

Análise do comportamento de portas lógicas CMOS com falhas *Stuck-On* em nanotecnologias

Alexandra L. Zimpeck, Cristina Meinhardt e Paulo F. Butzen

Resumo — Os avanços tecnológicos em circuitos integrados tem como foco principal a redução da dimensão dos transistores. No entanto, esta redução traz consequências indesejáveis, como o possível aumento no número de falhas. Neste contexto, este trabalho tem por objetivo analisar o comportamento de um tipo de falha permanente que ocorre em transistores MOS chamada de *Stuck-On*. Esta falha se caracteriza por manter um transistor sempre conduzindo, e sua principal característica para detecção é que o consumo de potência de um circuito tende a ser muito maior. Em nanotecnologias, a potência total sofre ainda com o aumento do consumo estático das novas tecnologias. Isso torna relevante avaliar o comportamento de portas lógicas em tecnologias nanométricas na presença de falhas *Stuck-On*. Para isso, foi adotado um conjunto de portas lógicas em tecnologia CMOS, injetando falhas únicas *Stuck-On* nos circuitos e avaliando os efeitos da falha no consumo de potência. Além disso, este trabalho investiga o impacto de inserir técnicas de tolerância a falhas para que o circuito continue em funcionamento correto mesmo na presença de falhas. Foi adotada a técnica de TMR nas portas lógicas avaliadas, comprovando a tolerância a falhas individuais e demonstrando o impacto nas características de potência, desempenho e área.

Palavras-chave — Tecnologia CMOS, Portas Lógicas, *Stuck-On*, Tolerância a Falhas.

I. INTRODUÇÃO

A microeletrônica é a área responsável pela criação de circuitos integrados (CI), visando principalmente o aumento do desempenho e a redução das dimensões dos componentes. O circuito integrado é o item mais importante que está presente nos produtos eletrônicos modernos. Ele é composto de vários dispositivos semicondutores constituídos sobre uma pastilha de silício.

Devido à miniaturização dos componentes, podemos obter arquiteturas eletrônicas bastante complexas em uma pequena área. Com os componentes em escalas muito menores, os produtos eletrônicos atuais apresentam um aumento no número de funcionalidades, assim como um aumento no desempenho.

Em meados de 1965, Gordon E. Moore constatou que a cada período de 18 meses o número de transistores de um *chip* aumentaria em 100%, mantendo os custos de fabricação constante [1]. Assim, surgiu a Lei de Moore, que até hoje guia a evolução tecnológica e comercial de processadores. Observando o rápido desenvolvimento da indústria desde a introdução do circuito integrado, a Lei de Moore foi criada a fim de tentar entender e até mesmo prever os próximos passos. As indústrias de semicondutores passaram a se basear nesta lei para traçar seus objetivos de produção, investindo em muitas pesquisas e desenvolvimento para manter o ritmo dos avanços tecnológicos.

Os avanços tecnológicos em circuitos integrados têm como foco principal a redução das dimensões dos transistores. Transistores menores representam menores áreas para os circuitos, maior fator de integração, melhor desempenho e menor consumo de potência. Além dos benefícios trazidos aos circuitos, essa redução à escalas nanométricas também trouxe efeitos indesejáveis, como o aumento dos efeitos devido ao envelhecimento, à variabilidade, as correntes de fuga, assim como, o aumento do número de possíveis falhas [2].

As falhas podem ser consideradas como uma operação incorreta no sistema, que pode ter sido originada de erros de projeto, defeitos de fabricação ou interferência externa [3]. Elas podem ser de caráter permanente ou transiente. Falhas permanentes são aquelas que uma vez que o componente falha, ele nunca volta a funcionar corretamente. Enquanto as falhas transientes são aquelas que possuem duração limitada, causada por um mau funcionamento temporário. Tais falhas podem também ser intermitentes, ocorrendo repetidamente por curtos intervalos de tempo.

Além disso, falhas podem ser classificadas em falhas de *software* ou falhas de *hardware*. Falhas de *software* estão relacionadas com as etapas de engenharia de *software* e de desenvolvimento de programas. Por outro lado, falhas de *hardware* estão relacionadas a falhas que afetam componentes do sistema computacional. Falhas de *hardware* podem acontecer em diferentes níveis do projeto, podendo ser divididas em falhas de projeto, falhas de portas lógicas e falhas em transistores.

Existem dois principais tipos de falhas permanentes que ocorrem em transistores MOS, as falhas *Stuck-On* e as falhas *Stuck-Open*. As falhas *Stuck-On* se caracterizam por fazer com que o transistor conduza permanentemente, independente da tensão aplicada ao seu terminal de controle, o *gate*. Diferentemente das falhas *Stuck-On*, as falhas *Stuck-Open* se caracterizam por fazer com que o transistor opere sempre

Alexandra L. Zimpeck é mestranda em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (PPGC/UFRGS). Possui graduação em Engenharia de Computação pela Universidade Federal do Rio Grande (FURG). E-mail: alexandra.zimpeck@gmail.com.

