até o momento da escrita desta dissertação, não há trabalhos na literatura que comparem uma grande variedade de implementações de circuitos votadores no nível de transistores quanto a sua robustez a falhas transientes. Esse trabalho investiga o impacto do arranjo de transistores na susceptibilidade dos circuitos à falhas SET, explorando a utilização de portas lógicas complexas e também da realocação de portas inversoras da saída para as entradas do circuito;
- **Metodologia de avaliação da robustez de circuitos combinacionais:** Desenvolvimento de uma metodologia de estimativa da susceptibilidade de circuitos combinacionais à falhas SET no nível de leiaute. Apesar dessa metodologia ter sido desenvolvida para ser utilizada na avaliação de circuitos votadores na arquitetura TMR, essa pode ser aplicada a qualquer circuito combinacional. Além disso, por explorar uma simplificação do leiaute para identificar as áreas ativas sensíveis às partículas de radiação, pode-se considerar uma metodologia independente de tecnologia;
- **Adaptação da metodologia desenvolvida para a arquitetura ATMR:** Inserção da importância dos vetores de entrada na metodologia de estimativa de susceptibilidade de circuitos votadores à falha SET. Desta forma é possível explorar o impacto da ordem dos módulos da arquitetura ATMR na robustez do sistema, gerando um estudo detalhado para cada implementação de votador majoritário analisada para ambas arquiteturas;
- **Análise elétrica:** Avaliação da diferença no LET *threshold* através da simulação elétrica para classificar as diversas implementações quanto à robustez elétrica. Além disso, a análise elétrica possibilita a validação da metodologia de estimativa da susceptibilidade de circuitos combinacionais proposta.

1.3 Organização da Dissertação

A sequência dessa dissertação está organizada da seguinte forma: no segundo capítulo é realizada uma revisão completa onde são abordados conceitos para uma melhor compreensão dessa dissertação. No terceiro capítulo serão apresentadas as metodologias desenvolvidas e

utilizadas nas etapas, mencionadas anteriormente nesse capítulo, com o intuito de atingir o objetivo principal desse trabalho. Na sequência serão discutidos os resultados encontrados a partir da aplicação das metodologias apresentadas. Os resultados são divididos de acordo com a etapa realizada, visto que cada etapa se refere a uma arquitetura redundante. Ao final uma recapitulação dos resultados é apresentada frisando a importância dos resultados gerados para cada arquitetura, além de uma breve discussão a respeito do impacto do dimensionamento dos transistores nos resultados encontrados e sobre o impacto que as modificações realizadas tiveram na robustez à falha SET.

2 Referencial Teórico

Nesse capítulo serão abordados conceitos necessários para uma melhor compreensão desta dissertação. Primeiramente será realizado na seção 2.1 um breve *overview* sobre o transistor, seu funcionamento, dois estilos lógicos estáticos que serão utilizados na implementação dos circuitos votadores e finalmente as diferentes representações de blocos lógicos que serão encontradas e abordadas ao longo do texto. Uma introdução a tolerância à falhas será realizada na seção 2.2. Efeitos de radiação do tipo SEEs serão abordados na seção 2.2.1 focando na falha transiente SET presente na metodologia deste trabalho. Por fim, as técnicas de redundância de hardware utilizadas nessa dissertação serão apresentadas na seção 2.2.3.

2.1 Transistor MOS

Na primeira metade do século XX, os circuitos eletrônicos eram compostos por tubos de vácuo. Pesquisas foram realizadas na tentativa de substituir os tubos de vácuo por serem dispositivos eletrônicos volumosos, pouco confiáveis e custosos. Esses consumiam muita energia e queimavam rapidamente, exigindo elevado nível de manutenção.

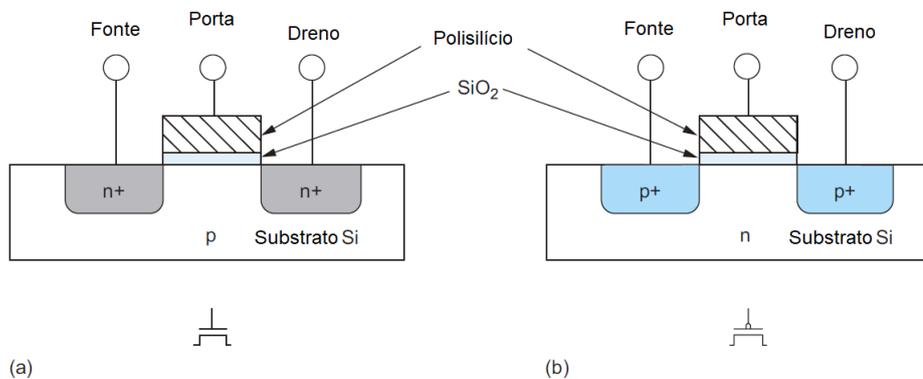
Em 1947, John Bardeen e Walter Brattain construíram o primeiro transistor de ponto de contato. Era composto por dois contatos de ouro em um cristal de germânio. Escolheu-se o nome transistor por se tratar de um resistor ou dispositivo semiconductor que amplifica sinais elétricos enquanto os transfere de um terminal de entrada para um terminal de saída (WESTE; HARRIS, 2015). A descoberta desse dispositivo revolucionou a indústria, impulsionando pesquisas que culminaram na construção dos primeiros transistores MOSFET (*Metal Oxide Semiconductor Field Effect Transistors*).

O semiconductor silício é o material-base para a produção da maioria dos circuitos integrados (TSIVIDIS, 1999). Através da técnica de dopagem, as indústrias injetam impurezas no material para modificar sua condutibilidade, tornando-o um semiconductor ou do tipo “n” ou do tipo “p”. Quando um semiconductor é dopado de forma que metade se torna do tipo “p” e a outra metade do tipo “n”, a borda onde as duas regiões se encontram é conhecida como junção PN (MALVINO, 1997).

Uma estrutura MOS (ou MOSFET) é criada através de várias camadas de materiais

condutores, semicondutores e isolantes, que são produzidas através de vários processos químicos que envolvem oxidação, implantação iônica, deposição, entre outras etapas (WESTE; HARRIS, 2015). Existem dois tipos de transistores MOS, como é possível observar na Figura 1. O transistor do tipo "n", ou NMOS, apresentado na Figura 1 (a) é composto por um substrato do tipo "p" e duas regiões de semiconductor do tipo "n" entre o terminal de porta, chamados terminais fonte e dreno. O transistor do tipo "p", ou PMOS, mostrado na Figura 1 (b) é complementar ao transistor NMOS. Trata-se de um dispositivo cujo substrato é do tipo "n" e entre o terminal de porta há duas regiões de semicondutores do tipo "p".

Figura 1 – Transistores (a) NMOS e (b) PMOS.



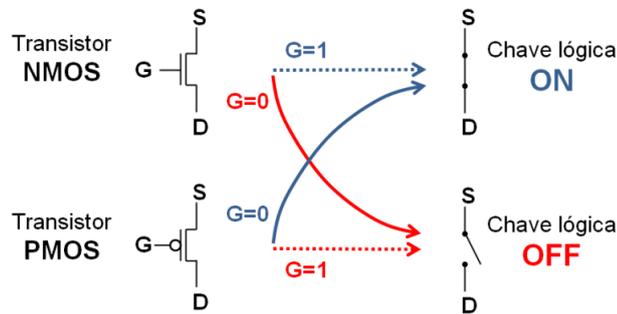
Fonte: Adaptado de (WESTE; HARRIS, 2015)

A Figura 2 ilustra o funcionamento dos transistores NMOS e PMOS aproximando seus comportamentos a de chaves lógicas. G é o sinal que controla o terminal de porta. Dependendo do valor que assume define se a chave estará fechada, havendo condução (*ON*), ou estará aberta, quando não há condução (*OFF*). O sinal conectado ao terminal de porta controla a existência de um caminho condutivo entre os terminais fonte (*S*) e dreno (*D*). Tendo isto em vista, o transistor NMOS comporta-se como uma chave fechada quando o sinal no terminal de porta (G) encontra-se em "1", caso contrário funciona como uma chave aberta. Já o transistor PMOS apresenta um comportamento complementar, funcionando como uma chave fechada quando o sinal de controle (G) for "0", caso contrário se comportará como uma chave aberta.

2.1.1 Static CMOS Logic

A implementação física destes componentes traz consigo algumas características que fazem com que os transistores tenham comportamento distinto do comportamento ideal anteriormente descrito. O transistor PMOS se comporta como uma chave ideal quando está passando

Figura 2 – Simbologia do funcionamento dos transistores NMOS e PMOS utilizando chaves lógicas.



Fonte: (BUTZEN et al.,)

"1" lógico, portanto passa um forte sinal "1". Entretanto, ao passar o "0" lógico, este sofre uma degradação e muitas vezes pode não ser mais interpretado como "0" lógico. Já o transistor NMOS opera como uma chave ideal para passagem do sinal "0" lógico e degrada o sinal "1" lógico (WESTE; HARRIS, 2015).

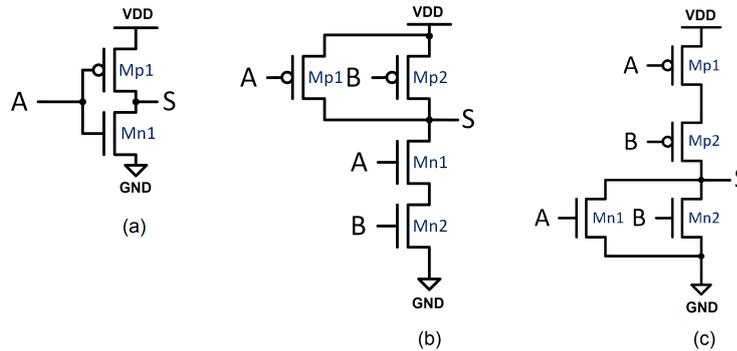
Um dos estilos lógicos mais utilizados pela indústria de circuitos integrados é a lógica estática CMOS (*Static Complementary Metal Oxide Semiconductor Logic Gates*) devido ao seu bom desempenho, imunidade à ruídos e amplamente conhecida e fácil metodologia de implementação (LAI; JIANG; CHU, 2005). Utiliza dois planos complementares, plano *Pull-Up* e plano *Pull-Down*. O plano *Pull-Up* é formado apenas por transistores do tipo "p" e conecta a saída no "1" lógico (VDD), enquanto o plano *Pull-Down* é formado por transistores do tipo "n", conectando a saída ao "0" lógico (GND).

A Figura 3 apresenta três portas lógicas utilizando lógica CMOS: inversor, NAND2 e NOR2. As portas lógicas são compostas por um conjunto de transistores PMOS e NMOS localizados em seus respectivos planos, *Pull-Up* e *Pull-Down*. Analisando as Figuras 3 (b) e (c), é possível perceber que os transistores que estão em série em um plano, no outro estão em paralelo, garantindo dessa forma a complementaridade dos planos. Desta forma, quando um plano estiver conduzindo, o outro estará em um estado não condutivo, possibilitando que a estrutura seja capaz de passar sinais "0" e "1" fortes.

2.1.2 Pass Transistor Logic

Dentre os estilos lógicos estáticos, a lógica PTL (*Pass Transistor Logic*) é utilizada em alguns casos para otimizar e substituir de forma eficiente a lógica CMOS (BERTACCO et

Figura 3 – Esquemático de transistor das portas lógicas: (a) inversor, (b) NAND2 e (c) NOR2

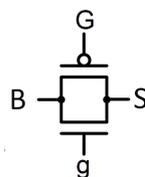


Fonte: O Autor.

al., 1997). A lógica PTL é conhecida por apresentar implementações otimizadas em relação ao número de transistores em comparação a lógica estática CMOS para circuitos aritméticos, como somadores e multiplicadores com uso predominantes de portas lógicas XORs (SHELAR, 2004).

Conforme explicado anteriormente, por uma característica da tecnologia, o transistor NMOS é eficiente na condução do "0" lógico e o transistor PMOS do "1" lógico. Quando um transistor, do tipo N ou do tipo P, é utilizado sozinho como uma chave imperfeita, este pode ser chamado de transistor de passagem (*pass transistor*) (WESTE; HARRIS, 2015). A Figura 4 ilustra uma estrutura chamada de *transmission gate*. Trata-se da combinação de dois transistores, um PMOS e outro NMOS, em paralelo, onde B é o sinal aplicado ao terminal de fonte compartilhado pelos transistores. Os sinais de controle aplicados aos terminais de porta dos transistores, respectivamente, NMOS e PMOS, são "g" e seu complemento "G". Desta forma, a estrutura quando estiver em estado de condução ("g"="1") passará ambos sinais fortes, "0" ou "1" lógico.

Figura 4 – Esquemático *transmission gate*



Fonte: O Autor.

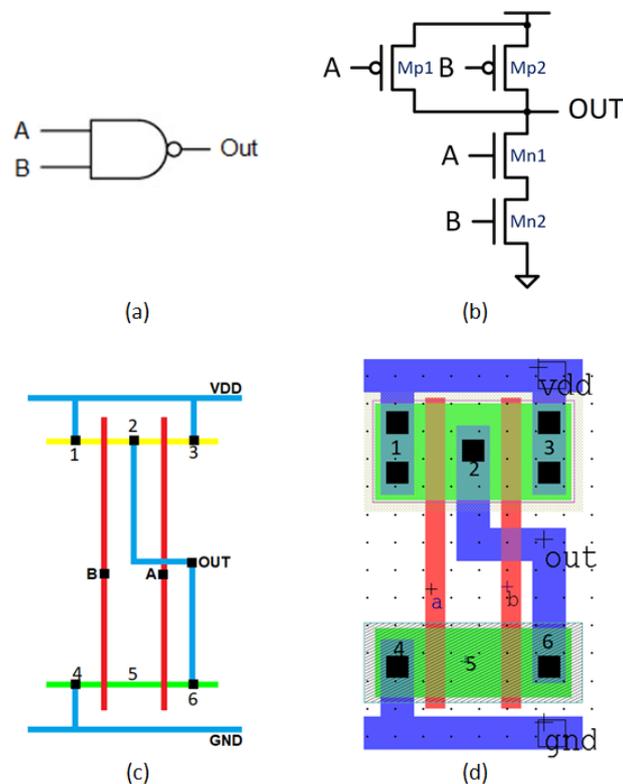
2.1.3 Representação de Portas Lógicas

As portas lógicas são construídas através do agrupamento de transistores, do tipo P e/ou do tipo N, para implementar uma determinada função lógica. Esses circuitos lógicos podem

ser representados de diferentes formas dependendo do nível de abstração escolhido, conforme ilustrado na Figura 5 para a porta lógica NAND2.

A Figura 5 (a) apresenta a representação à nível lógico. A porta lógica NAND2 é representada através de um símbolo e apenas suas entradas e saídas são especificadas. Uma representação mais detalhada do conteúdo da porta lógica é ilustrada na Figura 5 (b), onde é possível observar o arranjo de transistores utilizados na implementação da função NAND2. Já a Figura 5 (c) apresenta o diagrama *stick* da porta lógica. O diagrama *stick* trata-se da simplificação do leiaute, ilustrado na Figura 5 (d). No diagrama *stick* informações como área e regras de espaçamento não são consideradas, porém é possível observar o número de áreas ativas de cada plano e quais transistores compartilham a mesma área ativa nessa representação. Por outro lado, para utilização da representação em leiaute é necessário respeitar todas as regras de projeto, desta forma é possível obter as informações a cerca da área ocupada por cada área ativa, por exemplo.

Figura 5 – Representações da porta lógica NAND2 à nível (a) lógico, (b) transistor, (c) diagrama *stick* e (d) leiaute (SCHVITZ, 2020)



Fonte: O Autor.

2.2 Tolerância a Falhas

Devido aos efeitos resultantes da miniaturização dos transistores, rendimento e confiabilidade tornaram-se grandes preocupações no projeto e fabricação de circuitos integrados (CIs) (FRANCO; NAVINER; NAVINER, 2006), sendo a confiabilidade o fator preponderante quando estes circuitos são utilizados em aplicações críticas. São exemplos dessas aplicações, tecnologias aeroespacial e médica, que envolvem um alto nível de segurança, visto que uma falha pode acarretar na perda de vidas.

A confiabilidade pode ser definida como a probabilidade ou habilidade de um circuito executar corretamente sua função pré-estabelecida independentemente das condições a que esteja exposto durante um intervalo de tempo (BIROLINI, 2012). A confiabilidade dos circuitos é afetada devido a redução das dimensões dos dispositivos, que acarretaram em circuitos com maior sensibilidade a falhas de radiação, variabilidade de processo e efeitos de envelhecimento. A suscetibilidade a essas falhas de radiação está diretamente relacionada a redução da tensão de alimentação e o aumento da frequência de operação dos circuitos (HAZUCHA; SVENSSON, 2000). Com isso, a área de tolerância a falhas continuamente enfrenta o desafio de manter um nível aceitável de operacionalidade de um sistema, mesmo na presença de falhas (KOREN; KRISHNA, 2010). Portanto, a habilidade de tolerar falhas desempenha um importante papel no projeto de circuitos integrados, principalmente quando operam em ambientes hostis.

Visando tornar os circuitos mais robustos, consequentemente mais confiáveis, pesquisas na área são continuamente realizadas a fim de aprimorar e desenvolver técnicas de tolerância a falhas. Neste trabalho focaremos em técnicas de redundância de hardware, que são bastante utilizadas para mitigar efeitos de radiação. Esses tornaram-se foco de estudo desde o início dos anos 60 após um teste nuclear ocasionar na falha de um satélite de telecomunicações (VELAZCO; FOUILLAT; REIS, 2007). Nas próximas subseções abordaremos conceitos de falhas de efeitos singulares, focando em falhas transientes, tipos de mascaramento e, por fim, redundância de hardware, apresentando diferentes técnicas propostas na literatura.

2.2.1 *Single Event Effects*

As falhas de efeitos singulares, ou do inglês *Single Event Effects* (SEEs), são falhas oriundas da radiação. Essas são causadas quando partículas subatômicas altamente energéticas (como por exemplo, elétrons, prótons, nêutrons, partículas alfa, píons, múons ou outros íons pe-

sados) atingem regiões sensíveis de um circuito (DODD; MASSENGILL, 2003) (FRIEDBERG; COPELAND, 2011). Dependendo de vários fatores, o impacto dessas partículas pode causar desde nenhum efeito observável, uma interrupção transitória de uma operação, uma mudança no estado lógico ou até mesmo um dano permanente ao circuito integrado. Desta forma é possível classificar os efeitos singulares como não destrutivos, ou *Soft Errors*, e destrutivos, ou *Hard Errors* (GAILLARD, 2011).

Os SEEs classificados como destrutivos danificam permanentemente os circuitos afetados, isto é, nem mesmo uma reinicialização consegue resolver o defeito apresentado decorrente da colisão da partícula no circuito. Dentre os efeitos singulares classificados como destrutivos, podemos citar *Single Hard Error* (SHE), *Single Event Burnout* (SEB) and *Single Event Latchup* (SEL) (DODD, 2005) (SEXTON, 2003).