Cristina Meinhardt é mestre e doutoranda em Ciência de Computação pela Universidade Federal do Rio Grande do Sul (PPGC/UFRGS). Possui graduação em Engenharia de Computação pela Universidade Federal do Rio Grande (FURG). Atualmente atua como professora no Centro de Ciências Computacionais da Universidade Federal do Rio Grande (C3/FURG). E-mail: cristinameinhardt@furg.br.

Paulo F. Butzen é graduado em Engenharia de Computação, mestre em Ciência da Computação e doutor em Microeletrônica pela Universidade Federal do Rio Grande Sul (UFRGS). Atualmente atua como professor no Centro de Ciências Computacionais da Universidade Federal do Rio Grande (C3/FURG). E-mail: paulobutzen@furg.br.

como uma chave aberta, independe da tensão aplicada ao terminal de controle.

O aumento destas falhas permanentes em nanotecnologias torna relevante reavaliar o comportamento de circuitos na presença destas falhas. Neste contexto, este artigo foca na análise do comportamento de portas lógicas em tecnologias nanométricas na presença de falhas *Stuck-On*. Por manter transistores constantemente conduzindo, falhas *Stuck-On* tem impacto considerável no consumo de potência de circuitos. Em nanotecnologias, o consumo estático aumentou consideravelmente, e este artigo investiga como falhas *Stuck-On* afetam tanto a potência dinâmica como estática de circuitos atuais.

Na próxima seção serão explicados conceitos fundamentais para a melhor compreensão deste artigo. A seção III é destinada a mostrar conceitos sobre tolerância a falhas. O comportamento das falhas *Stuck-On*, bem como as suas propriedades serão apresentadas na seção IV. A seção V fala sobre as falhas *Stuck-On* em tecnologias atuais. A seção VI mostra as portas lógicas analisadas neste artigo. A seção VII apresentará os experimentos realizados e os resultados obtidos com e sem a presença da falha. Resultados referentes a aplicação de uma das técnicas de tolerância a falhas é mostrado na seção VIII. E por fim, a seção IX contém as conclusões.

II. CONCEITOS FUNDAMENTAIS DE MICROELETRÔNICA

Esta seção tem como objetivo apresentar alguns conceitos importantes da área da microeletrônica e tolerância a falhas. Estes conhecimentos são necessários para que haja uma boa compreensão dos experimentos e resultados deste artigo.

A. Transistores MOS

O transistor MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*), que também pode ser chamado pela abreviação MOS, é um componente elétrico que está presente em todos os produtos eletrônicos existentes.

É composto basicamente por quatro terminais que chamamos de dreno (*drain*), fonte (*source*), porta (*gate*) e substrato (*bulk*). Sua estrutura é mostrada na Fig. 1. As três camadas básicas que constituem um transistor são: o metal; o óxido de silício que isola o *gate* do restante do transistor; e o semiconductor (substrato) que é formado por silício monocristalino [4].

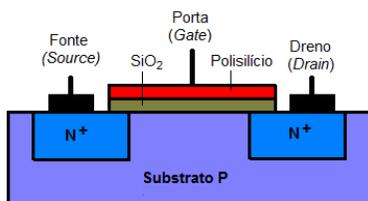


Fig. 1. Estrutura do transistor MOS

A Fig. 2 mostra a estrutura do transistor no espaço tridimensional para ilustrar as duas principais dimensões do canal: a largura (*width; W*) e o comprimento (*length; L*). São essas as medidas responsáveis por definir as principais características elétricas dos transistores.

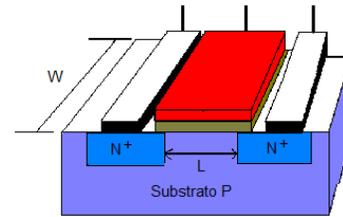


Fig. 2. Dimensões do canal do transistor

O comprimento (*length; L*) corresponde ao comprimento do canal, isto é, a tecnologia utilizada pelo transistor. Quando falamos que uma tecnologia é de X nanômetros, queremos dizer que a distância entre as difusões também é de X nanômetros. Esse valor normalmente é fixo para todos os transistores de um circuito digital.

A largura (*width; W*) está relacionada com o fluxo de corrente que irá passar pelos transistores quando eles estão conduzindo. Este valor pode ser variável para os transistores de um circuito digital. Portanto, quando circuitos integrados são projetados, é de grande relevância o dimensionamento da largura W dos transistores.

B. Funcionamento dos transistores

Os transistores MOS podem ser considerados dispositivos de três terminais que se comportam como chaves lógicas, onde o valor lógico no terminal de controle (*gate*) determina a existência ou não de conexão entre os outros dois terminais (*source* e *drain*) [5]. A fig. 3 mostra o funcionamento.

Existem dois tipos de transistores MOS: o NMOS que conduzirá quando o valor lógico '1' for aplicado em seu terminal de controle; e o PMOS que apresenta o comportamento contrário, e conduzirá quando o valor lógico '0' for aplicado no *gate*.