Já os SEEs classificados como não destrutivos ocorrem quando a colisão de partículas afeta temporariamente o funcionamento do circuito atingido porém sem danificá-lo permanentemente, onde o circuito atingido pode ser restaurado através da repetição da operação afetada (DODD et al., 2010). Nos circuitos sequenciais um efeito singular não destrutivo pode, por exemplo, alterar um bit armazenado em uma unidade de memória. Este efeito é chamado de *Single Event Upset* (SEU). Já nos circuitos combinacionais um SEE não destrutivo pode, por exemplo, alterar o estado lógico da saída do circuito através de uma perturbação na tensão desse, podendo se propagar pelo circuito e induzir um erro num elemento de memória. Este efeito é chamado de *Single Event Transient* (SET).

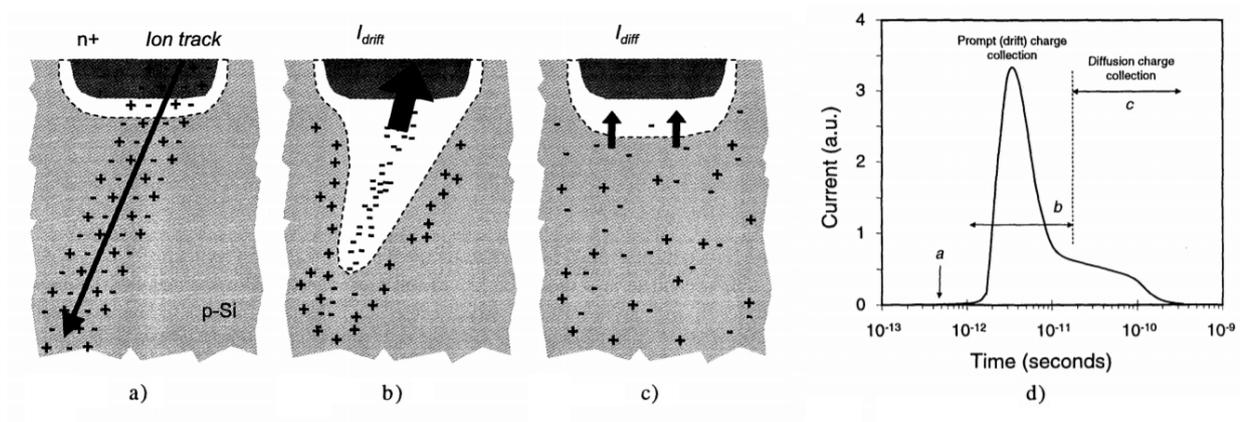
Single Event Transients (SET), ou falhas transientes, foram observadas primeiramente nos anos 80 com o desenvolvimento e aprimoramento de circuitos integrados para aplicações espaciais (KOGA et al., 1985) (HARBOE-SORENSEN et al., 1986). Naquele momento o foco das pesquisas estava no efeito das SEUs. As falhas SET só se tornaram um problema recorrente no final dos anos 90 com a miniaturização dos dispositivos atingindo dimensões submicro (<250nm) (FERLET-CAVROIS; MASSENGILL; GOUKER, 2013).

Uma falha SET é decorrente do impacto de uma partícula altamente energética em uma região PN inversamente polarizada em um circuito combinacional (DODD; MASSENGILL, 2003)(SCHVITZ et al., 2019). Essas regiões são consideradas as partes sensíveis dos circuitos (BAUMANN, 2005)(FERLET-CAVROIS; MASSENGILL; GOUKER, 2013)(AUTRAN et al., 2014)(SCHVITZ, 2020). A Figura 6 apresenta os estágios da incidência da partícula em uma

região ativa inversamente polarizada. A Figura 6 (a) ilustra o momento no qual a partícula incide na região ativa sensível de um circuito, produzindo pares de elétron-lacuna. Nessa imagem é mostrado o comportamento em uma junção PN de um transistor NMOS, uma situação similar ocorreria em uma junção PN de um transistor PMOS exceto que haveria coleção de lacunas e não de elétrons (BAUMANN, 2004).

A Figura 6 (b) apresenta o caminho resultante da passagem da partícula pela região de depleção do transistor. A distribuição de carga sem equilíbrio acarreta na deformação temporária em forma de funil (efeito *funneling*) ao longo da trajetória do evento, aprimorando a coleção de cargas por deriva (do inglês *drift*). Como resultado há a criação de uma alta corrente transiente no nodo afetada pela partícula. A Figura 6 (c) apresenta o processo de difusão que coleta os portadores resultantes, que ocorre após a finalização da etapa de coleção de carga e do efeito *funneling*. O gráfico da corrente transiente gerada com a incidência da partícula no transistor é apresentado na Figura 6 (d), onde é possível observar o comportamento da corrente na região afetada nos estágios previamente discutidos e ilustrados em (a), (b) e (c).

Figura 6 – Representação dos estágios da incidência de uma partícula em uma região PN



Fonte: Adaptada de (BAUMANN, 2004)

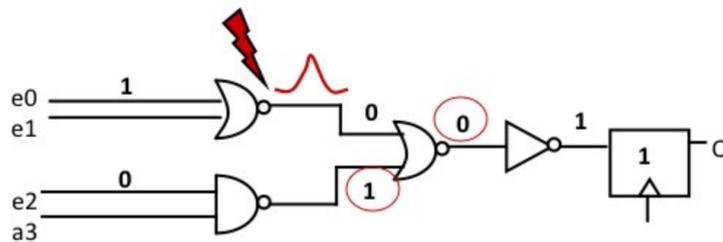
Com a evolução da tecnologia e a crescente sensibilidade dos circuitos aos efeitos de radiação, as pesquisas envolvendo falhas SET aumentaram consideravelmente nas últimas três décadas, tanto em relação ao estudo comportamental do efeito, quanto no desenvolvimento de uma variedade de modelos e técnicas de simulação e de mitigação. Nas próximas seções focaremos nos tipos de mascaramento de falhas e nas técnicas que utilizam redundância de hardware para aumentar a robustez dos circuitos a falhas transientes.

2.2.2 Tipos de Mascaramento

A ocorrência de uma falha em um circuito nem sempre significa que um erro será observado na sua saída. Isso acontece porque as falhas que ocorrem em nível de dispositivo podem sofrer algum tipo de mascaramento, impedindo que se propaguem e atinjam outros elementos do circuito. No geral, os mascaramentos são classificados de acordo com a sua origem.

A Figura 7 apresenta a exemplificação do mascaramento lógico. Observa-se na figura a ocorrência de uma falha SET na primeira porta lógica NOR do circuito ilustrado, resultando numa modificação indesejada de sua saída de "0" lógico para "1" lógico. Apesar dessa falha se propagar até o próximo elemento do circuito, esse é mascarado ao final devido a lógica da segunda porta lógica NOR. Essa segunda NOR recebe a saída falha como uma de suas entradas, porém como sua outra entrada é "1" lógico, sua saída permanece inalterada. Isso ocorre porque a entrada "1" lógico numa porta lógica NOR denomina sua saída para "0" lógico independentemente da outra entrada conectada a essa porta.

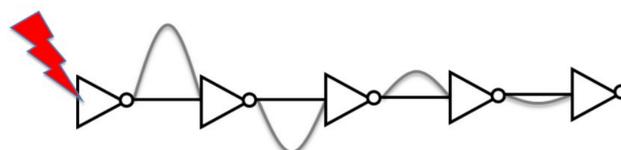
Figura 7 – Exemplificação do mascaramento lógico.



Fonte: Extraído de (AZAMBUJA, 2013)

O mascaramento elétrico ocorre quando a lógica do circuito enfraquece a propagação do pulso. Esse tipo de mascaramento pode ser melhor observado na Figura 8. Na figura, a porta lógica inversor é atingida por uma falha SET e o pulso transiente se propaga até a saída atingindo os elementos seguintes do circuito. Ao se propagar pela sequência de três inversores, o pulso acaba enfraquecendo até ser eletricamente mascarado ao atingir o último inversor, por não ter energia o suficiente para continuar se propagando.

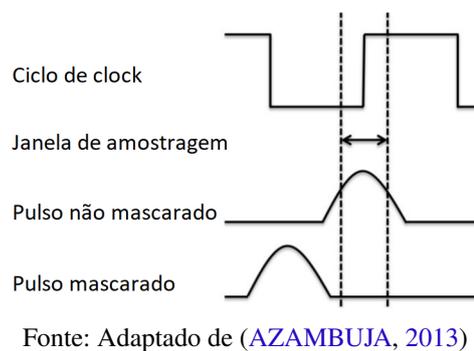
Figura 8 – Exemplificação do mascaramento elétrico.



Fonte: Extraído de (AZAMBUJA, 2013)

A Figura 9 ilustra o mascaramento por janela de amostragem. Observando o exemplo apresentado na figura citada, podemos observar o ciclo de *clock* seguido de sua janela de amostragem. Além disso, são apresentados dois cenários. No primeiro cenário o pulso ocorre e se mantém por todo o período da janela de amostragem, acarretando no armazenamento do erro. Portanto, nesse caso a falha não é mascarada. Já o segundo cenário apresenta justamente a ocorrência do mascaramento, onde o pulso não atinge a janela de amostragem da lógica sequencial.

Figura 9 – Exemplificação do mascaramento por janela de amostragem.



2.2.3 Redundância Modular

Com o intuito de aumentar a confiabilidade de um sistema, a redundância é uma das técnicas mais explorada na mitigação de falhas transientes. Técnicas de redundância de hardware são comumente utilizadas pelos projetista com o intuito de aumentar a robustez dos circuitos. Esta seção irá explorar técnicas de redundância de hardware que exploram o conceito de replicação de módulos e circuitos votadores.

Dentre as técnicas de redundância, a mais conhecida é a técnica de Redundância Modular Tripla (TMR) (NEUMANN, 1956). Diversas pesquisas foram desenvolvidas na literatura com o propósito de aprimorar a estrutura da arquitetura através de variações da técnica TMR, como por exemplo os trabalhos que exploram as técnicas variantes: Redundância Modular Tripla Aproximada (ATMR) (SIERAWSKI; BHUVA; MASSENGILL, 2006) (SANCHEZ-CLEMENTE et al., 2012) (GOMES; KASTENSMIDT, 2013), *Resettable Stateful TMR* (STMR) (MATSUMOTO; UEHARA; MORI, 2011), TMR Diversificado (DTMR) (HIARI; SADEH; RAWASHDEH, 2012) e *Full-ATMR* (FATMR) (GOMES, 2018). Além disso, alguns trabalhos focaram no aprimoramento dos votadores majoritários através de propostas de novas implementações dos circuitos a nível de bloco lógico (KSHIRSAGAR; PATRIKAR, 2009) (BAN; NAVINER, 2010) (BALA-

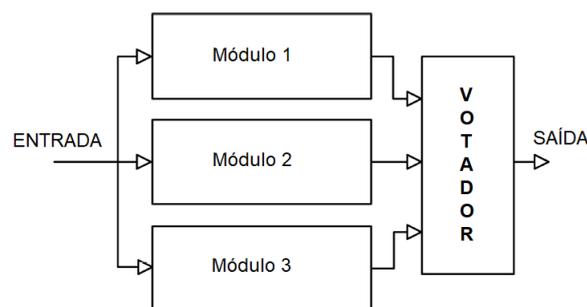
(SUBRAMANIAN; PRASAD; MASTORAKIS, 2016), estudos comparativos analisando um conjunto de votadores a nível de transistor (LIEBL, 2016) (OLIVEIRA; SCHVITZ; BUTZEN, 2018) e um estudo comparativo realizando uma análise a nível de leiaute (OLIVEIRA; SCHVITZ; BUTZEN, 2019) com o mesmo conjunto de votadores utilizados em (OLIVEIRA; SCHVITZ; BUTZEN, 2018). Nas próximas subsecções serão apresentadas a técnica TMR e uma abordagem variante que utiliza computação aproximada para lidar com os custos envolvidos na aplicação original.

2.2.3.1 Redundância Modular Tripla

A Redundância Modular Tripla (TMR), do inglês *Triple Modular Redundancy*, é uma técnica de redundância de hardware amplamente adotada para lidar com falhas oriundas da radiação. A Figura 10 apresenta a arquitetura TMR composta por três módulos idênticos, que executam uma mesma função, conectados a um circuito votador responsável pela saída do sistema.

A ideia por trás da técnica TMR é que um módulo, que esteja propagando uma falha, seja mascarado pelos outros dois módulos livres de falhas através da implementação da função majoritária, garantindo assim o mascaramento total de uma falha única. A grande vantagem da utilização da redundância modular tripla é a cobertura de 100% das falhas no caso de uma falha única em um dos módulos. Entretanto, a técnica tem um ponto crítico: o votador majoritário. Em um sistema TMR ideal, a confiabilidade do circuito votador deveria ser bem maior do que a confiabilidade dos demais componentes da arquitetura. Na prática não é o que ocorre (SADEGHI; SOLTAN; KHAYYAMBASHI, 2015), caso a falha incida no circuito votador não é possível garantir que haverá mascaramento total da falha.

Figura 10 – Exemplificação da arquitetura de um sistema TMR.



Fonte: Adaptado de (VIAL et al., 2008).

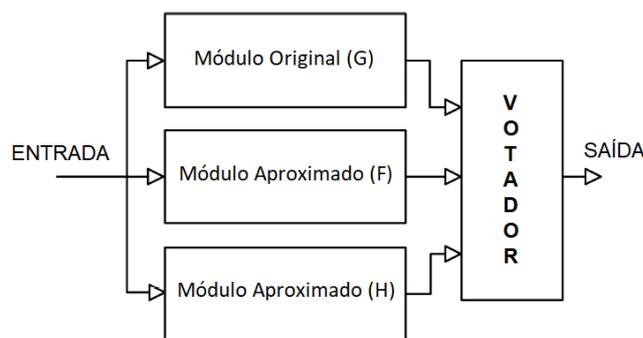
Outro ponto importante a ser levantado sobre a técnica TMR é o aumento em área de

mais de 200%. Esse valor leva em consideração a adição dos dois módulos extra e a área ocupada pelo votador majoritário. Para aplicações críticas, a penalidade em área pode ser considerada aceitável ao levar em consideração a garantia de cobertura de falhas que a técnica promete. Porém, aplicações que podem ser flexíveis quanto a confiabilidade, a utilização da redundância modular tripla torna-se um custo alto demais para o projeto.

2.2.3.2 Redundância Modular Tripla Aproximada

Visando um equilíbrio aceitável entre confiabilidade e área, surgiu a técnica de Redundância Modular Tripla Aproximada (ATMR) (SIERAWSKI; BHUVA; MASSENGILL, 2006), do inglês *Approximate Triple Modular Redundancy*. A técnica ATMR utiliza lógica aproximada nos módulos para diminuir o aumento de área que a técnica original TMR gera. Porém, para isso, existe uma penalidade na cobertura de falhas (SANCHEZ-CLEMENTE et al., 2016)(GOMES; KASTENSMIDT, 2013). A arquitetura ATMR é similar a da técnica TMR. Utilizam-se três módulos conectados a um votador majoritário responsável pela saída do sistema, conforme ilustrado na Figura 11. A diferença está na implementação desses módulos, que não são idênticos, utilizando lógica aproximativa.

Figura 11 – Exemplificação da arquitetura de um sistema ATMR.

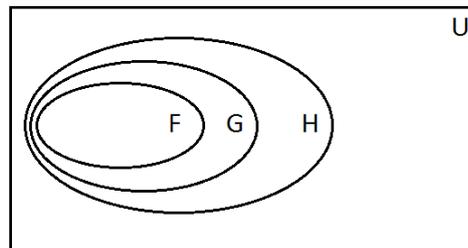


Fonte: O Autor.

Para entender a técnica de redundância modular tripla aproximada e como essa se diferencia da técnica TMR, é necessário compreender primeiro o que é um circuito aproximado. Um circuito é considerado aproximado quando implementa uma função lógica que diverge alguns de seus mintermos dos da função original. Na arquitetura de um sistema ATMR, um dos módulos implementa a função original (G) e os outros dois módulos implementam funções aproximadas da função original (F e H). Os módulos aproximados divergem em alguns vetores de entrada do módulo original, porém sempre obedecendo a regra de que $F \subseteq G \subseteq H$.

O módulo que implementa a função F é denominado *under-approximated*. Significa que a função aproximada F está contida na função original G , conforme ilustra Figura 12. Para que $F \subseteq G$ seja verdadeiro, G deve conter todos os mintermos existentes em F . Por outro lado, o módulo que implementa a função H é denominado *over-approximated*. Isto é, a função aproximada H contém a função original G . Portanto para que $G \subseteq H$ seja verdadeiro, H deve conter todos os mintermos existentes na função original G .

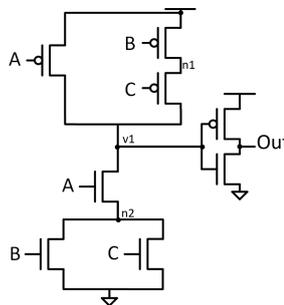
Figura 12 – Representação da relação de mintermos entre as funções F , G e H .



Fonte: O Autor.

Para exemplificar a geração das funções aproximadas de acordo com a regra ilustrada pela Figura 12, será utilizada a função $G = A * (B + C)$ apresentada na Figura 13. A Tabela 1 apresenta a tabela verdade de uma arquitetura ATMR, onde G representa o módulo que implementa a função original $G = A * (B + C)$, F representa o módulo aproximado que implementa a função $F = A * B$, H representa o módulo aproximado que implementa a função $H = A$ e Out representa a saída do votador majoritário.

Figura 13 – Esquemático de transistores da expressão lógica $A * (B + C)$.



Fonte: O Autor.

Observando os valores das saídas dos módulos (G , F e H) é possível observar que em dois vetores de entrada existe uma divergência dos mintermos em relação ao módulo original G . Essa divergência ocorre respeitando a regra $F \subseteq G \subseteq H$. Portanto, quando G for "0" lógico, F obrigatoriamente também será "0" lógico. Desta forma, os únicos vetores de entrada que

podem divergir do módulo original G são quando a saída desse for "1" lógico. O mesmo ocorre para a função aproximada H , porém de forma complementar. Quando G for "1" lógico, H obrigatoriamente será "1" lógico. E os vetores de entrada que podem divergir do módulo original G para geração do módulo aproximado H são quando a saída do módulo G for "0" lógico.

Tabela 1 – Tabela verdade da expressão lógica $A * (B + C)$ utilizando a técnica ATMR.