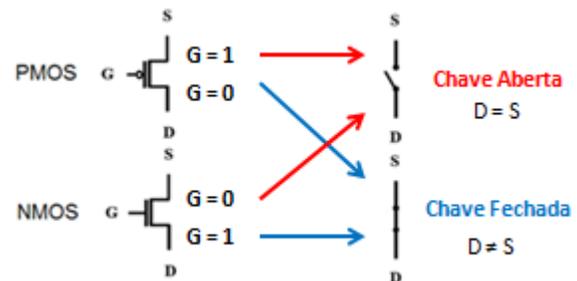


Fig. 3. Simbologia e funcionamento dos transistores MOS como chave

Há um valor de tensão específico que, quando aplicado ao terminal de *gate*, indica a troca no modo de operação do transistor [4]. Essa troca pode ser de condução (*On*) para corte (*off*) ou vice-versa. Este valor de tensão é chamado de tensão de limiar ou tensão de *threshold* (V_{th}).

Os transistores NMOS são formados pelo substrato com cargas positivas (lacunas) e por dois blocos com dopantes de carga negativa (elétrons). Quando uma tensão positiva é aplicada no *gate* (V_G), os elétrons presentes no substrato são atraídos e as lacunas são repelidas.

Quando a tensão de *gate* for suficientemente maior do que a tensão de *threshold* ($V_G > V_{th}$), um canal será formado. Este

canal terá elétrons que flutuam entre o dreno e a fonte carregados com cargas negativas [6]. Sendo assim, um transistor NMOS funciona idealmente como chave fechada quando $V_G \geq V_{th}$ e como chave aberta quando $V_G < V_{th}$. Este comportamento pode ser verificado na Fig. 4.

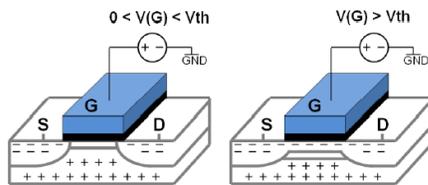


Fig. 4. Mecanismo de funcionamento do transistor NMOS [4]

Os transistores PMOS possuem comportamento similar aos transistores NMOS, contudo a polaridade das cargas é invertida. O substrato é composto por cargas negativas e os blocos com dopantes por cargas positivas. Além disso, o substrato no transistor PMOS é ligado à fonte de alimentação (V_{DD}), assim qualquer tensão aplicada a *gate* que for menor do que V_{DD} será considerada negativa. Com essa tensão negativa, os elétrons presentes no substrato são repelidos e as lacunas são atraídas. Sendo assim, o transistor PMOS funciona idealmente como chave fechada quando $V_G \leq V_{th}$ e como chave aberta quando $V_G > V_{th}$.

Vamos considerar a tensão de *threshold* de um transistor NMOS, isto é, que conduz corrente quando $V_G \geq V_{th}$. Quando os valores de tensão estão próximos ou abaixo de V_{th} , dizemos que o circuito está operando em *near-threshold* ou *subthreshold*, respectivamente.

Idealmente, não deveria existir corrente quando o transistor está cortado ($V_G < V_{th}$). Entretanto, como mostrado na Fig. 5, a condução de corrente inicia ainda quando a tensão no *gate* é menor do que V_{th} . A corrente de *subthreshold* aumenta drasticamente de acordo com a redução da tensão de *threshold* [6].

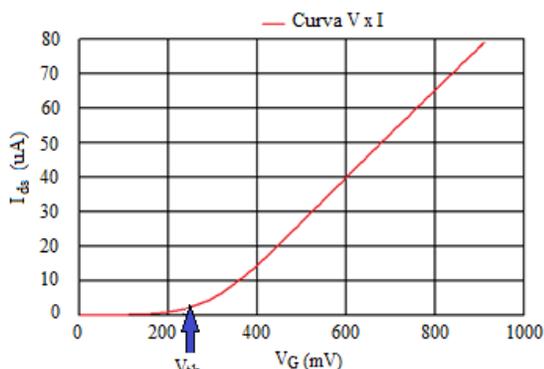


Fig. 5. Curva V x I

C. Lógica com transistores

Dentro de um circuito é possível interligar os transistores por meio de associações. O comportamento das associações varia conforme a ligação entre eles, as quais podem ser em série ou em paralelo. Também podem existir outros tipos de ligações que não se classificam nem como série e nem como

paralela, permitindo que qualquer função lógica possa ser construída.

A associação em série (Fig. 6) consiste em colocar os transistores em seqüências, ou seja, ligar o dreno do transistor A na fonte do transistor B ou vice-versa. Além disso, pode-se perceber que apenas haverá fluxo de corrente quando todos os transistores associados estiverem polarizados como chave fechada. Desta forma, pode-se afirmar que transistores em série emulam a lógica E.

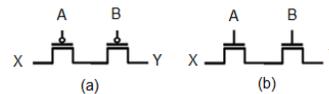


Fig. 6. Associação em série de transistores (a) PMOS e (b) NMOS

A associação em paralelo (Fig. 7) ocorre quando dois ou mais transistores estão ligados entre dois pontos (X e Y), ou seja, liga o dreno dos transistores A e B ao ponto X e a fonte dos transistores A e B ao ponto Y ou vice-versa. Só não haverá fluxo de corrente se ambos os transistores estiverem operando como chaves abertas. Desta forma, é possível afirmar que transistores em paralelo emulam a lógica OU.

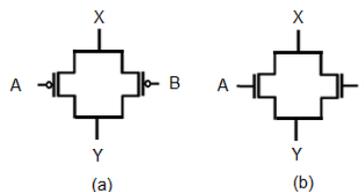


Fig. 7. Associação em paralelo de transistores (a) PMOS e (b) NMOS

D. Portas Lógicas

As portas lógicas de um circuito integrado são formadas por transistores que são interligados de forma a executarem a função lógica desejada, sendo capaz de gerar em sua saída um determinado nível lógico em função das suas entradas.

O uso da tecnologia CMOS (*Complementary Metal-Oxide-Semiconductor*) ganhou destaque, pois apresenta um consumo de potência muito inferior ao da tecnologia NMOS, que foi intensamente usada até o início dos anos oitenta.

As portas lógicas CMOS são compostas por transistores MOS, de modo a formar dois planos: o plano *pull-up* e o plano *pull-down* [6], a representação destes planos pode ser vista na Fig. 8.

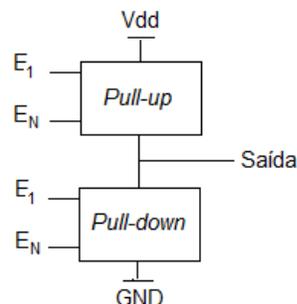


Fig. 8. Esquema geral das portas lógicas CMOS

O plano *pull-up* é composto de transistores do tipo PMOS, sendo responsável por ligar VDD até a saída da porta lógica. Os transistores do tipo NMOS fazem parte do plano *pull-down*, e tem como função formar um caminho que ligue GND à saída. Esses planos são logicamente complementares, isto é, a partir do momento em que um plano está conduzindo, apenas esse plano conectará a saída a um dos terminais de alimentação.

Como a lógica CMOS é voltada para a construção de lógica negada, as portas lógicas consideradas como básicas são o inversor, a NOR e a NAND. O inversor é o responsável por inverter o valor de entrada. As portas NOR e NAND são o inverso das funções OR e AND, respectivamente. A Fig. 9 mostra o símbolo, a tabela verdade e o diagrama esquemático (arranjo dos transistores) para estas portas lógicas CMOS básicas.

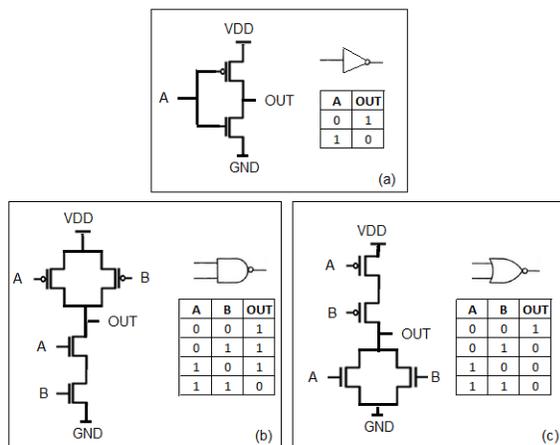


Fig. 9. Símbolo, tabela verdade e esquemático das portas lógicas (a) inversor, (b) NAND e (c) NOR

E. Características elétricas de portas lógicas

Existem duas características elétricas fundamentais quando lidamos com portas lógicas, o desempenho e a potência.

Para melhor entender o consumo de potência, é importante saber como ela é composta em circuitos CMOS. O cálculo da potência é dividido em duas partes, ao qual chamamos de potência estática e potência dinâmica. A composição da potência total do circuito é mostrada em (1).

$$P_{TOTAL} = P_{ESTÁTICA} + P_{DINÂMICA} \quad (1)$$

A potência dinâmica é consumida enquanto o circuito está mudando de estado, isto é, pelo carregamento e descarregamento de capacitâncias existentes nas conexões e nos *gates* dos transistores MOS. Já a potência estática é consumida enquanto o circuito não está mudando de estado, ou seja, está em uma condição estática. Essa potência era praticamente nula e desconsiderada, porém com a redução da tecnologia ela deixou de ser desprezível.

Idealmente, quando o transistor está cortado, considera-se que não há nenhuma corrente passando por ele. Porém, na prática esta afirmação é incorreta, já que existem correntes de baixa magnitude que geram consumo estático [5]. Estas correntes são chamadas de correntes de fuga, e estão se

tornando maiores à medida que a dimensão dos transistores vem diminuindo.