A	B	C	G	F	H	Out
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	0	0	1	0
1	0	1	1	0	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

A técnica TMR possui um aumento de 200% em área conforme mostrado na Tabela 2, considerando apenas os módulos da arquitetura. O tamanho do votador é desconsiderado neste exemplo. Os valores de área apresentados na Tabela 2 e no decorrer dessa dissertação refletem apenas a quantidade de transistores utilizados nas implementações, demais detalhes de área física real de um circuito são abstraídos. Por não haver divergência entre os mintermos dos módulos, a técnica garante 100% de proteção dos vetores de entrada. Por tratar-se de uma técnica que possui cobertura total de uma falha única, considerando os casos possíveis de ocorrência de uma falha (8 casos por módulo, 24 no total), a arquitetura TMR garante que todos os casos estarão protegidos.

Tabela 2 – Comparativo das arquiteturas redundantes TMR e ATMR levando em consideração o exemplo da expressão lógica $A * (B + C)$.

Arquitetura Redundante	Total de Transistores	Aumento de Área	Vetores Despr.	% Vetores Protegidos	Casos Críticos/ Total de Casos	% Casos Protegidos
TMR	24	200%	0	100%	0/24	100%
ATMR	14	75%	2	75%	4/24	83%

A utilização de módulos aproximados (F e H) da técnica ATMR proporciona uma troca entre a cobertura de falhas e área. Observando a Tabela 2, o aumento de área para a técnica ATMR é de apenas 75%, o que significa uma redução em área de 62,5% comparada a técnica TMR. Por outro lado, a cobertura dos vetores de entrada é de 75%, uma redução de 25%. Em relação aos casos protegidos apresenta uma taxa de 83%, uma redução de 17% comparado aos

valores do TMR. É possível perceber que o ganho em área é consideravelmente maior do que a perda de proteção.

3 Votadores Majoritários

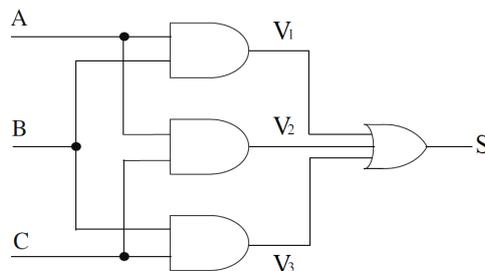
Nesse capítulo será apresentada a função majoritária implementada como circuito lógico e sua importância nas arquiteturas redundantes. Além disso, serão apresentadas diferentes implementações do votador majoritário que foram selecionadas para essa dissertação. Essas implementações foram geradas e derivadas de topologias existentes na literatura. Ao final do capítulo será apresentado um breve comparativo entre os votadores majoritários selecionados.

O votador majoritário é o componente responsável por comparar a saída de cada módulo bit-a-bit para votar a saída correta das arquiteturas que utilizam redundância modular, como o TMR e o ATMR. A função do votador possui três entradas (A , B e C) e uma saída S . A Equação (3.1) é a expressão booleana correspondente ao comportamento da seleção majoritária da saída do votador.

$$S = AB + AC + BC \quad (3.1)$$

A Figura 14 apresenta o esquemático que implementa a Equação (3.1) utilizando portas lógicas. A , B e C representam as saídas dos módulos das técnicas apresentadas na subseção 2.2.3, sendo as entradas do votador, e S é a saída do circuito. O circuito votador é totalmente responsável pela correta seleção da saída do sistema redundante. Observa-se que existe uma grande dependência por parte da arquitetura redundante em relação ao votador majoritário, visto que a saída do votador e a saída do sistema TMR são a mesma. Desta forma destaca-se a importância de assegurar a confiabilidade desse circuito. Caso uma falha incida no circuito votador, poderá ser observado um erro na saída da arquitetura.

Figura 14 – Esquemático extraído da expressão booleana referente à função majoritária.



Fonte: Adaptado de (KSHIRSAGAR; PATRIKAR, 2009).

Diferentes topologias foram propostas com o intuito de melhorar a robustez do bloco. Foram selecionadas e modificadas diferentes implementações de votadores majoritários propostas

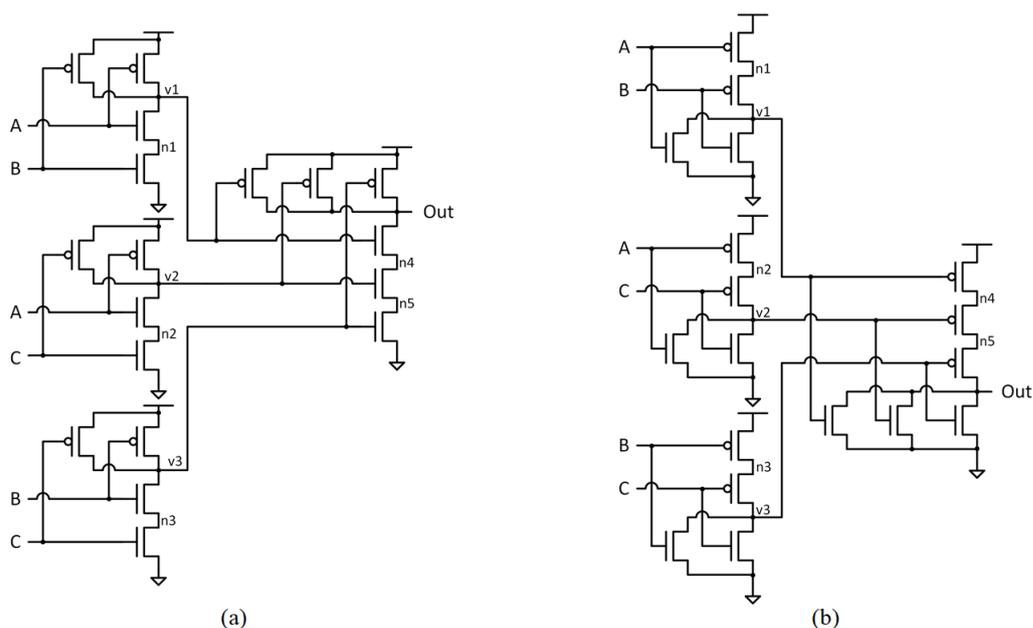
na literatura, resultando na seleção de catorze circuitos votadores para fazer parte da metodologia desta dissertação.

3.1 Votador NANDS e Votador NORS

A Figura 15 apresenta duas versões de circuitos votadores da implementação do esquemático da função booleana canônica do votador majoritário apresentado na Figura 14. A utilização dessas versões na análise tem o intuito de investigar o impacto da remoção dos inversores presentes no esquemático de transistores do circuito lógico. Essa decisão foi tomada levando em consideração resultados coletados de estudos preliminares realizados em (OLIVEIRA; SCHVITZ; BUTZEN, 2018)(OLIVEIRA; SCHVITZ; BUTZEN, 2019), que demonstraram a sensibilidade da porta lógica inversor a falha transiente.

Na Figura 15 (a) observamos a implementação do votador Clássico utilizando apenas portas lógicas NAND e na Figura 15 (b) apenas portas lógicas NOR. Desta forma, elimina-se em ambas as versões os quatro inversores necessários na implementação do circuito lógico da Figura 14, além de reduzir em torno de 30% o número de transistores utilizados.

Figura 15 – Implementações do Votador Clássico: (a) NANDS e (b) NORS.

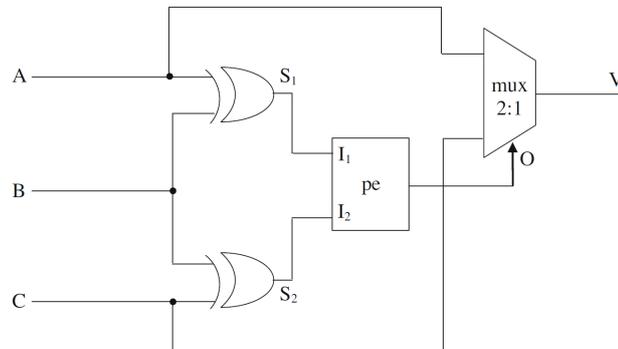


Fonte: Adaptado de (LIEBL, 2016).

3.2 Votador Ban

A Figura 16 apresenta o esquemático a nível de bloco lógico do circuito votador proposto por (KSHIRSAGAR; PATRIKAR, 2009). O votador Kshirsagar foi projetado para ser tolerante a falhas através da utilização de um codificador de prioridade (*pe*) e um multiplexador (*mux*) para implementar a função majoritária na Equação (3.1). O modelo foi proposto para mascarar falhas a nível lógico, tais como as falhas de colagem do tipo *Stuck-at* e falhas transientes. Entretanto, apesar de robusto para falhas a nível lógico, trata-se de um votador custoso em relação a área. Dependendo da implementação utilizada, pode necessitar em torno de 36 transistores (OLIVEIRA; SCHVITZ; BUTZEN, 2019).

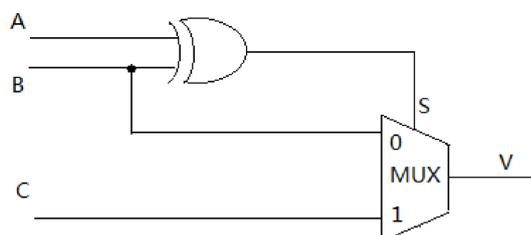
Figura 16 – Esquemático a nível de bloco lógico do Votador Kshirsagar proposto na literatura.



Fonte: Extraído de (KSHIRSAGAR; PATRIKAR, 2009)

Para resolver o problema de área do votador Kshirsagar, o votador Ban foi proposto por (BAN; NAVINER, 2010). Esta nova solução trata-se de uma simplificação do circuito Kshirsagar, utilizando o mesmo conceito de multiplexador, sem utilizar um codificador de prioridade e reduzindo o número de portas lógicas XOR de duas para uma, conforme podemos observar na Figura 17, reduzindo assim 50% dos custos em relação a área. De acordo com (BAN; NAVINER, 2010), um votador com menos transistores possui menor probabilidade de ser afetado por uma falha. Portanto, a maior vantagem deste circuito é o tamanho de área reduzido.

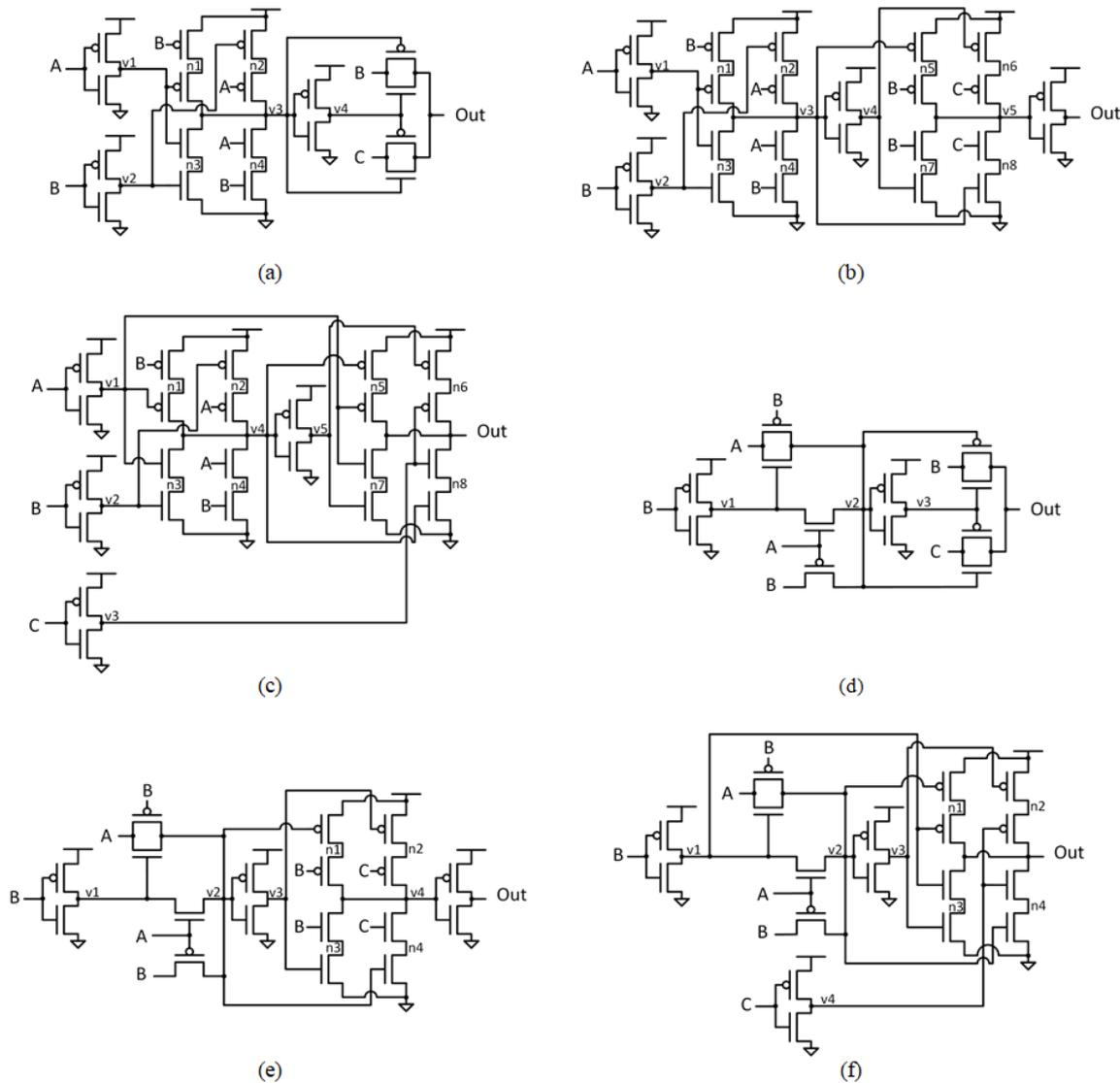
Figura 17 – Esquemático a nível de bloco lógico do Votador Ban proposto na literatura.



Fonte: Extraído de (BAN; NAVINER, 2010)

A Figura 18 ilustra seis diferentes implementações a nível de transistor do Votador Ban. Os circuitos foram gerados através da combinação de duas implementações da porta lógica XOR e três implementações do multiplexador.

Figura 18 – Implementações do Votador Ban utilizando combinações de 3 diferentes implementações de multiplexadores para a porta lógica XOR utilizando lógica estática CMOS: (a) Ban (LIEBL, 2016)(OLIVEIRA; SCHVITZ; BUTZEN, 2018), (b) Ban 2, (c) Ban 3; e para a porta lógica XOR utilizando lógica PTL: (d) Ban 4, (e) Ban 5 e (f) Ban 6.



Fonte: O Autor.

A escolha das XORs e MUXs utilizados nessa dissertação foi determinada a partir de um estudo preliminar. Para as portas lógicas XORs, optou-se por explorar os estilos lógicos estáticos citados anteriormente, CMOS e PTL, escolhendo assim uma implementação com cada estilo lógico. A escolha foi baseada no tamanho e na robustez das portas. Para os multiplexadores, além de optar pela exploração dos estilos lógicos estáticos, optou-se por utilizar uma terceira

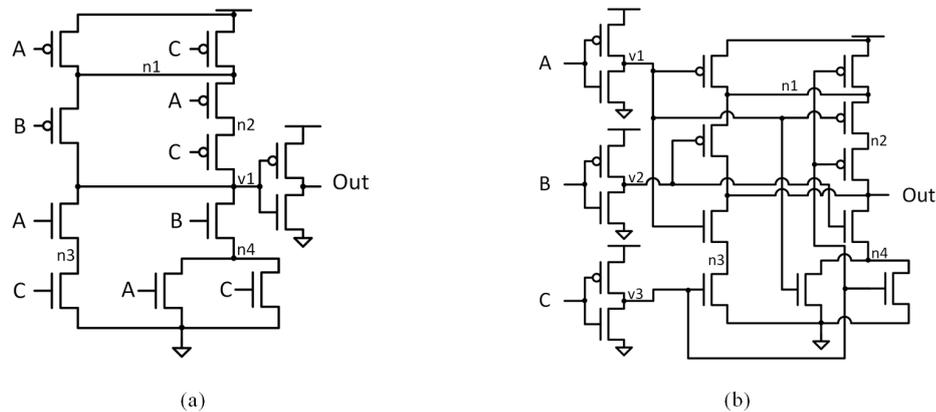
opção, uma versão da lógica estática CMOS transferindo o inversor presente na saída para a entrada.

3.3 Votador CMOS

O circuito apresentado na Figura 14 possui muitos nodos internos, o que o torna mais sensível a falhas a nível lógico, como demonstrado em (KSHIRSAGAR; PATRIKAR, 2009). A estrutura ilustrada na Figura 19 (a) parte do conceito de diminuir os nodos existentes utilizando uma porta lógica complexa, chamado neste trabalho de votador CMOS.

A maior vantagem dessa solução é a utilização de um número consideravelmente baixo de transistores para projetá-la, apenas 12 transistores foram utilizados. Porém, a estrutura apresenta uma porta lógica inversor conectado a saída do circuito. Visando explorar a robustez da estrutura, mesmo com um ganho extra equivalente a quatro transistores em área, removeu-se o inversor da saída e adicionou-se três inversores na entrada, gerando o votador CMOS 2 apresentado na Figura 19 (b).

Figura 19 – Implementações do Votador CMOS: (a) votador CMOS (OLIVEIRA; SCHVITZ; BUTZEN, 2019) e (b) votador CMOS 2.



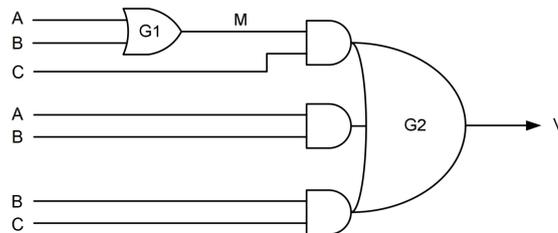
Fonte: O Autor.

3.4 Votador Bala

Seguindo como referência o votador CMOS, o votador apresentado na Figura 20 foi proposto por (BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016) com o intuito de ser mais robusto que os circuitos Clássico (Figura 14), Kshirsagar e Ban. O modelo é composto por uma porta lógica OR de duas entradas cuja saída é conectada a uma das entradas da porta lógica

complexa. Essa topologia ao ser proposta, assim como as demais, foi analisada a nível lógico, porém com um diferencial, sua taxa de mascaramento de falhas também foi analisada resultando em valores maiores que os circuitos utilizados na comparação.