As correntes de fuga consideradas como mais importantes são a corrente de *subthreshold* e a corrente através do óxido de *gate* (Fig. 10). A corrente de *subthreshold* ocorre entre os terminais de fonte e dreno quando o transistor não está conduzindo. Sua magnitude se tornou elevada devido à redução da tensão de *threshold*, necessária para manter o desempenho dos dispositivos a reduzidas tensões de alimentação. Esta redução na tensão de alimentação foi necessária para estabilizar o consumo dinâmico que estava aumentando a cada nova tecnologia, devido a grande integração de dispositivos em nanotecnologias.

A corrente através do óxido de *gate* ocorrerá sempre que existir uma diferença de potencial através do óxido, podendo estar presente quando o transistor está conduzindo ou não. Essa corrente teve aumento devido à redução da espessura do óxido para controlar outros efeitos.

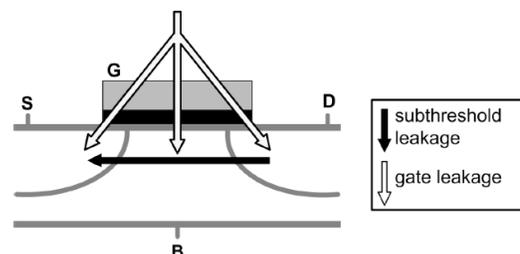


Fig. 10 - Principais correntes de fuga [4]

A segunda característica se refere ao desempenho de um circuito, o qual é medido através dos atrasos. Os atrasos são responsáveis por medir a diferença de tempo entre uma variação em um sinal de saída e a variação do sinal de entrada que gerou a transição na saída [7]. Para que se possa comparar um circuito em termos de desempenho é importante simular os tempos de propagação denominados de *delay high-low* (T_{pHL}) e *delay low-high* (T_{pLH}).

Os tempos de propagação são medidos como o tempo que uma transição na saída (V_{out}) demora a chegar até 50% do valor de V_{DD} a partir do momento que a entrada (V_{in}) diminui 50% do valor de V_{DD} . Os intervalos para medir os atrasos de propagação podem ser vistos na Fig. 11.

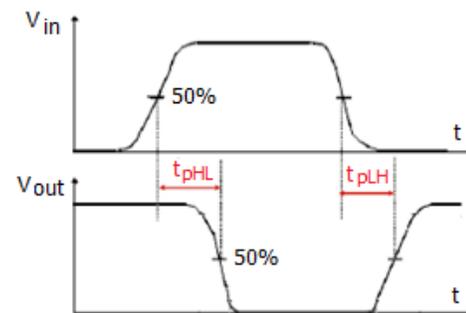


Fig. 11. Atrasos de propagação (Rabaey, et al., 2002)

Quanto maior forem os atrasos, mais lento será o circuito e consequentemente menor será seu desempenho. Para cada porta lógica analisada, os atrasos devem ser medidos considerando todos os arcos das funções lógicas. Um arco, se

tratando de circuitos eletrônicos, se refere à transição que apenas uma variável de entrada é transicionada.

F. Dimensionamento das portas lógicas

Para equalizar o tempo de resposta dos transistores PMOS e NMOS de um circuito, é necessário dimensionar os transistores para aumentar a sua velocidade. Os transistores PMOS são aproximadamente duas vezes mais lentos que os transistores NMOS, em função disso, uma regra básica de dimensionamento indica que a largura (W_p) dos transistores PMOS deve ser aproximadamente duas vezes a largura (W_n) dos transistores NMOS para obter desempenho semelhante.

Uma das principais técnicas adotadas para o dimensionamento de portas lógicas é o *Logical Effort*, também conhecido em português como Esforço Lógico. Com esta técnica é possível obter estimativas de atrasos em circuitos CMOS baseados apenas na topologia do caminho combinacional [8].

O *Logical Effort* busca amenizar o atraso de propagação de uma porta lógica a partir do redimensionamento dos transistores de um circuito. Quanto maior for a largura do transistor, menor será o atraso produzido por ele. A Fig. 12 mostra dois exemplos de dimensionamento por Esforço Lógico considerando um inversor e a porta lógica NAND.

O Esforço Lógico determina que quando existem arranjos de transistores em série deve-se duplicar a largura (W) de todos eles. Quando os transistores estão em paralelo não é preciso modificar a sua largura. Além disso, transistores PMOS serão sempre duplicados por serem mais lentos que transistores NMOS.

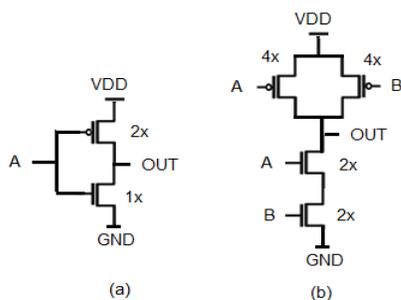


Fig. 12. Dimensionamento por Esforço Lógico no (a) inversor e na (b) NAND

G. Variabilidade de Processo

É normal que em processos de fabricação nanométricos aconteçam pequenos desvios nas dimensões de alguns dispositivos. Estes desvios recebem o nome de variabilidade [9]. Considerando o processo de fabricação de transistores com tamanhos diferentes, percebe-se que a variação será sempre mais acentuada nos dispositivos com tamanhos menores [10]. A principal desvantagem da variabilidade é a incerteza sobre o funcionamento esperado do circuito quando analisamos desempenho e potência após a fabricação [11][12][13].