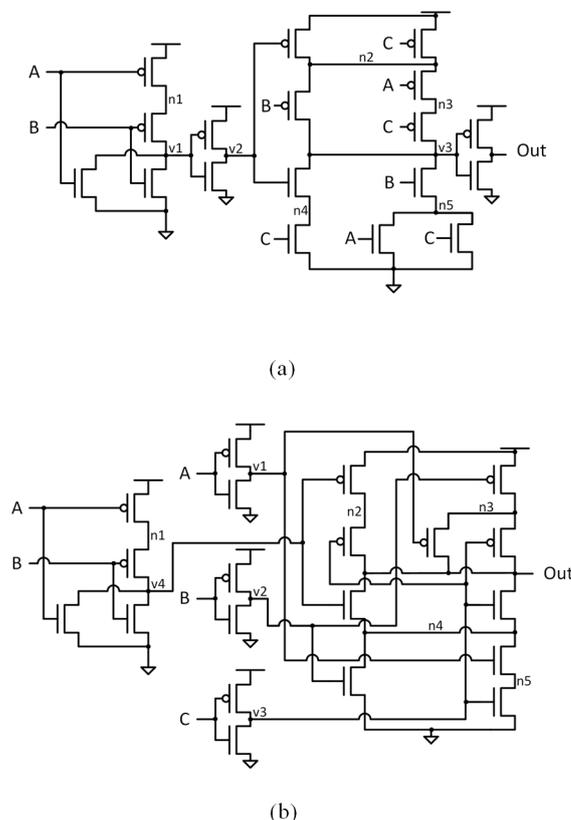
Figura 20 – Esquemático a nível de porta lógica do Votador Bala proposto na literatura.



Fonte: Adaptado de (BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016)

A Figura 21 (a) apresenta a implementação a nível de transistores do votador proposto por (BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016), utilizando 18 transistores conforme os autores declararam que o votador poderia ser construído. Já na Figura 21 (b) explorou-se a realocação do inversor da saída para a entrada.

Figura 21 – Implementações do Votador Bala: (a) votador Bala (OLIVEIRA; SCHVITZ; BUTZEN, 2019) e (b) votador Bala 2.



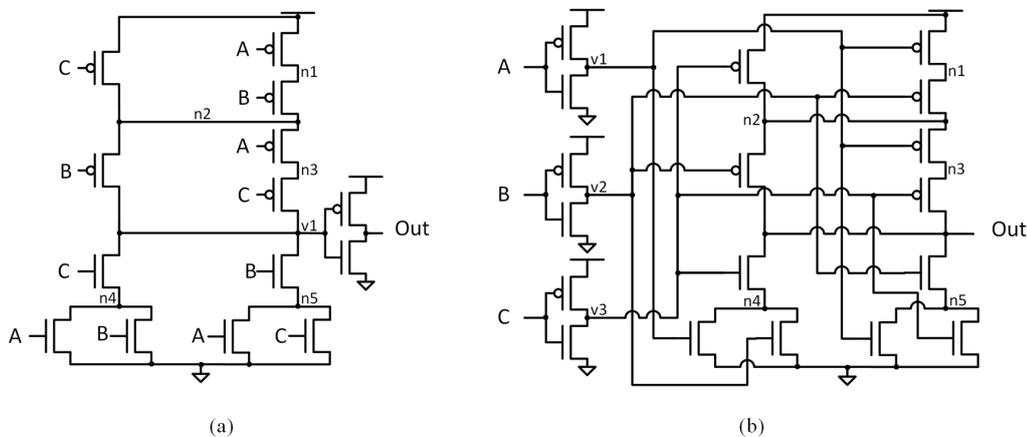
Fonte: O Autor.

3.5 Votador Bala CMOS

A implementação do votador Bala CMOS é apresentada na Figura 22 (a) (OLIVEIRA; SCHVITZ; BUTZEN, 2019). Trata-se de uma implementação *single stage* do votador Bala. O objetivo da utilização dessa topologia é de poder explorar a redução do número de transistores comparada com a versão proposta por (BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016). Além disso, permite explorar novamente a robustez de implementações em estágio único e múltiplos estágios de portas lógicas.

Outra versão do votador é apresentada na Figura 22 (b), o votador Bala CMOS 2. Essa implementação explora a eliminação da porta lógica inversor na saída do circuito. Apesar dessa versão ter uma penalidade em área, em comparação ao votador Bala CMOS, utiliza o mesmo número de transistores necessário para implementação das versões do votador Bala apresentadas na Figura 21.

Figura 22 – Implementações do Votador Bala CMOS: (a) votador Bala CMOS (OLIVEIRA; SCHVITZ; BUTZEN, 2019) e (b) votador Bala CMOS 2.



Fonte: O Autor.

3.6 Conclusões do Capítulo

Conforme explicado anteriormente, o circuito votador desempenha um importante papel dentro das arquiteturas redundantes TMR e ATMR. Visto que a saída da arquitetura e a saída do votador são a mesma, a confiabilidade do circuito votador é de grande importância no projeto. Desta forma o votador majoritário tem sido explorado e analisado na literatura.

A Tabela 3 apresenta um comparativo breve das quatorze implementações apresentadas nesse capítulo. Na tabela são apresentados as seguintes informações, respectivamente, para cada

Tabela 3 – Tabela comparativa das quatorze implementações apresentadas da função majoritária.

Votadores Majoritários	Quant. de transistores	Uso de PLC	Uso de Inversor na saída	Uso de TP	Uso de TP na saída
NANDS	18	-	-	-	-
NORS	18	-	-	-	-
CMOS	12	Sim	Sim	-	-
CMOS 2	16	Sim	-	-	-
Bala	18	Sim	Sim	-	-
Bala 2	18	Sim	-	-	-
Bala CMOS	14	Sim	Sim	-	-
Bala CMOS 2	18	Sim	-	-	-
Ban	18	-	-	Sim	Sim
Ban 2	24	-	Sim	-	-
Ban 3	24	-	-	-	-
Ban 4	12	-	-	Sim	Sim
Ban 5	18	-	Sim	Sim	-
Ban 6	18	-	-	Sim	-

votador majoritário: a quantidade necessária de transistores para sua implementação; a utilização de porta lógica complexa (PLC) em seu projeto; a existência da porta lógica inversor conectada à saída do circuito votador; a utilização de transistor de passagem (TP) na sua implementação; e por fim, se existe transistor de passagem (TP) conectado diretamente à saída do votador. Todas essas informações são importantes características que, conforme será visto nos próximos capítulos, contribuem com a robustez do votador.

4 Metodologia

Conforme visto no Capítulo 1, o objetivo dessa proposta consiste em investigar a robustez à falha SET de diferentes topologias de votadores majoritários considerando aspectos relacionados a técnica de Redundância Modular Tripla (TMR). Para atingir o objetivo principal dessa proposta, o trabalho foi dividido em duas etapas. Portanto, nesse capítulo apresentaremos as metodologias aplicadas a fim de alcançar o objetivo desse trabalho.

4.1 Votadores na arquitetura TMR

Conforme visto anteriormente, o votador majoritário é o ponto crítico da técnica TMR e suas derivadas. O circuito votador desempenha um papel de extrema importância dentro da arquitetura. O sistema TMR como um todo é totalmente dependente do correto funcionamento do votador, visto que a saída do votador e a da arquitetura são a mesma. Portanto é imprescindível estudar a vulnerabilidade do votador majoritário a falhas de radiação para garantir a escolha mais adequada para um projeto.

Nessa etapa foram analisadas quatorze diferentes topologias de votadores majoritários, apresentadas no Capítulo 3, considerando a incidência de uma falha transiente única no circuito votador. Por tratar-se de uma falha única no circuito votador, os módulos da técnica TMR foram considerados isentos de falhas, portanto os únicos vetores de entradas utilizados nessa análise foram $A = B = C$ (000 e 111). A avaliação da robustez dos votadores majoritários no caso da incidência de uma falha transiente única é realizada em duas partes: análise lógica e análise elétrica.

4.1.1 Análise Lógica

Devido a necessidade única de conhecer as áreas ativas compartilhadas pelos transistores, abstraindo informações referentes à tecnologia adotada e regras de espaçamento, para essa avaliação utilizou-se uma simplificação do leiaute: o diagrama *stick*. A partir dos *sticks* das topologias, apresentados no Apêndice A, é possível determinar as regiões ativas sensíveis de cada circuito. Trata-se do conjunto de áreas ativas que não estão conectadas a fontes de alimentação e portanto podem fazer com que a junção PN do transistor esteja reversamente polarizada em algum

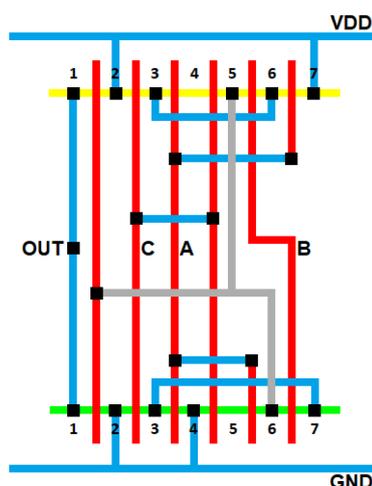
momento da operação (BAUMANN, 2005)(FERLET-CAVROIS; MASSENGILL; GOUKER, 2013).

A falha transiente é inserida em cada área ativa sensível, uma por vez, para cada vetor de entrada e a saída do votador é analisada. Para inserir a falha, simplifica-se seu comportamento, onde se considera que por um espaço de tempo suficiente existe a mudança de sinal no circuito e observa-se se existe a possibilidade desta ser propagada até a saída do votador. Se a falha for propagada, a área ativa é considerada uma área ativa crítica (GILL et al., 2005). Por outro lado, se não for observado um erro na saída do votador, isto é, houve um mascaramento da falha SET, a área ativa sensível é utilizada como base para o cálculo da taxa de mascaramento de falhas (FMR), conforme Equação 4.1.

$$FMR = \frac{\sum TM}{N} \quad (4.1)$$

O cálculo do FMR é a razão entre a soma das áreas ativas sensíveis (TM) pelo número de áreas ativas sensíveis totais do circuito (N). O FMR é calculado para cada vetor de entrada, sendo o valor final do FMR a média entre os valores de FMR dos vetores de entrada analisados. Para exemplificar o cálculo do FMR, a Figura 23 apresenta o diagrama *stick* do Votador CMOS, apresentado no Capítulo 3. Através do diagrama *stick* do circuito, identifica-se as áreas ativas.

Figura 23 – Diagrama *stick* do Votador CMOS



Fonte: Extraído de (OLIVEIRA; SCHVITZ; BUTZEN, 2019).

A Tabela 4 apresenta o resultado da vulnerabilidade do circuito votador CMOS à falha SET para o vetor de entrada "000" cuja a saída esperada é "0" lógico. Para cada plano são apresentadas as áreas ativas sensíveis do circuito, isto é, as áreas ativas que não estão conectadas

diretamente a fontes de alimentação, e a saída do circuito dada a incidência de uma falha SET na área ativa sensível testada. Para realizar o cálculo da taxa de mascaramento de falhas soma-se a quantidade de áreas ativas sensíveis cuja saída permanece inalterada e divide-se pelo total de áreas ativas sensíveis do circuito. No exemplo apresentado na Tabela 4, o FMR do votador CMOS para o vetor de entrada "000" é $FMR = \frac{8}{10} = 0,8$.

Tabela 4 – Resultado da análise da vulnerabilidade à falha SET das áreas ativas sensíveis (AAS) do Votador CMOS para o vetor de entrada "000".

AAS PMOS	Saída	AAS NMOS	Saída
1	X	1	0
3	0	3	0
4	0	5	0
5	0	6	X
6	0	7	0

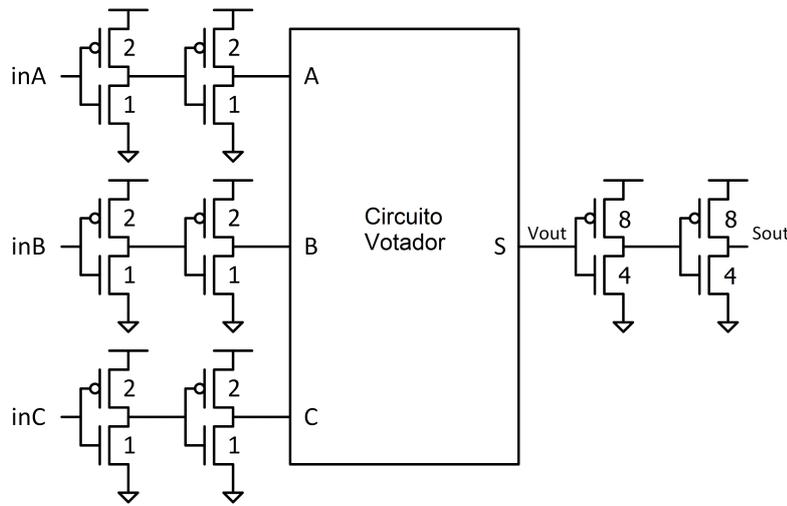
4.1.2 Análise Elétrica

A segunda parte dessa etapa visa complementar os resultados obtidos anteriormente. Simulações elétricas são realizadas utilizando o simulador elétrico NGSPICE para determinar os valores de *Linear Energy Transfer Threshold* (LET_{th}), quantidade de energia mínima transferida necessária para ocorrer uma mudança no estado lógico da saída do votador. Os circuitos votadores foram descritos em linguagem SPICE. Todas as simulações foram realizadas utilizando tecnologia preditiva de 32nm de alto desempenho (ZHAO; CAO, 2006).

Conforme ilustrado na Figura 24, inversores foram inseridos nas entradas e saída do votador majoritário. Os inversores foram inseridos na saída do circuito com largura (W) do transistor 4 vezes maior que os das entradas, a fim de simular a existência de outros circuitos conectados ao bloco (WESTE; HARRIS, 2015). Essa configuração é utilizada para tornar a simulação mais realista. Os transistores PMOS e NMOS foram projetados com valores diferentes de W , sendo W_p o dobro de W_n e W_n o mínimo da tecnologia. Além disso, a métrica para o dimensionamento dos votadores foi realizada utilizando o dimensionamento por esforço lógico, do inglês *Logical Effort* (SUTHERLAND et al., 1999).

Single Event Transient (SET) é uma falha transiente que ocorre quando uma partícula altamente energizada incide numa área ativa criando um caminho condutivo, podendo se propagar pelo circuito atingido através de um pulso transiente (DODD, 1996). O pulso transiente é definido através de uma fonte de corrente transiente definida na forma de uma exponencial

Figura 24 – Metodologia adotada na análise a nível elétrico.



Fonte: Adaptado de (LIEBL, 2016).

dupla (MESSENGER, 1982). Para realizar a simulação elétrica de uma SET, insere-se a fonte de corrente transiente num nodo. Os nodos em que o pulso transiente é aplicado são os nodos referentes as áreas ativas críticas encontradas na análise anterior. A fonte de corrente é definida no simulador elétrico conforme a descrição abaixo (VOGT; HENDRIX; NENZI, 2017).

$$I[\text{nome}] \text{ nodo1 nodo2 EXP (I1 I2 TD1 TAU1 TD2 TAU2)}$$

A declaração da fonte de corrente exponencial dupla mostrada acima requer que alguns parâmetros sejam conhecidos: define-se o nome da fonte de corrente que deseja declarar em *nome*; *nodo1* e *nodo2* são os nodos de conexão da fonte de corrente no circuito afetado; *I1* (*Initial Value*) refere-se ao valor inicial da corrente; *I2* (*Pulsed Value*) refere-se ao valor da amplitude do pulso e é calculado de acordo com a Equação (4.4), derivadas das Equações (4.2) e (4.3) (MESSENGER, 1982); *TD1* (*Rise Delay Time*) trata-se do tempo de atraso de subida do pulso, isto é, refere-se ao instante que a falha será inserida no circuito; *TD2* (*Fall Delay Time*) é o tempo de atraso de descida; *TAU1* (*Rise Time Constant*) e *TAU2* (*Fall Time Constant*) são, respectivamente, a constante de tempo de subida e de descida, e são definidos $TAU1 = T_{\beta}$ e $TAU2 = T_{\alpha}$, que geralmente assumem os valores $T_{\alpha} = 1,64 \times 10^{-10}s$ e $T_{\beta} = 5 \times 10^{-11}s$ (CARRENO; CHOI; IYER, 1990).

A Equação (4.3) refere-se a carga total coletada (Q_{coll}) na região da junção PN, cujo parâmetro L , profundidade da coleção de carga, possui um valor normalmente definido como $2\mu m$ para cada *LET* (*Linear Energy Transfer*), e refere-se a quantidade de energia liberada por

uma partícula por unidade de comprimento do caminho que essa partícula percorreu, igual a $1 \text{ MeV} \cdot \text{cm}^2 / \text{mg}$. A constante $10,8 \text{ fC}$ corresponde a carga depositada pela partícula para cada $1 \mu\text{m}$ (WANG; AGRAWAL, 2008).

$$I2 = \frac{Q_{coll}}{T_{\alpha} - T_{\beta}} \quad (4.2)$$

$$Q_{coll} = 10,8 \times L \times LET \quad (4.3)$$

$$I2 = \frac{10,8 \times L \times LET}{T_{\alpha} - T_{\beta}} \quad (4.4)$$

Os valores de *Linear Energy Transfer Threshold* (LET_{th}) obtidos através da simulação elétrica servem para identificar a robustez de cada nodo dos circuitos votadores. Para identificar os valores de LET_{th} , inicialmente define-se um conjunto de 10 LET s dentro do intervalo de 0,5 a $3 \text{ MeV} \cdot \text{cm}^{-2} / \text{mg}$ com um passo de $0,25 \text{ MeV} \cdot \text{cm}^{-2} / \text{mg}$. Simula-se a incidência da falha SET em cada nodo crítico das topologias, informação obtida na primeira parte dessa etapa, para cada LET definido. A saída do votador majoritário é analisada verificando se o circuito consegue mascarar a falha transiente. Considera-se que houve uma mudança no estado lógico da saída do votador, decorrente da falha SET inserida, quando o pulso transiente que tenha se propagado até a saída atinja 50% do valor da tensão de alimentação.

Então, é escolhido o último LET no qual a saída se mantém inalterada mesmo na presença da falha e o primeiro que apresenta uma mudança no estado lógico da saída. Define-se então um novo intervalo de LET e uma busca binária manual é realizada para encontrar o LET_{th} de cada nodo crítico dos circuitos votadores. Desta forma, para cada topologia, podemos identificar os valores de LET mínimo e máximo, isto é, o menor valor de LET que afeta pelo menos um nodo crítico do circuito e o maior valor de LET , que caso ocorra na topologia um erro será observado na saída independentemente do nodo crítico que incidir.

4.2 Votadores na arquitetura ATMR

Nessa segunda etapa, uma análise da robustez dos votadores majoritários é realizada considerando a arquitetura ATMR. Conforme visto no capítulo anterior, a técnica ATMR utiliza um módulo que implementa a função original (G) e dois módulos que implementam funções

aproximativas (F e H). A escolha das funções aproximativas F e H deve ser feita respeitando a regra $F \subseteq G \subseteq H$.

Essa etapa difere da primeira em relação aos vetores de entrada que serão utilizados. Sabe-se que na técnica ATMR um dos módulos pode divergir dos outros dois em um ou mais mintermos. Portanto é necessário realizar a análise considerando também os vetores de entrada que sejam diferentes de $A = B = C$. Sendo assim, todos os vetores de entrada serão avaliados nas análises lógica e elétrica.