H. Efeitos de envelhecimento

O envelhecimento dos circuitos também tem uma grande influência no desempenho dos transistores. Várias pesquisas tem mostrado que a corrente de saturação do transistor degrada ao longo dos anos por causa do desgaste do óxido [4]. A tendência é que esta degradação aumente à medida que novas tecnologias de transistores forem criadas [14][15].

III. TOLERÂNCIA A FALHAS

As falhas em sistemas de computação são inevitáveis, porém, suas consequências podem ser controladas pelo uso de técnicas adequadas. Alguns métodos impedem, por exemplo, a perda de dados, os colapsos no sistema e a interrupção no fornecimento de algum serviço [16].

As técnicas de tolerância a falhas permitem que o sistema se comporte de maneira aceitável. Mas, mesmo assim, conhecer os motivos que levaram o projeto a apresentar falha, estudar a falha ocorrida, tratar os problemas gerados por ela e recuperar os dados após sua ocorrência, é de fundamental importância para garantir um bom funcionamento de um sistema.

Nas próximas subseções serão apresentados os conceitos de falhas, erros, defeitos e tipos de mascaramentos. Também serão apresentadas algumas técnicas de tolerância a falhas.

A. Falha, erro e defeito

Tolerância a falhas é uma área de pesquisa bem abrangente, envolvendo pesquisadores de diferentes linhas de pesquisa, em diferentes níveis de abstração. Sendo assim, é comum ver vários pesquisadores usarem um mesmo termo para nomear diferentes conceitos de falhas. A fim de evitar conflitos entre termos e significados, o artigo irá considerar falha, erro e defeito de acordo com [3].

As falhas são consideradas como uma condição inesperada, que poderá ou não levar um sistema a atingir estados anormais. Normalmente, a falha é a causa raiz do efeito anômalo que poderá ser observado. Falhas podem ser originadas de erros de projeto, defeitos físicos ou interferências externas.

Dependendo das funcionalidades afetadas, uma falha poderá levar a um erro, isto é, uma alteração indesejada no estado do sistema. Assim, define-se que um sistema estará no estado errôneo se o processamento posterior, a partir do momento que ocorreu a falha, conduzir a um defeito. Um defeito é definido como um desvio da especificação do circuito, e não pode ser tolerado, apenas evitado.

Na Fig. 13 é mostrada uma simplificação para estes conceitos, classificando eles em três universos: o universo físico onde ocorrem as falhas, o universo da informação onde acontecem os erros e o terceiro, o universo do usuário onde aparecem os defeitos [17].

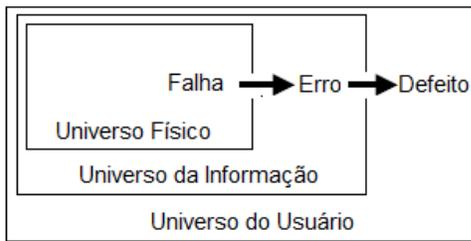


Fig. 13. Simplificação de falha, erro e defeito

B. Mascaramentos

O mascaramento permite que um sistema apresente uma resposta correta mesmo na presença de falhas. Neste caso, a falha não irá levar a um erro, pois são mascaradas em sua origem. Consequentemente também não haverá defeitos visíveis ao usuário de um sistema. Existem três principais mascaramentos:

1) Lógico

O mascaramento lógico (Fig. 14) vai ocorrer quando em algum nó do circuito ocorrer uma falha, porém a saída da porta lógica não vai depender da parte afetada. Sua saída pode ser determinada apenas pelas entradas não atingidas pela falha.

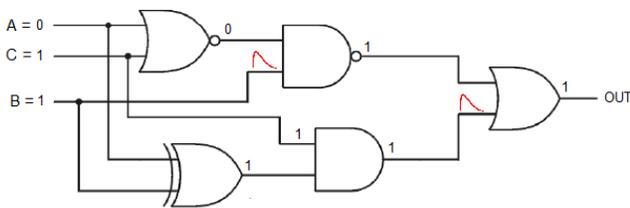


Fig. 14. Mascaramento Lógico

2) Elétrico

O mascaramento elétrico (Fig. 15) ocorre quando um pulso se propaga através das portas lógicas, mas é atenuado pelo caminho do circuito. Assim, ele é extinguido antes de ser armazenado por um elemento de memória.

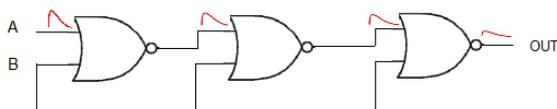


Fig. 15. Mascaramento Elétrico

3) Janela de Amostragem

Este mascaramento ocorre quando um pulso transiente se propaga através do circuito até um elemento de memória, porém durante a sua transição não ocorre transição do relógio, ou seja, o pulso atinge as linhas de dados fora da área da janela de amostragem. Assim, este pulso não chega a ser armazenado na memória (Fig. 16).

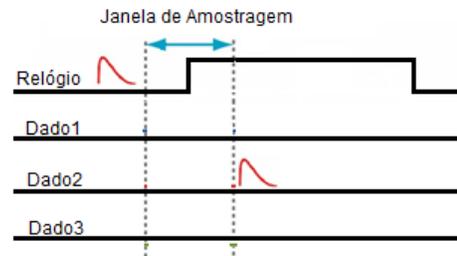


Fig. 16. Mascaramento por janela de amostragem

C. Técnicas de Tolerância a falhas

Técnicas de tolerância a falhas são responsáveis por garantir o funcionamento correto de um sistema mesmo com falhas. A maioria das técnicas de tolerância a falhas é baseada em redundância, exigindo componentes adicionais ou algoritmos especiais [16].

A aplicação de redundância para criar técnicas de tolerância a falhas pode ser de várias formas. A Redundância Modular Tripla (TMR) é um dos métodos mais tradicionais para inserir tolerância a falhas em projetos de hardware. O TMR (Fig. 17) consiste em triplicar o hardware fazendo com que a mesma função seja realizada em paralelo. Essa técnica gera entradas para um circuito de votação que retorna o voto da maioria observando a saída dos três módulos, resultando em apenas uma saída final. As desvantagens que essa técnica provoca são os aumentos da área ocupada e do consumo de potência [18].

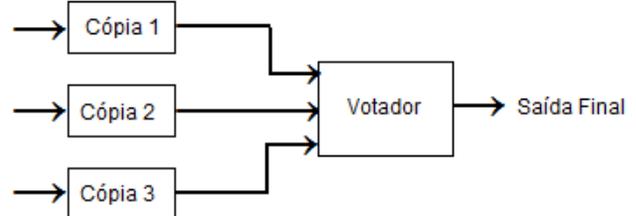


Fig. 17. Arquitetura da Redundância Modular Tripla (TMR)

O votador também é um componente do sistema e, portanto, está sujeito à ocorrência de falhas. No entanto, sem replicá-lo ele passa a ser um ponto único de falhas. A solução para contornar a fragilidade do votador também é triplicá-lo ou realizar a votação por *software*.

IV. FALHAS *STUCK-ON*

Dentre as consequências da redução da tecnologia está o aumento de componentes em uma mesma área. Quanto maior o número de componentes, maior será a probabilidade de um circuito apresentar falhas.

O objetivo dessa seção é apresentar o comportamento de uma das falhas permanentes que ocorre em transistores MOS, conhecida como *Stuck-On* [18]. Quando essa falha ocorre em um componente, este nunca mais voltará a funcionar corretamente.

O resultado de uma porta lógica com as falhas *Stuck-On* pode se propagar dentro de um circuito, fazendo com que a saída seja diferente do esperado. Por outro lado, em alguns casos, as falhas são mascaradas, não prejudicando a saída final do circuito. Os transistores de um circuito podem ser modelados como chaves e seu controle é feito através da

tensão aplicada no terminal de *gate*, que indicará o regime de operação do transistor, ou seja, se existe ou não um fluxo de corrente.

Um transistor com falha *Stuck-On* (SOnF) conduz permanentemente, independente do sinal aplicado ao terminal de porta. A fim de exemplificar o efeito causado pelas falhas *Stuck-On*, a Fig. 18 mostra a porta lógica NOR2 com sua tabela verdade correta e com falha. Neste exemplo, considera-se que a falha ocorreu no transistor NMOS N1 da rede *pull-down*.

Para entender o comportamento dessa falha, foi verificado cada estado da tabela verdade. Na porta lógica NOR2 abaixo, a SOnF está inserida em um transistor que faz parte da associação em paralelo. Os efeitos da falha se manifestarão se os dois transistores da rede *pull-up* estiverem conduzindo juntos, formando um caminho entre VDD e a saída do circuito.

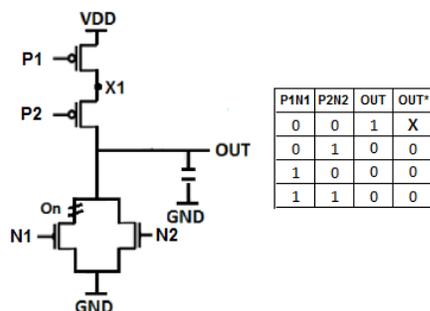


Fig. 18. Falha *Stuck-On* na porta lógica NOR2 com sua saída esperada (OUT) e sua saída obtida (OUT*)

Para o vetor $AB = 01$, o transistor PMOS P1 está desligado, fazendo com que a rede *pull-down* coloque na saída o valor lógico correto. De forma semelhante ocorre com o vetor $AB = 10$, a diferença é que o transistor PMOS P2 que está desligado. Para o vetor $AB = 11$, ambos os transistores da rede *pull-up* estão desligados, também gerando um resultado correto.