4.2.1 Análise Lógica

Ao cumprir a regra de geração das funções aproximativas implementadas pelos módulos F e H , ilustrada na Fig. 12, teremos que quando G for "0" lógico, F obrigatoriamente será "0". O mesmo ocorre quando G for "1" lógico, H obrigatoriamente será igual a "1". Deste modo é possível concluir que dependendo da ordem dos módulos na arquitetura, certos vetores de entrada jamais ocorrerão no circuito votador.

Primeiramente é necessário definir as possíveis ordens dos módulos de entrada a serem utilizados nessa análise. Trata-se de um arranjo de 3 elementos distintos, os módulos da arquitetura (G , F e H), agrupados de 3 a 3, resultando em 6 possíveis combinações de entrada para o circuito votador: GFH , GHF , FGH , FHG , HGF e HFG . Para cada ordem existem quatro vetores de entrada possíveis, sendo dois quando $G = F = 0$ e os outros dois quando $G = H = 1$. Para uma melhor compreensão considere como exemplo a combinação GFH , essa ordem de módulos resultaria nos possíveis vetores de entrada: "000", "001", "101" e "111".

Definidas as ordens e seus respectivos vetores entradas, é realizada a análise lógica a nível de diagrama *stick*. A metodologia utilizada nessa etapa corresponde a mesma empregada na Subseção 4.1.1. Com os resultados encontrados nessa etapa será possível escolher a melhor opção de circuito votador e será possível investigar qual a melhor opção de ordenamento dos módulos para utilização na técnica ATMR.

4.2.2 Análise Elétrica

Por fim, uma análise a nível elétrico é realizada visando complementar os resultados obtidos na Subseção 4.2.1. Levando em consideração as mesmas informações obtidas a respeito dos vetores de entrada e ordenamento dos módulos, são realizadas simulações elétricas a fim

de simular o comportamento das topologias na presença de uma falha única do tipo SET. A metodologia empregada na análise a nível elétrico é a mesma para ambas as arquiteturas. Todas as informações elétricas utilizadas na Subseção 4.1.2 foram aplicadas nessa análise.

5 Resultados

Neste capítulo serão apresentados os resultados obtidos após a execução da metodologia discutida anteriormente no capítulo 4. Os resultados obtidos no estudo da vulnerabilidade dos votadores nas arquiteturas TMR tradicional e aproximada serão divididos de acordo com a arquitetura analisada. Quatorze topologias de votadores majoritários participaram do estudo. As análises foram realizadas em duas etapas para cada arquitetura. De forma a uma melhor visualização e organização dos resultados, maiores detalhes podem ser visualizados no Apêndice B, onde são apresentados os resultados de todos os vetores de entrada para cada votador majoritário.

Primeiramente são utilizadas as informações de áreas ativas sensíveis extraídas dos diagramas *sticks*. Então, uma falha única do tipo SET é considerada nestas áreas ativas, uma por vez, dos votadores em teste e a saída do mesmo é analisada. Os dados extraídos dessa primeira análise refletem o cenário de pior caso à incidência de uma única falha SET em uma área ativa.

Após, utilizando as informações levantadas na etapa anterior, simulações elétricas são realizadas com o intuito de determinar a energia necessária para que ocorra uma mudança no estado lógico da saída do votador majoritário. Nesta etapa são considerados parâmetros referentes a tecnologia a ser utilizada, diferentes resultados são esperados para diferentes tecnologias. Independentemente da tecnologia adotada, o mais importante nessa etapa é a relação entre os valores do que os valores absolutos obtidos.

5.1 Resultados considerando arquitetura TMR

Os resultados considerando a arquitetura TMR serão apresentados divididos de acordo com as análises realizadas: análise lógica e análise elétrica. Para essa avaliação considerou-se o impacto de uma falha única no circuito votador. Portanto, os módulos da arquitetura TMR tradicional foram considerados isentos de falhas, desta forma os únicos vetores de entrada analisados nessa etapa foram "000" e "111".

5.1.1 Análise Lógica

Os resultados de FMR obtidos nessa primeira análise e informações a respeito das áreas ativas dos circuitos estão presentes na Tabela 5. Essa tabela apresenta, para cada votador

Tabela 5 – Resultados obtidos para as topologias de votadores majoritários considerando a arquitetura TMR tradicional.

Votadores Majoritários	Total de Transistores	Áreas Ativas	Áreas Ativas Sensíveis	Áreas Ativas Críticas	FMR
NANDS	18	21	14	6	0,789
NORS	18	21	14	6	0,789
CMOS	12	14	10	4	0,800
CMOS 2	16	20	14	2	0,929
Bala	18	22	15	4	0,866
Bala 2	18	25	18	3	0,919
Bala CMOS	14	17	12	4	0,833
Bala CMOS 2	18	23	16	2	0,938
Ban	18	24	18	4	0,889
Ban 2	24	28	20	4	0,900
Ban 3	24	28	20	3	0,925
Ban 4	12	16	14	3	0,893
Ban 5	18	23	17	4	0,882
Ban 6	18	23	17	2	0,941

majoritário, considerando ambos vetores de entrada possíveis, os seguintes valores: o número total de transistores utilizados na sua implementação, conforme mostrado nos esquemáticos de transistores dos votadores; o número de áreas ativas totais, informação extraída dos diagramas sticks implementados; o número de áreas ativas sensíveis, que são as áreas ativas que não estão conectadas diretamente a fontes de alimentação; áreas ativas críticas (AAC), que são as áreas ativas sensíveis que na incidência de uma falha, essa, no pior caso, propaga-se até a saída do circuito; e o valor de FMR obtido através da Equação 4.1, que é a taxa de mascaramento das falhas nas áreas ativas sensíveis.

Observando as informações apresentadas na tabela, se considerarmos os valores de FMR, deduz-se que o votador majoritário mais robusto é o Ban 6. Entretanto, observando as áreas ativas críticas, as topologias Bala CMOS 2 e CMOS 2 apresentam o mesmo valor que a topologia Ban 6. A diferença nos valores de FMR se dá pela forma como este é calculado. Um circuito com um maior número de áreas ativas sensíveis terá um FMR maior que um circuito com um número menor, mesmo se ambos tiverem o mesmo número de áreas ativas críticas. Com isso em vista, apenas os valores de FMR, como usado em (BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016), não são o suficiente para identificar qual votador é mais robusto. É necessário analisar todo o conjunto de informações fornecido na tabela, principalmente o número de áreas ativas críticas de cada circuito.

Analisando os três votadores que apresentaram o menor número de áreas ativas críticas,

o votador CMOS 2 possui a menor área, sendo implementado por 16 transistores. Porém, não obtém a menor área dentre todos os votadores analisados. Os votadores CMOS e Ban 4 apresentaram a menor área dentre os demais, sendo implementados por apenas 12 transistores. Ambos votadores são implementados com metade dos transistores necessários para implementar os votadores majoritários de maior área analisados, Ban 2 e Ban 3. Apesar dos votadores Ban 2 e Ban 3 necessitarem do maior número de transistores, os votadores não obtiveram os piores números em relação as áreas ativas críticas.

Os votadores NANDS e NORS são implementados com 33% menos transistores em comparação a Ban 2 e Ban 3, mas apresentam o maior número de áreas ativas críticas, o triplo das AACs observadas nas topologias Ban 6, Bala CMOS 2 e CMOS 2. Portanto, levando em consideração o número de transistores, FMR e áreas ativas críticas, a melhor opção dentre os votadores majoritários analisados seria o votador CMOS 2. Este apresenta o menor número de áreas ativas críticas, possui um dos maiores valores de FMR e, em comparação aos votadores de menor área, possui aproximadamente 33% de aumento de área, o equivalente a utilização de 4 transistores a mais em sua implementação.

5.1.2 Análise Elétrica

Os resultados sintetizados de LET_{th} obtidos nas simulações elétricas realizadas na análise elétrica considerando a arquitetura TMR tradicional, podem ser observados na Tabela 6. O valor de $LET_{threshold}$ corresponde à quantidade de energia necessária para que ocorra uma mudança no estado lógico da saída do circuito sob teste. A Tabela 6 apresenta para cada vetor de entrada os valores de LET_{th} mínimo, menor valor obtido, e máximo, maior valor obtido, referentes às áreas ativas críticas de cada votador majoritário para a tecnologia de 32nm. O valor de LET_{th} mínimo refere-se ao menor valor de LET_{th} que afeta pelo menos um nodo crítico do circuito analisado. O valor de LET_{th} máximo refere-se ao maior valor de LET_{th} obtido. Caso esse LET ocorra na topologia, independente do nodo crítico que incidir, um erro será observado na saída do circuito. Em destaque na tabela estão: os maiores valores de LET_{th} , em verde; e os menores valores de LET_{th} , em amarelo.

Dentre os valores destacados em verde, o votador NORS apresenta o maior valor de LET_{th} tanto máximo quanto mínimo, considerando o vetor de entrada "000". Porém, esse resultado não é repetido no vetor "111". O mesmo comportamento é observado no votador NANDS, porém de forma complementar em relação aos vetores de entrada. Isso se deve à

Tabela 6 – Simplificação da tabela de LET_{th} ($MeV.cm^{-2}/mg$) para arquitetura TMR considerando a tecnologia de 32nm.

Votadores	Vetor de Entrada "000"		Vetor de Entrada "111"	
	Min LET_{th}	Max LET_{th}	Min LET_{th}	Max LET_{th}
NANDS	0,78	1,78	2,44	2,44
NORS	1,95	1,95	0,77	1,40
CMOS	0,68	1,75	0,83	1,72
CMOS 2	1,70	1,70	1,70	1,70
Bala	0,68	1,75	0,83	1,72
Bala 2	1,64	1,64	1,77	1,77
Bala CMOS	0,68	1,95	0,83	1,97
Bala CMOS 2	1,94	1,94	1,91	1,91
Ban	0,39	0,97	0,41	1,22
Ban 2	0,68	0,81	0,76	0,83
Ban 3	0,77	0,77	0,69	0,80
Ban 4	0,39	0,39	0,41	1,72
Ban 5	0,68	0,81	0,76	0,83
Ban 6	0,77	0,77	0,81	0,81

complementariedade das portas lógicas NOR e NAND que são utilizadas na implementação desses votadores.

Em relação aos valores destacados em amarelo, o votador Ban 4 apresenta o menor valor de LET_{th} tanto máximo quanto mínimo, considerando o vetor de entrada "000", repetindo esse resultado em relação ao LET_{th} mínimo para o vetor "111". O Ban 4 é o circuito votador, dentre os 14 analisados, que utiliza o maior número de transistores de passagem em sua implementação. Analisando os resultados, todos os valores abaixo de $0,68 MeV.cm^{-2}/mg$ estão diretamente relacionados a utilização de transistores de passagem nas saídas das implementações dos votadores majoritários.

Observando os números de LET_{th} , máximo e mínimo, apenas os votadores CMOS 2, Bala 2 e Bala CMOS 2 apresentam esses valores acima de $1,00 MeV.cm^{-2}/mg$ em ambos vetores de entrada, conforme destaca-se em vermelho na Tabela 6. Considerando o vetor "000", ou o vetor "111", esses votadores não apresentam diferença entre seus valores de LET . Isso ocorre porque essas topologias apresentam apenas um único nodo crítico por vetor de entrada, resultando em valores de LET_{th} máximo e mínimo iguais. Dentre os três votadores mencionados, somente a topologia CMOS 2 não apresenta diferença dos valores, máximo e mínimo, entre os vetores de entrada analisados.

Considerando todas as informações levantadas independente de tecnologia, e conside-

rando o cenário de pior caso, na Tabela 5 e os resultados de LET_{th} disponibilizados na Tabela 6, os votadores CMOS 2 e Bala CMOS 2 foram os circuitos que apresentaram bons resultados nos quesitos apontados. Desta forma, é possível afirmar que ambas topologias seriam boas escolhas em termos de robustez, além de suportarem LET_{sth} aproximadamente 5x maior para o vetor "000" e 4,5x maior para o vetor "111", no caso do votador Bala CMOS 2, e LET_{th} aproximadamente 4x maior para ambos vetores no caso do votador CMOS 2.

5.2 Resultados considerando arquitetura ATMR

Os resultados considerando a técnica ATMR serão apresentados também divididos de acordo com a análise realizada: análise lógica e análise elétrica. Conforme pontuado anteriormente no Capítulo 4, a técnica TMR aproximada difere-se da TMR tradicional pela utilização de dois módulos (F e H) que implementam funções aproximadas com o intuito de obter um bom *trade-off* entre área e cobertura de falhas. Sabe-se que um dos módulos aproximados poderá divergir em um determinado momento do módulo que implementa a função original (G) e do segundo módulo aproximado. Portanto, mesmo considerando o impacto de uma falha única no circuito votador, é necessário analisar todas as possibilidades de vetores de entrada do circuito.

A arquitetura ATMR apresenta a regra $F \subseteq G \subseteq H$ para a geração das funções aproximadas F e H . Ao cumprir essa regra, ilustrada na Fig. 12, teremos que quando G for "0" lógico, F obrigatoriamente será "0", assim como quando G for "1" lógico, H obrigatoriamente também será igual a "1". Dependendo da ordem dos módulos na arquitetura, certos vetores de entrada jamais ocorrerão no circuito votador. Portanto, o impacto da ordem dos módulos na robustez dos votadores majoritários é analisado.

5.2.1 Análise Lógica

A fim de demonstrar o impacto da ordem dos módulos na escolha do circuito votador, foram escolhidos dois votadores majoritários de acordo com seus resultados nessa análise. Um com o menor valor de FMR, o votador NANDS, e o outro com o maior valor, o votador Ban 6. A Tabela 7 apresenta a quantidade de áreas ativas críticas obtidas pelo votador NANDS para todas as possíveis combinações de ordem dos módulos da arquitetura ATMR.

Cada ordem é composta por quatro possibilidades de vetores de entrada, onde dois refletem os módulos sem discordância entre si e outros dois vetores onde existe uma divergência,

Tabela 7 – Quantidade de áreas ativas críticas obtidas pelo votador NANDS.

Vetor de Entrada	GFH	GHF	FGH	FHG	HGF	HFG
000	5	5	5	5	5	5
001	5	-	5	-	-	-
010	-	6	-	6	-	-
011	-	-	4	4	-	-
100	-	-	-	-	7	7
101	3	-	-	-	-	3
110	-	2	-	-	2	-
111	1	1	1	1	1	1
Total	14	14	15	16	15	16

porém respeitando a regra da arquitetura ATMR. Os vetores de entrada que não ocorrem na ordem indicada pela coluna estão sinalizados por um traço simples. Ao final da tabela é possível observar o somatório do número de áreas ativas críticas presentes em cada ordem.

Observando os valores finais das áreas ativas críticas para as 6 combinações de ordem dos módulos, podemos verificar que existe diferença de uma ordem para outra. No caso do votador NANDS, a diferença existente entre o número de AACs é relativamente pequeno, não ultrapassando 2 áreas ativas entre o maior e o menor valor.

Entretanto, se analisarmos a Tabela 8, referente ao votador Ban 6, existe uma diferença significativa na quantidade de áreas ativas entre as ordens, chegando a 75% de redução. Visto que pode existir diferenças significativas de redução de áreas ativas críticas no circuito votador na utilização de uma determinada ordem, é importante analisar o impacto de todas as possibilidades de ordens de módulos para cada votador a fim de determinar as melhores escolhas para cada circuito.

Tabela 8 – Quantidade de áreas ativas críticas obtidas pelo votador Ban 6.

Vetor de Entrada	GFH	GHF	FGH	FHG	HGF	HFG
000	1	1	1	1	1	1
001	6	-	6	-	-	-
010	-	4	-	4	-	-
011	-	-	2	2	-	-
100	-	-	-	-	2	2
101	6	-	-	-	-	6
110	-	4	-	-	4	-
111	1	1	1	1	1	1
Total	14	10	10	8	8	10

A Tabela 9 apresenta as melhores e as piores opções de ordem para cada votador, assim

como os resultados sintetizados de áreas ativas críticas obtidos. Para cada opção de ordem, essas são apresentadas juntamente com a quantidade de áreas ativas críticas além da informação do valor de FMR. Uma comparação entre as melhores e as piores ordens é realizada com o intuito de demonstrar o impacto que a escolha da ordem tem na robustez de um votador majoritário, conforme é possível observar na coluna que apresenta a redução das AACs.

Observando os valores obtidos, os votadores CMOS, CMOS 2 e Ban 6 apresentam na sua melhor ordem, a menor quantidade de AACs, sendo implementados respectivamente por 12, 16 e 18 transistores. Os votadores majoritários que obtêm a maior quantidade de áreas ativas críticas, NANDS e NORS, são os votadores com menor impacto, em relação a pior ordem, em comparação aos outros votadores em estudo. A topologia Ban 6 além da menor quantidade de AACs na melhor ordem também apresenta a maior porcentagem de redução de áreas ativas críticas comparado com a pior ordem em relação aos demais votadores analisados. Portanto, a escolha da ordem dos módulos ao utilizar o votador Ban 6 pode impactar significativamente na suscetibilidade a falhas, na pior ordem aumentando em 6 o número de áreas ativas críticas.

Considerando todas as informações apresentadas na Tabela 9, assim como o número de transistores utilizados em cada implementação, o votador CMOS se destaca. A topologia é uma das implementadas com a menor quantidade de transistores em comparação aos demais votadores analisados, além de obter o menor número de áreas ativas críticas em sua melhor ordem.

Tabela 9 – Resultados sintetizados de áreas ativas críticas (AAC) classificados de acordo com a melhor ou pior ordem dos módulos para cada votador majoritário.

Votadores Majoritários	Melhor(es) Ordem(ns)			Pior(es) Ordem(ns)			Red. AAC
	Ordem	AAC	FMR	Ordem	AAC	FMR	
NANDS	GFH, GHF	14	0,750	FHG, HFG	16	0,722	14,3%
NORS	FHG, HFG	14	0,750	GFH, GHF	16	0,722	14,3%
CMOS	FGH	8	0,800	GHF, HFG	11	0,725	37,5%
CMOS 2	HGF	8	0,857	GFH, FHG	11	0,804	37,5%
Bala	FGH, HGF	10	0,833	GFH	15	0,752	50,0%
Bala 2	FHG	9	0,875	GFH, HGF	14	0,806	55,6%
Bala CMOS	HGF	9	0,813	GFH	13	0,729	44,4%
Bala CMOS 2	FGH	9	0,859	GHF	13	0,797	44,4%
Ban	HGF	11	0,847	GFH	17	0,764	54,5%
Ban 2	HGF, HFG	12	0,850	GFH, GHF	17	0,788	41,7%
Ban 3	GHF	10	0,875	FGH	15	0,788	50,0%
Ban 4	FGH, FHG, HGF	10	0,821	GFH	14	0,750	40,0%
Ban 5	FGH, FHG, HGF, HFG	12	0,825	GFH, GHF	16	0,764	33,3%
Ban 6	FHG, HGF	8	0,882	GFH	14	0,795	75,0%

5.2.2 Análise Elétrica

A Tabela 10 apresenta os valores de LET_{th} máximo e mínimo de todas as possíveis ordens de módulos da arquitetura ATMR para cada circuito votador. Em relação a comparação entre os votadores para cada ordem, dada as condições adotadas, como o dimensionamento, existe uma diferença significativa dos valores de LET_{th} , até duas vezes no mínimo e até três vezes no máximo.

Comparando o ordenamento dos módulos para cada votador, a diferença dos valores de LET_{th} mínimo é inexistente ou praticamente inexistente. A diferença dos valores de LET_{th} máximo, exceto nas topologias Ban e Ban 5, apresentam o mesmo comportamento do LET_{th} mínimo. O votador majoritário Ban obteve uma diferença do LET_{th} máximo no ordenamento de quase 40%. Já o circuito votador Ban 5 apresentou quase o dobro de diferença entre os valores de LET_{th} máximo.

Tabela 10 – LET_{th} ($MeV.cm^{-2}/mg$) máximo e mínimo de todas as possíveis ordens de módulos para arquitetura ATMR considerando a tecnologia de 32nm.

Votadores	GFH		GHF		FGH		FHG		HGF		HFG	
	Min	Max										
Majoritários												
NANDS	0,74	2,44	0,74	2,44	0,74	2,44	0,74	2,44	0,74	2,44	0,74	2,44
NORS	0,69	1,95	0,69	1,95	0,69	1,95	0,69	1,95	0,69	1,95	0,69	1,95
CMOS	0,68	1,75	0,68	1,75	0,68	1,75	0,68	1,75	0,68	1,75	0,68	1,75
CMOS 2	0,71	1,70	0,68	1,70	0,69	1,70	0,69	1,70	0,68	1,70	0,71	1,70
Bala	0,68	1,75	0,68	1,75	0,68	1,75	0,68	1,75	0,68	1,75	0,68	1,75
Bala 2	0,70	1,77	0,69	1,77	0,73	1,77	0,73	1,77	0,69	1,77	0,70	1,77
Bala CMOS	0,68	1,97	0,68	1,97	0,68	1,97	0,68	1,97	0,68	1,97	0,68	1,97
Bala CMOS 2	0,69	1,94	0,69	1,94	0,72	1,94	0,72	1,94	0,69	1,94	0,69	1,94
Ban	0,39	1,22	0,39	1,68	0,39	1,22	0,39	1,22	0,39	1,68	0,39	1,22
Ban 2	0,68	0,90	0,68	0,88	0,68	0,90	0,68	0,83	0,68	0,88	0,68	0,83
Ban 3	0,69	0,89	0,69	0,88	0,68	0,89	0,68	0,88	0,69	0,87	0,69	0,87
Ban 4	0,37	1,72	0,38	1,72	0,37	1,72	0,39	1,72	0,38	1,72	0,37	1,72
Ban 5	0,39	1,60	0,38	0,88	0,39	0,83	0,40	0,83	0,38	0,88	0,42	1,60
Ban 6	0,39	0,92	0,39	0,87	0,39	0,90	0,42	0,87	0,39	0,88	0,41	0,92

Os resultados sintetizados da Tabela 10 podem ser observados na Tabela 11. Nessa tabela são apresentados, para cada topologia, o número de áreas ativas críticas e os valores de LET_{th} , mínimo e máximo, da melhor ordem e da pior ordem, respectivamente. Destacados em azul, estão os maiores valores tanto do LET_{th} máximo quanto LET_{th} mínimo, para ambas ordens. Esses valores além de serem iguais entre as ordens, são observados no mesmo circuito votador, o NANDS.

Destacados em vermelho, estão os menores valores de LET_{th} para as duas ordens. Os votadores majoritários Ban, Ban 4, Ban 5 e Ban 6 obtiveram, respectivamente, os piores valores de LET_{th} mínimo para ambas as ordens de módulos. Os quatro circuitos são implementados por transistores de passagem. Em relação ao LET_{th} máximo, os votadores que apresentaram o menor valor, para ambas as ordens, foram Ban 2 e Ban 5, e o votador que apresentou o maior desvio padrão entre as ordens foi o circuito votador Ban.

Observando os valores de LET_{th} entre as ordens, é possível perceber que a ordem, com raras exceções, não muda muito a questão dos valores de LET . Os votadores CMOS, CMOS 2 e Ban 6 apesar de apresentarem a menor quantidade de áreas ativas, não obtiveram os melhores resultados de LET_{th} , principalmente o votador Ban 6, cujos baixos valores de LET_{th} está diretamente relacionado a utilização de transistores de passagem em sua implementação.

Por outro lado, os votadores Bala 2, Bala CMOS e Bala CMOS 2 apresentaram resultados um pouco maiores com a penalidade de uma área ativa crítica a mais. Dentre esses três votadores, o Bala CMOS é implementado com quatro transistores a menos, num total de 14 transistores, sendo uma boa opção ao levar em consideração os valores de LET . Já o votador CMOS é um dos menores votadores implementados e apresenta uma área ativa a menos, porém é um pouco menos robusto em relação aos $LETs$ do que o votador Bala CMOS.

Tabela 11 – Resultados sintetizados de LET_{th} ($MeV.cm^{-2}/mg$) considerando a tecnologia de $32nm$.

Votadores	Melhor Ordem			Pior Ordem		
	Majoritários	AAC	Min LET_{th}	Max LET_{th}	AAC	Min LET_{th}
NANDS	14	0,74	2,44	16	0,74	2,44
NORS	14	0,69	1,95	16	0,69	1,95
CMOS	8	0,68	1,75	11	0,68	1,75
CMOS 2	8	0,68	1,70	11	0,69	1,70
Bala	10	0,68	1,75	15	0,68	1,75
Bala 2	9	0,73	1,77	14	0,69	1,77
Bala CMOS	9	0,68	1,97	13	0,68	1,97
Bala CMOS 2	9	0,72	1,94	13	0,69	1,90
Ban	11	0,39	1,68	17	0,39	1,22
Ban 2	12	0,68	0,83	17	0,68	0,88
Ban 3	10	0,69	0,88	15	0,68	0,89
Ban 4	10	0,37	1,72	14	0,37	1,72
Ban 5	12	0,38	0,83	16	0,38	0,88
Ban 6	8	0,39	0,87	14	0,39	0,92

6 Considerações Finais

A dissertação apresentada visa investigar a robustez à falha SET de diferentes topologias de votadores majoritários considerando aspectos relacionados a técnica de Redundância Modular Tripla. A principal contribuição dessa dissertação é prover um estudo detalhado da robustez do circuito votador. Para isso foram selecionadas e modificadas diferentes implementações da função majoritária propostas na literatura (BAN; NAVINER, 2010)(BALASUBRAMANIAN; PRASAD; MASTORAKIS, 2016)(OLIVEIRA; SCHVITZ; BUTZEN, 2019), resultando num conjunto de catorze circuitos votadores selecionados.

A metodologia proposta foi dividida em 2 etapas e apresentada no Capítulo 4. Cada etapa corresponde à arquitetura redundante a qual os circuitos votadores foram inseridos para análise. Além disso, as etapas foram divididas em duas partes cada de acordo com a análise realizada: a nível de leiaute utilizando diagramas *sticks*, independente de tecnologia, e a nível elétrico, utilizando o modelo preditivo de 32nm. Resultados parciais obtidos durante o desenvolvimento da primeira etapa foram publicados em (OLIVEIRA; SCHVITZ; BUTZEN, 2018) (OLIVEIRA; SCHVITZ; BUTZEN, 2019).

A complexidade da análise aumenta na segunda etapa ao considerarmos o votador majoritário presente na arquitetura ATMR, visto que a ordem dos módulos da arquitetura interfere nos vetores de entrada a serem analisados no circuito votador. Desta forma este trabalho contribui apresentando uma análise detalhada dos circuitos ao considerarmos a arquitetura ATMR, demonstrando que a ordem dos módulos impacta na robustez das topologias dos circuitos votadores. Com isso, um importante ponto é abordado neste trabalho de forma a discutir as melhores opções de votadores majoritários não só na arquitetura TMR, mas também nas arquiteturas aproximativas, as quais nem sempre a melhor opção de votador para TMR será a mesma opção para uma arquitetura ATMR. Este ponto abordado nesta dissertação, de acordo com o conhecimento dos autores durante a escrita deste texto, é o primeiro trabalho que busca otimizar a escolha do votador para arquiteturas aproximativas provendo um estudo dos vetores de entrada e seu impacto na robustez das implementações obtidas para a função majoritária.

Várias das abordagens exploradas nessa dissertação podem ser aplicadas em outros circuitos combinacionais. Um potencial exemplo é a metodologia de estimativa de susceptibili-

dade proposta nessa dissertação para identificar as regiões ativas críticas de um circuito votador e estimar sua robustez a uma falha única SET considerando o pior caso. Isto é, no caso de não existir um mascaramento elétrico. Essa metodologia pode ser aplicada a qualquer circuito combinacional, como por exemplo em circuitos somadores. Outro ponto levantado e analisado nesse trabalho que pode ser aplicado a outros circuitos combinacionais é evitar a utilização de inversores nas saídas dos circuitos. Além disso, pode-se explorar o dimensionamento dos transistores que compartilham áreas ativas críticas para aumentar os valores de *LETs* suportados.

6.1 Produção Científica

1. OLIVEIRA, I. F. V.; SCHVITZ, R. B. ; BUTZEN, P. F., **Fault masking ratio analysis of majority voters topologies**, IEEE Latin-American Test Symposium (LATS), São Paulo, março de 2018.
2. OLIVEIRA, I. F. V.; SCHVITZ, R. B. ; BUTZEN, P. F., **Robustez de circuitos votadores à radiação**, Mostra de Produção Universitária (MPU), FURG, Rio Grande, outubro de 2018.
3. OLIVEIRA, I. F. V.; SCHVITZ, R. B. ; BUTZEN, P. F., **Single event transient sensitivity analysis of different 32nm CMOS majority voters designs**, Microelectronics Reliability, setembro de 2019.
4. OLIVEIRA, I. F. V., **Single event transient analysis of different majority voters designs considering approximate TMR**, Prêmio LATS 2020 Best Master Thesis Award (LATS), Jatiúca (Maceió), março de 2020.

Referências

- AGUIAR, Y. Q. et al. Analysis of the charge sharing effect in the set sensitivity of bulk 45 nm standard cell layouts under heavy ions. *Microelectronics Reliability*, Elsevier, v. 88, p. 920–924, 2018. Citado na página 17.
- AUTRAN, J.-L. et al. Particle monte carlo modeling of single-event transient current and charge collection in integrated circuits. *Microelectronics Reliability*, Elsevier, v. 54, n. 9-10, p. 2278–2283, 2014. Citado 2 vezes nas páginas 17 e 29.
- AZAMBUJA, J. R. F. d. *Designing and evaluating hybrid techniques to detect transient faults in processors embedded in FPGAs*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, 2013. Citado 2 vezes nas páginas 31 e 32.
- BALASUBRAMANIAN, P.; PRASAD, K.; MASTORAKIS, N. E. A fault tolerance improved majority voter for tmr system architectures. *arXiv preprint arXiv:1605.03771*, 2016. Citado 8 vezes nas páginas 18, 19, 33, 43, 44, 45, 56 e 65.
- BAN, T.; NAVINER, L. A. de B. A simple fault-tolerant digital voter circuit in tmr nanoarchitectures. In: IEEE. *NEWCAS Conference (NEWCAS), 2010 8th IEEE International*. [S.l.], 2010. p. 269–272. Citado 5 vezes nas páginas 18, 19, 32, 41 e 65.
- BAUMANN, R. C. Soft errors in commercial integrated circuits. *International Journal of High Speed Electronics and Systems*, World Scientific, v. 14, n. 02, p. 299–309, 2004. Citado 2 vezes nas páginas 17 e 30.
- BAUMANN, R. C. Radiation-induced soft errors in advanced semiconductor technologies. *IEEE Transactions on Device and materials reliability*, IEEE, v. 5, n. 3, p. 305–316, 2005. Citado 3 vezes nas páginas 17, 29 e 48.
- BERTACCO, V. et al. Decision diagrams and pass transistor logic synthesis. In: *Int’l Workshop on Logic Synth.* [S.l.: s.n.], 1997. v. 168. Citado na página 26.
- BIROLINI, A. *Quality and reliability of technical systems: theory, practice, management*. [S.l.]: Springer Science & Business Media, 2012. Citado na página 28.
- BUTZEN, P. F. et al. Efeitos físicos nanométricos em circuitos inte-grados digitais. Citado na página 25.
- CARRENO, V. A.; CHOI, G.; IYER, R. Analog-digital simulation of transient-induced logic errors and upset susceptibility of an advanced control system. 1990. Citado na página 50.
- DODD, P. et al. Current and future challenges in radiation effects on cmos electronics. *IEEE Transactions on Nuclear Science*, IEEE, v. 57, n. 4, p. 1747–1763, 2010. Citado na página 29.
- DODD, P. E. Device simulation of charge collection and single-event upset. *IEEE Transactions on Nuclear Science*, IEEE, v. 43, n. 2, p. 561–575, 1996. Citado na página 49.
- DODD, P. E. Physics-based simulation of single-event effects. *IEEE Transactions on Device and Materials Reliability*, IEEE, v. 5, n. 3, p. 343–357, 2005. Citado na página 29.

- DODD, P. E.; MASSENGILL, L. W. Basic mechanisms and modeling of single-event upset in digital microelectronics. *IEEE Transactions on nuclear Science*, IEEE, v. 50, n. 3, p. 583–602, 2003. Citado 2 vezes nas páginas 17 e 29.
- FERLET-CAVROIS, V.; MASSENGILL, L. W.; GOUKER, P. Single event transients in digital cmos—a review. *IEEE Transactions on Nuclear Science*, IEEE, v. 60, n. 3, p. 1767–1790, 2013. Citado 3 vezes nas páginas 17, 29 e 48.
- FRANCO, D. T.; NAVINER, J.-F.; NAVINER, L. Yield and reliability issues in nanoelectronic technologies. *Annals of Telecommunications*, Springer, v. 61, n. 11, p. 1422–1457, 2006. Citado na página 28.
- FRIEDBERG, W.; COPELAND, K. *Ionizing radiation in Earth's atmosphere and in space near earth*. [S.l.], 2011. Citado 2 vezes nas páginas 17 e 29.
- GAILLARD, R. Single event effects: Mechanisms and classification. In: *Soft errors in modern electronic systems*. [S.l.]: Springer, 2011. p. 27–54. Citado na página 29.
- GILL, B. S. et al. Node sensitivity analysis for soft errors in cmos logic. In: IEEE. *IEEE International Conference on Test, 2005*. [S.l.], 2005. p. 9–pp. Citado na página 48.
- GOMES, I. A.; KASTENSMIDT, F. G. Reducing tmr overhead by combining approximate circuit, transistor topology and input permutation approaches. In: IEEE. *2013 26th Symposium on Integrated Circuits and Systems Design (SBCCI)*. [S.l.], 2013. p. 1–6. Citado 2 vezes nas páginas 32 e 34.
- GOMES, I. A. C. *Use of approximate triple modular redundancy for fault tolerance in digital circuits*. Tese (Doutorado), 2018. Citado 2 vezes nas páginas 18 e 32.
- HARBOE-SORENSEN, R. et al. The seu risk assessment of z80a, 8086 and 80c86 microprocessors intended for use in a low altitude polar orbit. *IEEE Transactions on Nuclear Science*, IEEE, v. 33, n. 6, p. 1626–1631, 1986. Citado na página 29.
- HAZUCHA, P.; SVENSSON, C. Impact of cmos technology scaling on the atmospheric neutron soft error rate. *IEEE Transactions on Nuclear science*, IEEE, v. 47, n. 6, p. 2586–2594, 2000. Citado na página 28.
- HIARI, O.; SADEH, W.; RAWASHDEH, O. Towards single-chip diversity tmr for automotive applications. In: IEEE. *2012 IEEE International Conference on Electro/Information Technology*. [S.l.], 2012. p. 1–6. Citado na página 32.
- KOGA, R. et al. Techniques of microprocessor testing and seu-rate prediction. *IEEE Transactions on Nuclear Science*, IEEE, v. 32, n. 6, p. 4219–4224, 1985. Citado na página 29.
- KOREN, I.; KRISHNA, C. M. *Fault-tolerant systems*. [S.l.]: Morgan Kaufmann, 2010. Citado 2 vezes nas páginas 17 e 28.
- KSHIRSAGAR, R. V.; PATRIKAR, R. M. Design of a novel fault-tolerant voter circuit for tmr implementation to improve reliability in digital circuits. *Microelectronics Reliability*, Elsevier, v. 49, n. 12, p. 1573–1577, 2009. Citado 5 vezes nas páginas 18, 32, 39, 41 e 43.
- LAI, Y.-T.; JIANG, Y.-C.; CHU, H.-M. Bdd decomposition for mixed cmos/ptl logic circuit synthesis. In: IEEE. *2005 IEEE International Symposium on Circuits and Systems*. [S.l.], 2005. p. 5649–5652. Citado na página 25.

- LIEBL, E. *Avaliação da robustez de diferentes topologias de circuitos votadores*. Dissertação (Mestrado), 2016. Citado 5 vezes nas páginas 9, 33, 40, 42 e 50.
- MALVINO, A. P. *Eletrônica: Vol. 1. Makron, São Paulo*, 1997. Citado na página 23.
- MATSUMOTO, K.; UEHARA, M.; MORI, H. Evaluations of resettable stateful nmr. In: IEEE. *2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications*. [S.l.], 2011. p. 740–745. Citado na página 32.
- MESSENGER, G. Collection of charge on junction nodes from ion tracks. *IEEE Transactions on Nuclear Science*, IEEE, v. 29, n. 6, p. 2024–2031, 1982. Citado na página 50.
- NEUMANN, J. V. Probabilistic logics and the synthesis of reliable organisms from unreliable components. *Automata studies*, v. 34, p. 43–98, 1956. Citado 2 vezes nas páginas 18 e 32.
- OLIVEIRA, I. F.; SCHVITZ, R. B.; BUTZEN, P. F. Fault masking ratio analysis of majority voters topologies. In: IEEE. *2018 IEEE 19th Latin-American Test Symposium (LATS)*. [S.l.], 2018. p. 1–6. Citado 6 vezes nas páginas 9, 18, 33, 40, 42 e 65.
- OLIVEIRA, I. F. V.; SCHVITZ, R. B.; BUTZEN, P. F. Single event transient sensitivity analysis of different 32nm cmos majority voters designs. *Microelectronics Reliability*, Elsevier, 2019. Citado 15 vezes nas páginas 9, 10, 18, 19, 33, 40, 41, 43, 44, 45, 48, 65, 73, 74 e 75.
- SADEGHI, M.; SOLTAN, H.; KHAYYAMBASHI, M. The study of hardware redundancy techniques to provide a fault tolerant system. *Cumhuriyet Science Journal*, v. 36, n. 4, p. 236–245, 2015. Citado na página 33.
- SANCHEZ-CLEMENTE, A. et al. Logic masking for set mitigation using approximate logic circuits. In: IEEE. *2012 IEEE 18th International On-Line Testing Symposium (IOLTS)*. [S.l.], 2012. p. 176–181. Citado na página 32.
- SANCHEZ-CLEMENTE, A. J. et al. Error mitigation using approximate logic circuits: A comparison of probabilistic and evolutionary approaches. *IEEE Transactions on Reliability*, IEEE, v. 65, n. 4, p. 1871–1883, 2016. Citado 2 vezes nas páginas 18 e 34.
- SCHVITZ, R. et al. A simplified layout-level method for single event transient faults susceptibility on logic gates. In: IEEE. *2019 IFIP/IEEE 27th International Conference on Very Large Scale Integration (VLSI-SoC)*. [S.l.], 2019. p. 185–190. Citado 2 vezes nas páginas 17 e 29.
- SCHVITZ, R. B. *Análise da suscetibilidade de portas lógicas na presença de falhas de efeitos singulares*. Tese (Doutorado) — Universidade Federal de Pelotas, 2020. Citado 3 vezes nas páginas 9, 27 e 29.
- SEXTON, F. W. Destructive single-event effects in semiconductor devices and ics. *IEEE Transactions on Nuclear Science*, IEEE, v. 50, n. 3, p. 603–621, 2003. Citado na página 29.
- SHELAR, R. S. *Synthesis for nanometer technologies*. Tese (Doutorado) — University of Minnesota, 2004. Citado na página 26.
- SIERAWSKI, B. D.; BHUVA, B. L.; MASSENGILL, L. W. Reducing soft error rate in logic circuits through approximate logic functions. *IEEE transactions on nuclear science*, IEEE, v. 53, n. 6, p. 3417–3421, 2006. Citado 3 vezes nas páginas 18, 32 e 34.

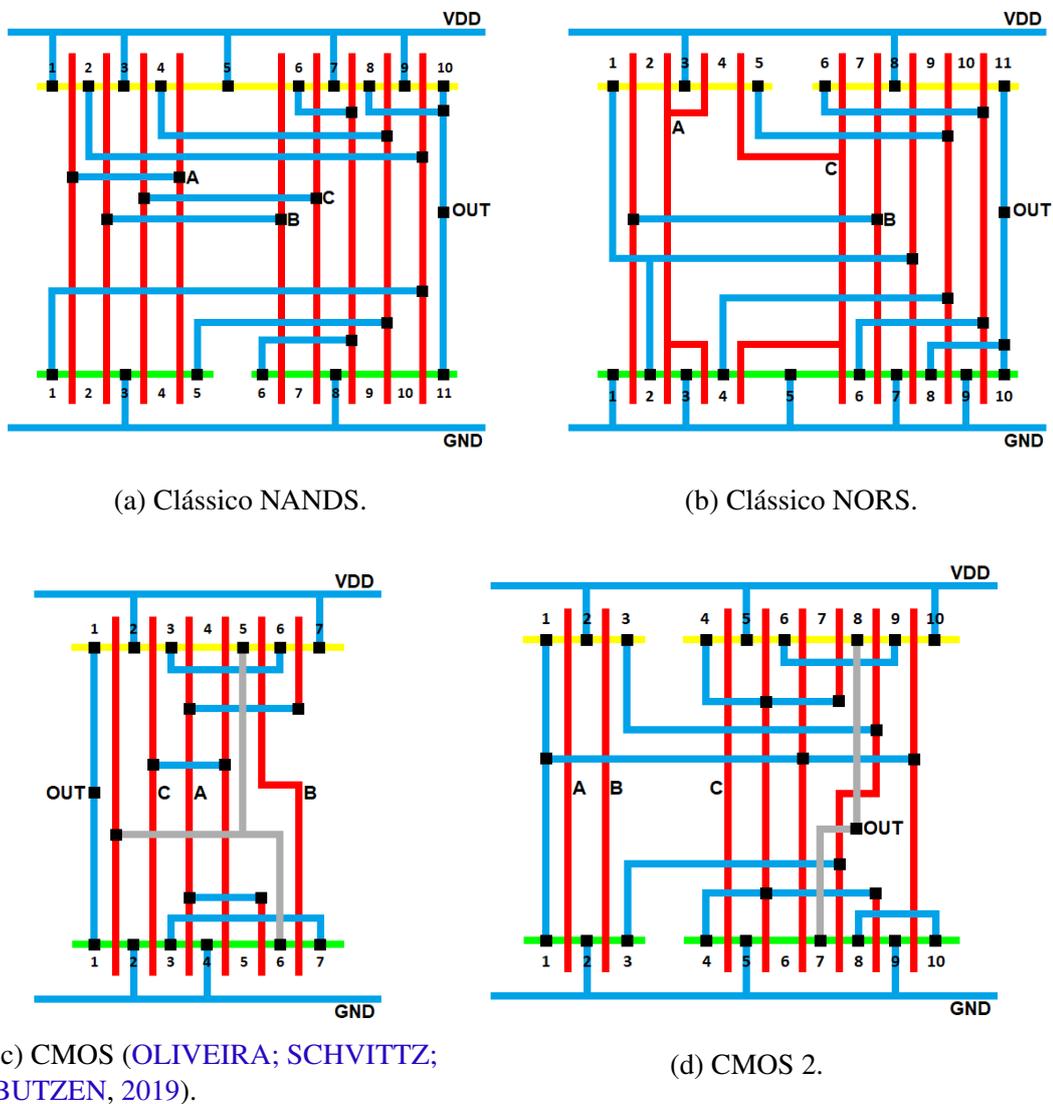
- SUTHERLAND, I. et al. *Logical effort: designing fast CMOS circuits*. [S.l.]: Morgan Kaufmann, 1999. Citado na página 49.
- TSIVIDIS, Y. *Operation and modeling of the MOS transistor*. [S.l.]: WCB, McGraw-Hill, 1999. Citado na página 23.
- VELAZCO, R.; FOUILLAT, P.; REIS, R. *Radiation effects on embedded systems*. [S.l.]: Springer Science & Business Media, 2007. Citado na página 28.
- VIAL, J. et al. Using tmr architectures for yield improvement. In: IEEE. *Defect and Fault Tolerance of VLSI Systems, 2008. DFTVS'08. IEEE International Symposium on*. [S.l.], 2008. p. 7–15. Citado na página 33.
- VOGT, H.; HENDRIX, M.; NENZI, P. Ngspice users manual version 27 (describes ngspice-27 release version). 2017. Citado na página 50.
- WANG, F.; AGRAWAL, V. D. Single event upset: An embedded tutorial. In: IEEE. *VLSI Design, 2008. VLSID 2008. 21st International Conference on*. [S.l.], 2008. p. 429–434. Citado na página 51.
- WESTE, N. H.; HARRIS, D. *CMOS VLSI design: a circuits and systems perspective*. [S.l.]: Pearson Education India, 2015. Citado 5 vezes nas páginas 23, 24, 25, 26 e 49.
- ZHAO, W.; CAO, Y. New generation of predictive technology model for sub-45 nm early design exploration. *IEEE Transactions on Electron Devices*, IEEE, v. 53, n. 11, p. 2816–2823, 2006. Citado na página 49.

Apêndices

APÊNDICE A – Diagramas *Sticks*

Nesta sessão são apresentados os diagramas *sticks* gerados das catorze topologias de votadores majoritários utilizados nessa dissertação. A Figura 25 apresenta os *sticks* das implementações do votador Clássico e do votador CMOS.

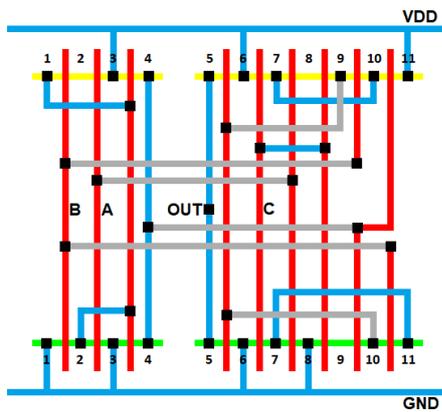
Figura 25 – Diagramas *sticks* dos circuitos votadores Clássico e CMOS.



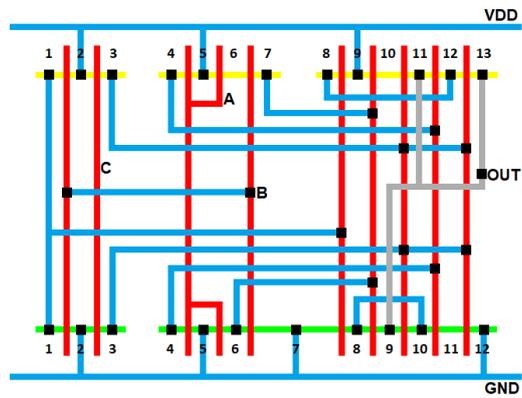
Fonte: O autor

Os diagramas *sticks* dos circuitos votadores Bala e Bala CMOS podem ser observados na Figura 26. E os *sticks* das implementações do votador Ban estão ilustrados na Figura 27.

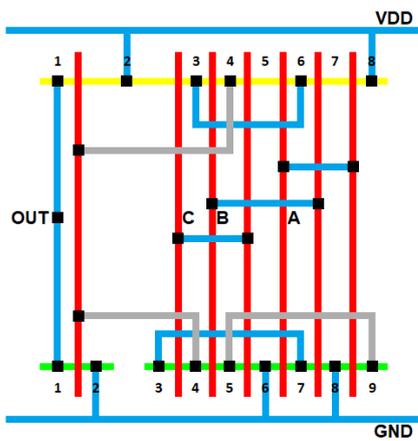
Figura 26 – Diagramas *sticks* dos circuitos votadores Bala e Bala CMOS.



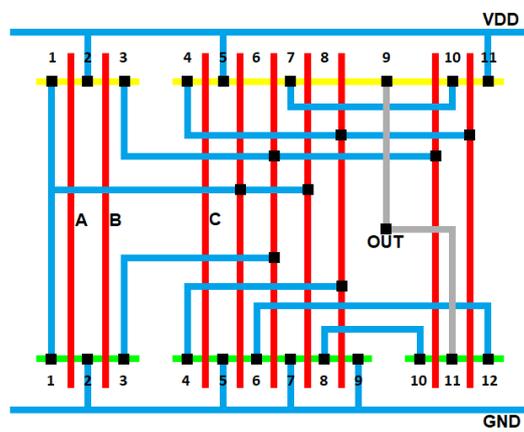
(a) Bala (OLIVEIRA; SCHVITZ; BUTZEN, 2019)



(b) Bala 2.



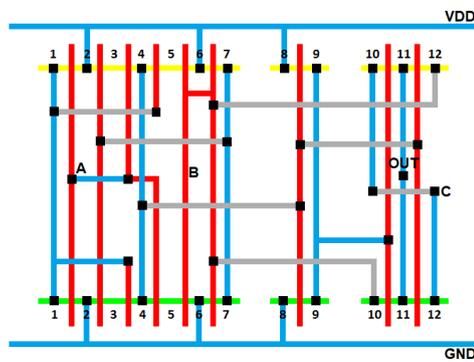
(c) Bala CMOS (OLIVEIRA; SCHVITZ; BUTZEN, 2019)



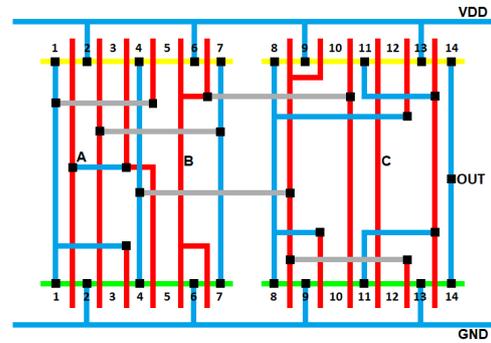
(d) Bala CMOS 2.

Fonte: O autor.

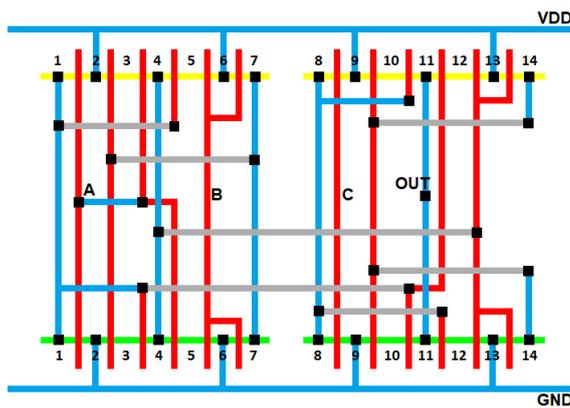
Figura 27 – Diagramas *sticks* dos circuitos votadores Ban.



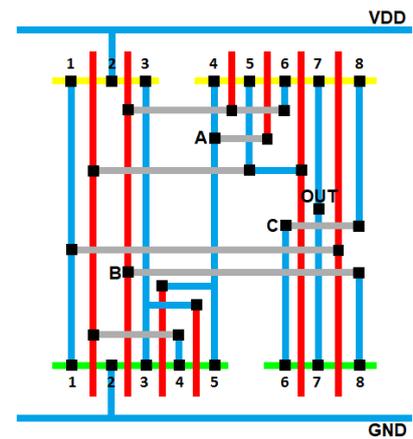
(a) Ban (OLIVEIRA; SCHVITZ; BUTZEN, 2019)



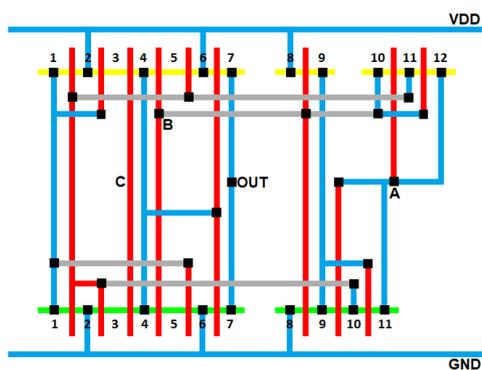
(b) Ban 2.



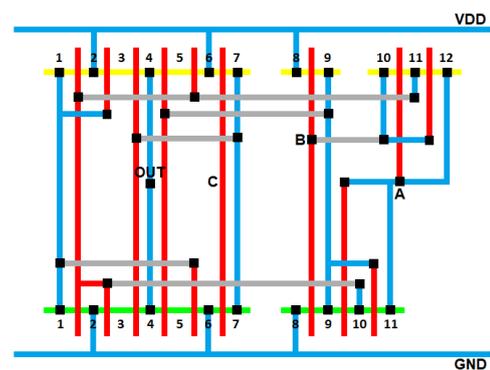
(c) Ban 3.



(d) Ban 4.



(e) Ban 5.



(f) Ban 6.

Fonte: O autor.

Tabela 14 – Análise das áreas ativas sensíveis para o votador majoritário CMOS para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS					Áreas Ativas NMOS				
	1	3	4	5	6	1	3	5	6	7
000	X	0	0	0	0	0	0	0	X	0
001	X	0	0	0	0	0	0	0	X	0
010	X	0	0	0	0	0	X	0	X	X
011	1	1	1	X	1	X	1	1	1	1
100	X	0	0	0	0	0	0	X	X	0
101	1	X	1	X	X	X	1	1	1	1
110	1	1	X	X	1	X	1	1	1	1
111	1	1	1	X	1	X	1	1	1	1

Tabela 15 – Análise das áreas ativas sensíveis para o votador majoritário CMOS 2 para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS							Áreas Ativas NMOS						
	1	3	4	6	7	8	9	1	3	4	6	7	8	10
000	0	0	0	0	0	X	0	0	0	0	0	0	0	0
001	0	0	0	0	X	X	0	X	X	0	0	0	0	0
010	0	0	0	X	0	X	X	X	0	X	0	0	0	0
011	1	X	X	1	1	1	1	1	1	1	X	X	1	1
100	0	0	0	0	0	X	0	0	X	X	0	0	0	0
101	X	1	X	1	1	1	1	1	1	1	1	X	X	X
110	X	X	1	1	1	1	1	1	1	1	1	X	1	1
111	1	1	1	1	1	1	1	1	1	1	1	X	1	1

Tabela 16 – Análise das áreas ativas sensíveis para o votador majoritário Bala para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS								Áreas Ativas NMOS						
	1	2	4	5	7	8	9	10	2	4	5	7	9	10	11
000	0	0	0	X	0	0	0	0	0	0	0	0	0	X	0
001	0	0	X	X	0	0	0	0	X	0	0	0	0	X	0
010	0	0	0	X	0	0	0	0	0	0	0	X	X	X	X
011	1	1	1	1	1	1	X	1	1	1	X	1	1	1	1
100	0	0	0	X	0	0	0	0	0	0	0	0	X	X	0
101	X	X	1	1	X	1	X	X	1	X	X	1	1	1	1
110	1	1	1	1	1	X	X	1	1	1	X	1	1	1	1
111	1	1	1	1	1	1	X	1	1	1	X	1	1	1	1

Tabela 20 – Análise das áreas ativas sensíveis para o votador majoritário Ban para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS									Áreas Ativas NMOS								
	1	3	4	5	7	9	10	11	12	1	3	4	5	7	9	10	11	12
000	0	0	0	0	0	0	0	X	X	0	0	0	0	0	0	0	0	0
001	0	X	X	0	0	0	0	X	X	X	0	0	X	X	0	0	0	0
010	0	0	0	0	X	X	X	X	0	0	X	X	0	0	0	0	0	0
011	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	X	X	0
100	0	0	0	0	0	0	X	X	0	0	0	0	0	0	0	0	0	0
101	X	1	1	1	1	X	1	1	1	1	1	X	X	1	1	1	X	X
110	1	1	X	X	1	1	1	1	1	1	1	1	1	X	X	X	X	1
111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	X	X	X	1

Tabela 21 – Análise das áreas ativas sensíveis para o votador majoritário Ban 2 para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS											Áreas Ativas NMOS									
	1	3	4	5	7	8	10	11	12	14	1	3	4	5	7	8	10	11	12	14	
000	0	0	0	0	0	0	0	0	0	X	0	0	0	0	0	0	0	X	0	0	
001	0	X	X	0	0	0	0	0	0	X	X	0	0	0	X	0	0	X	X	0	
010	0	0	0	0	X	X	0	0	0	X	0	X	X	0	0	0	X	X	0	0	
011	1	1	1	1	1	1	1	X	1	1	1	1	1	1	1	1	1	1	1	X	
100	0	0	0	0	0	0	0	0	0	X	0	0	0	0	0	0	X	0	0	0	
101	X	1	1	1	1	1	X	X	1	1	1	1	X	X	1	1	1	1	1	X	
110	1	1	X	X	1	1	1	X	X	1	1	1	1	1	X	1	1	1	1	X	
111	1	1	1	1	1	1	1	X	1	1	1	1	1	1	1	1	1	1	1	X	

Tabela 22 – Análise das áreas ativas sensíveis para o votador majoritário Ban 3 para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS											Áreas Ativas NMOS									
	1	3	4	5	7	8	10	11	12	14	1	3	4	5	7	8	10	11	12	14	
000	0	0	0	0	0	0	0	X	0	0	0	0	0	0	0	0	0	0	0	0	
001	0	X	X	0	0	0	X	X	0	0	X	0	0	0	X	0	0	0	0	X	
010	0	0	0	0	0	0	0	X	0	0	0	0	0	0	X	0	0	0	0	0	
011	1	1	1	1	X	X	1	1	1	X	1	X	X	1	1	1	X	X	1	1	
100	0	0	0	0	0	0	0	X	X	0	0	0	X	X	0	X	0	0	0	0	
101	X	1	1	1	1	X	1	1	1	1	1	1	1	1	1	1	X	1	1	1	
110	X	1	X	X	1	1	1	1	1	1	1	1	1	1	1	1	X	X	X	1	
111	X	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	X	1	1	1	

Tabela 23 – Análise das áreas ativas sensíveis para o votador majoritário Ban 4 para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS							Áreas Ativas NMOS						
	1	3	4	5	6	7	8	1	3	4	5	6	7	8
000	0	0	0	0	0	X	0	0	0	0	0	0	0	0
001	0	0	X	X	X	X	0	X	0	0	0	0	0	0
010	X	X	0	0	0	X	X	0	0	X	0	0	0	0
011	1	1	1	1	1	1	1	1	1	1	1	X	X	1
100	0	0	0	0	0	X	X	0	0	0	0	0	0	0
101	X	1	1	1	1	1	1	1	X	X	X	X	X	1
110	1	X	1	X	1	1	1	X	1	1	1	1	X	X
111	1	1	1	1	1	1	1	1	1	1	1	1	X	X

Tabela 24 – Análise das áreas ativas sensíveis para o votador majoritário Ban 5 para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS									Áreas Ativas NMOS								
	1	3	4	5	7	9	10	11	12	1	3	4	5	7	9	10	11	
000	0	0	0	0	X	0	0	0	0	0	0	X	0	0	0	0	0	
001	0	0	0	0	X	0	X	X	X	0	X	X	0	0	0	0	0	
010	X	0	0	0	X	X	0	0	0	0	0	X	X	0	0	X	0	
011	1	1	X	1	1	1	1	1	1	1	1	1	1	X	1	1	1	
100	0	0	0	0	X	0	0	0	0	0	0	X	0	0	0	0	0	
101	1	1	X	X	1	1	1	1	1	1	1	1	1	X	X	X	X	
110	1	X	X	1	1	X	1	X	1	X	1	1	1	X	1	1	1	
111	1	1	X	1	1	1	1	1	1	1	1	1	1	X	1	1	1	

Tabela 25 – Análise das áreas ativas sensíveis para o votador majoritário Ban 6 para todos os vetores de entrada

Vetor de Entrada	Áreas Ativas PMOS									Áreas Ativas NMOS								
	1	3	4	5	7	9	10	11	12	1	3	4	5	7	9	10	11	
000	0	0	X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
001	0	X	X	0	0	0	X	X	X	X	0	0	0	0	0	0	0	
010	0	0	X	X	0	0	0	0	0	0	0	0	0	X	0	X	0	
011	1	1	1	1	X	1	1	1	1	1	1	X	1	1	1	1	1	
100	0	0	X	0	0	0	0	0	0	0	0	0	0	X	0	0	0	
101	X	1	1	1	X	1	1	1	1	1	1	X	X	1	1	X	X	
110	1	1	1	1	1	X	1	X	1	1	X	X	1	1	1	1	1	
111	1	1	1	1	1	1	1	1	1	1	1	X	1	1	1	1	1	

Tabela 26 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Clássico considerando a tecnologia de 32nm.

Vetor de Entrada	Clássico NANDS			Clássico NORs		
	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}
000	PMOS 8	out	0,78	PMOS 11	out	1,95
	PMOS 10					
	NMOS 1	v1	1,76			
	NMOS 5	v2	1,77			
	NMOS 6	v3	1,78			
001	PMOS 8	out	0,78	PMOS 9	n4	0,70
	PMOS 10			PMOS 10	n5	0,70
	NMOS 1	v1	1,76	PMOS 11	out	0,69
	NMOS 5	v2	0,89	NMOS 2	v1	0,82
	NMOS 6	v3	0,90			
010	PMOS 8	out	0,79	PMOS 10	n5	0,69
	PMOS 10			PMOS 11	out	0,69
	NMOS 1	v1	0,89	NMOS 4	v2	0,81
	NMOS 5	v2	1,77			
	NMOS 6	v3	0,91			
	NMOS 7	n3	0,90			
011	PMOS 6	v3	0,74	PMOS 1	v1	0,71
	NMOS 9	n5	0,84	PMOS 5	v2	0,71
	NMOS 10	n4	0,84	PMOS 6	v3	1,37
	NMOS 11	out	0,84	NMOS 8	out	0,77
				NMOS 10		
100	PMOS 8	out	0,78	PMOS 11	out	0,69
	PMOS 10			NMOS 6	v3	0,79
	NMOS 1	v1	0,89			
	NMOS 2	n1	0,89			
	NMOS 4	n2	0,89			
	NMOS 5	v2	0,90			
NMOS 6	v3	1,78				
101	PMOS 4	v2	0,74	PMOS 1	v1	0,73
	NMOS 10	n4	0,84	PMOS 2	n1	0,72
	NMOS 11	out	0,85	PMOS 5	v2	1,38
				PMOS 6	v3	0,70
				NMOS 8	out	0,77
			NMOS 10			
110	PMOS 2	v1	0,74	PMOS 1	v1	1,40
	NMOS 11	out	0,83	PMOS 4	n2	0,71
				PMOS 5	v2	0,72
				PMOS 6	v3	0,71
				PMOS 7	n3	0,70
				NMOS 8	out	0,77
			NMOS 10			
111	NMOS 11	out	2,44	PMOS 1	v1	1,40
				PMOS 5	v2	1,38
				PMOS 6	v3	1,37
				NMOS 8	out	0,77
				NMOS 10		

Tabela 27 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores CMOS considerando a tecnologia de $32nm$.

Vetor de Entrada	CMOS			CMOS 2		
	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}
000	PMOS 1	out	0,68	PMOS 8	out	1,70
	NMOS 6	v1	1,75			
001	PMOS 1	out	0,68	PMOS 7	n2	0,78
	NMOS 6	v1	0,99	PMOS 8	out	0,78
				NMOS 1	v1	0,91
				NMOS 3	v2	0,87
010	PMOS 1	out	0,68	PMOS 6	n1	0,78
	NMOS 6	v1	0,96	PMOS 9		
	NMOS 3	n4	0,94	PMOS 8	out	0,77
	NMOS 7			NMOS 1	v1	0,89
			NMOS 4	v3	0,89	
011	PMOS 5	v1	0,75	PMOS 3	v2	0,69
	NMOS 1	out	0,83	PMOS 4	v3	0,73
				NMOS 6	n3	0,99
				NMOS 7	out	0,99
100	PMOS 1	out	0,68	PMOS 8	out	0,77
	NMOS 5	n3	1,00	NMOS 3	v2	0,86
	NMOS 6	v1	1,01	NMOS 4	v3	0,88
101	PMOS 3	n1	0,77	PMOS 1	v1	0,71
	PMOS 6			PMOS 4	v3	0,71
	PMOS 5	v1	0,77	NMOS 7	out	0,94
	NMOS 1	out	0,83	NMOS 8	n4	0,93
			NMOS 10			
110	PMOS 4	n2	0,77	PMOS 1	v1	0,72
	PMOS 5	v1	0,78	PMOS 3	v2	0,68
	NMOS 1	out	0,83	NMOS 7	out	0,97
111	PMOS 5	v1	1,72	NMOS 7	out	1,70
	NMOS 1	out	0,83			

Tabela 28 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Bala considerando a tecnologia de 32nm.

Vetor de Entrada	Bala			Bala 2		
	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}
000	PMOS 5	out	0,68	PMOS 11	out	1,64
	NMOS 10	v3	1,75	PMOS 13		
001	PMOS 4	v2	0,71	PMOS 10	n2	0,92
	PMOS 5	out	0,68	PMOS 8	n3	0,92
	NMOS 2	v1	0,82	PMOS 12	out	0,92
	NMOS 10	v3	0,99	PMOS 11		
				PMOS 13		
				NMOS 1		
				NMOS 6	v4	0,83
010	PMOS 5	out	0,68	PMOS 11	out	0,80
	NMOS 9	n4	0,79	PMOS 13	v3	0,83
	NMOS 10	v3	0,80	NMOS 3		
	NMOS 7	n5	0,79	NMOS 4		
	NMOS 11					
011	PMOS 9	v3	1,48	PMOS 3	v3	0,73
	NMOS 5	out	0,83	NMOS 9	out	1,55
100	PMOS 5	out	0,68	PMOS 8	n3	0,92
	NMOS 9	n4	1,00	PMOS 12	out	0,92
	NMOS 10	v3	1,01	PMOS 11		
				PMOS 11	v2	0,91
				NMOS 3	v3	0,90
101	PMOS 1	v1	0,69	PMOS 3	v3	0,71
	PMOS 2	n1	0,68	PMOS 6	n1	0,70
	PMOS 9	v3	0,77	PMOS 7	v4	0,71
	PMOS 7	n2	0,77	NMOS 9	out	0,82
	PMOS 10					
	NMOS 4	v2	0,87			
	NMOS 5	out	0,83			
110	PMOS 8	n3	0,77	PMOS 1	v2	0,70
	PMOS 9	v3	0,78	PMOS 4	v1	0,69
	NMOS 5	out	0,83	NMOS 9	out	0,81
				NMOS 8	n4	0,80
				NMOS 10		
111	PMOS 9	v3	1,72	NMOS 9	out	1,77
	NMOS 5	out	0,83			

Tabela 29 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Bala CMOS considerando a tecnologia de 32nm.

Vetor de Entrada	Bala CMOS			Bala CMOS 2		
	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}
000	PMOS 1	out	0,68	PMOS 9	out	1,94
	NMOS 4	v1	1,95			
001	PMOS 1	out	0,68	PMOS 8	n3	0,78
	NMOS 3	n4	0,97	PMOS 9	out	0,78
	NMOS 7			NMOS 1	v1	0,90
	NMOS 4	v1	0,98	NMOS 3	v2	0,87
010	PMOS 1	out	0,68	PMOS 6	n1	0,79
	NMOS 4	v1	0,94	PMOS 9	out	0,78
	NMOS 5	n5	0,93	PMOS 7	n2	0,80
	NMOS 9			PMOS 10		
				NMOS 1	v1	0,91
				NMOS 4	v3	0,91
011	PMOS 4	v1	1,49	PMOS 3	v2	0,72
	NMOS 1	out	0,83	PMOS 4	v3	0,75
				NMOS 11	out	1,22
100	PMOS 1	out	0,68	PMOS 9	out	1,48
	NMOS 4	v1	1,25	NMOS 3	v2	0,97
				NMOS 4	v3	0,97
101	PMOS 4	v1	0,78	PMOS 1	v1	0,73
	PMOS 3	n2	0,79	PMOS 4	v3	0,69
	PMOS 6			NMOS 8	n5	0,93
	PMOS 7	n1	0,78	NMOS 10		
	NMOS 1	out	0,83	NMOS 11	out	0,93
110	PMOS 4	v1	0,78	PMOS 1	v1	0,75
	PMOS 5	n3	0,77	PMOS 3	v2	0,69
	NMOS 1	out	0,83	NMOS 11	out	0,96
				NMOS 6	n4	0,96
111	PMOS 4	v1	1,97	NMOS 11	out	1,91
	NMOS 1	out	0,83			

Tabela 30 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Ban considerando a tecnologia de 32nm (parte I).

Vetor de Entrada	Ban			Ban 2			Ban 3		
	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}
000	PMOS 11	out	0,39	PMOS 14	out	0,68	PMOS 11	out	0,77
	PMOS 12	sb	0,97	NMOS 11	v5	0,81			
001	PMOS 3	n2	0,75	PMOS 3	n2	0,78	PMOS 3	n2	0,78
	PMOS 4	v3	0,76	PMOS 4	v3	0,79	PMOS 4	v4	0,78
	PMOS 11	out	0,39	PMOS 14	out	0,68	PMOS 10	n6	0,77
	PMOS 12	sb	0,71	NMOS 1	v1	0,88	PMOS 11	out	0,77
	NMOS 1	v1	0,86	NMOS 7	v2	0,90	NMOS 1	v1	0,87
	NMOS 7	v2	0,87	NMOS 11	v5	0,81	NMOS 7	v2	0,89
	NMOS 9	v4	0,90	NMOS 12	n8	0,80	NMOS 14	v5	0,89
010	PMOS 7	v2	0,68	PMOS 7	v2	0,69	PMOS 11	out	0,77
	PMOS 9	v4	0,99	PMOS 8	v4	0,68	NMOS 8	v3	0,88
	PMOS 10	sc	0,74	PMOS 14	out	0,68			
	PMOS 11	out	0,39	NMOS 3	n3	0,81			
	NMOS 3	n3	0,83	NMOS 4	v3	0,82			
	NMOS 4	v3	0,84	NMOS 10	n7	0,80			
011				NMOS 11	v5	0,81			
	NMOS 11	out	0,41	PMOS 11	v5	0,76	PMOS 7	v2	0,69
	NMOS 12	sc	0,94	NMOS 14	out	0,83	PMOS 8	v3	0,68
							PMOS 14	v5	0,69
							NMOS 3	n3	0,81
							NMOS 4	v4	0,82
100							NMOS 10	n7	0,79
							NMOS 11	out	0,80
	PMOS 10	sc	0,74	PMOS 14	out	0,68	PMOS 12	n5	0,77
	PMOS 11	out	0,39	NMOS 11	v5	0,81	PMOS 11	out	0,77
							NMOS 4	v4	0,84
101							NMOS 5	n4	0,83
							NMOS 8	v3	0,87
	PMOS 1	v1	0,68	PMOS 1	v1	0,69	PMOS 1	v1	0,73
	PMOS 9	v4	0,69	PMOS 10	n5	0,75	PMOS 8	v3	0,69
	NMOS 4	v3	0,78	PMOS 11	v5	0,76	NMOS 11	out	0,80
110	NMOS 5	n4	0,78	NMOS 4	v3	0,83			
	NMOS 12	sc	0,94	NMOS 5	n4	0,83			
	NMOS 11	out	0,41	NMOS 14	out	0,83			
	PMOS 4	v3	0,79	PMOS 4	v3	0,78	PMOS 1	v1	0,69
	PMOS 5	n1	0,78	PMOS 5	n1	0,77	PMOS 4	v4	0,80
111	NMOS 9	v4	1,68	PMOS 11	v5	0,76	PMOS 5	n1	0,79
	NMOS 10	sb	0,91	PMOS 12	n6	0,75	NMOS 11	out	0,80
	NMOS 11	out	0,41	NMOS 8	v4	0,88	NMOS 12	n8	0,79
				NMOS 14	out	0,83			
111	NMOS 10	sb	1,22	PMOS 11	v5	0,76	PMOS 1	v1	0,69
	NMOS 11	out	0,41	NMOS 14	out	0,83	NMOS 11	out	0,80

Tabela 31 – Resultados de LET_{th} ($MeV.cm^{-2}/mg$) referente às áreas críticas dos votadores Ban considerando a tecnologia de 32nm (parte II).

Vetor de Entrada	Ban 4			Ban 5			Ban 6											
	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}	AAC	Nodo	LET_{th}									
000	PMOS 7	out	0,39	PMOS 7	out	0,68	PMOS 4	out	0,77									
				NMOS 4	v4	0,81												
001	PMOS 4	sa	0,71	PMOS 7	out	0,68	PMOS 3	n2	0,77									
	PMOS 5	v2	0,37	PMOS 10	sb	0,71	PMOS 4	out	0,77									
	PMOS 6	sb	0,79	PMOS 11	v2	0,39	PMOS 10	sb	0,70									
	PMOS 7	out	0,39	PMOS 12	sa	0,73	PMOS 11	v2	0,39									
	NMOS 1	v3	0,89	NMOS 3	n4	0,80	PMOS 12	sa	0,73									
010	PMOS 1	v3	0,80	PMOS 1	v3	0,68	PMOS 4	out	0,77									
										PMOS 3	v1	0,68	PMOS 7	out	0,68	PMOS 5	n1	0,77
										PMOS 7	out	0,39	PMOS 9	v1	0,68	NMOS 7	v4	0,87
										PMOS 8	sc	0,74	NMOS 4	v4	0,81	NMOS 10	v2	0,42
										NMOS 4	v2	0,41	NMOS 5	n3	0,80			
011	NMOS 6	sc	0,94	PMOS 4	v4	0,76	PMOS 1	v3	z									
				NMOS 7	out	0,41	NMOS 7	out	0,83	PMOS 7	v4	0,69						
100	PMOS 7	out	0,39	PMOS 1	v3	z	PMOS 4	out	0,77									
										PMOS 8	sc	0,74	PMOS 7	out	0,68	NMOS 7	v4	0,88
101	PMOS 1	v3	0,69	PMOS 4	v4	0,76	PMOS 1	v3	0,69									
										NMOS 3	v1	1,47	PMOS 5	n1	0,75	PMOS 7	v4	0,68
										NMOS 4	v2	0,37	NMOS 7	out	0,83	NMOS 4	out	0,80
										NMOS 5	sa	0,90	NMOS 9	v1	1,60	NMOS 5	n3	0,79
										NMOS 6	sc	0,94	NMOS 10	v2	0,42	NMOS 10	v2	0,41
										NMOS 7	out	0,41	NMOS 11	sa	0,95	NMOS 11	sa	0,92
110	PMOS 3	v1	0,81	PMOS 3	n2	0,75	PMOS 9	v1	0,69									
										PMOS 5	v2	0,38	PMOS 4	v4	0,76	PMOS 11	v2	0,39
										NMOS 1	v3	1,68	PMOS 9	v1	0,80	NMOS 3	n4	0,80
										NMOS 7	out	0,41	PMOS 11	v2	0,38	NMOS 4	out	0,80
										NMOS 8	sb	0,83	NMOS 1	v3	0,88			
111	NMOS 7	out	0,41	PMOS 4	v4	0,76	NMOS 4	out	0,81									
										NMOS 8	sb	1,72	NMOS 7	out	0,83			