Entretanto, o vetor $AB = 00$ faz com que a falha seja percebida na saída do circuito. Com esse vetor, os dois transistores da rede *pull-up* estão ligados. Como a falha está no transistor NMOS N1, tanto a rede *pull-up* como a rede *pull-down* estão conduzindo juntas, gerando um curto-circuito. Quando um curto-circuito ocorre, provoca um consumo de potência excessivo e a saída poderá ser um estado lógico incorreto [4].

Para simular este tipo de falha é necessário forçar os transistores PMOS ou NMOS a permanecerem sempre fechados através do terminal *gate* (Fig. 19). Quando a falha for injetada nos transistores da rede *pull-up*, o terminal *gate* receberá o valor lógico zero. Porém, quando a simulação da falha estiver em transistores da rede *pull-down*, o valor lógico um deve ser atribuído ao terminal de controle de porta.

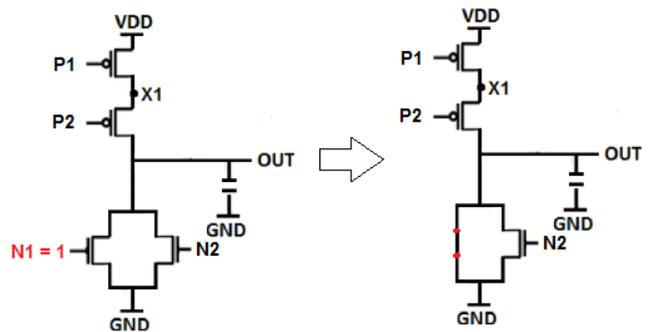


Fig. 19. Inserção de SOnF pelo terminal de controle de porta

O estado da tabela verdade que faz a saída apresentar um curto-circuito vai depender de qual transistor a falha está injetada. Na porta lógica NOR2, com a falha no transistor NMOS N1 ou no transistor NMOS N2, o estado que apresenta falha é o $AB=00$. Se a falha estivesse injetada no transistor PMOS P1 ou no PMOS P2, os vetores que poderiam gerar um curto-circuito seria $AB=10$ e $AB=01$, respectivamente.

Quando analisamos falhas, o ideal é sempre fazer duas simulações: uma para obter o resultado esperado e outra para ver o resultado obtido com a inserção da falha. A primeira simulação efetuada tem o intuito de validar o circuito logicamente, isto é, verificar se um circuito está cumprindo a sua função. Um circuito cumpre sua função enviando um resultado para a saída quando recebe determinada condição de entrada. Esta análise tem importância, pois se o circuito não estiver um funcionamento correto, qualquer outra simulação irá fornecer inconsistência nos dados.

A Fig. 20 mostra o resultado para a validação lógica da porta NOR2, onde a cada 200ns de simulação, o estado lógico da tabela verdade é modificado. A segunda simulação teve por objetivo mostrar o comportamento da mesma porta NOR2 quando é inserida uma falha no transistor NMOS N1, sendo o resultado obtido é mostrado na Fig. 21. Ambas as simulações foram realizadas na tecnologia preditiva de 45nm [19].

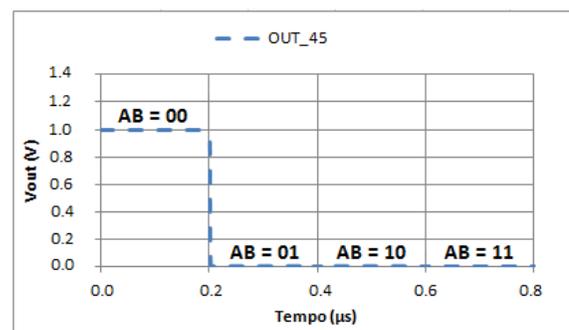


Fig. 20. Validação lógica para porta lógica NOR2 com SOnF

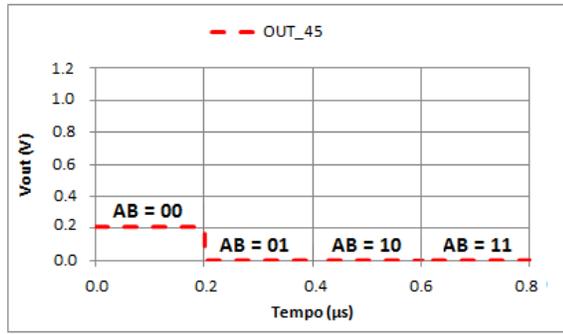


Fig. 21. Porta Lógica NOR2 com SOnF no transistor PMOS P1

Para testar falhas *Stuck-On* em circuitos CMOS é necessário pelo menos um vetor de teste. Este tipo de falha não gera atrasos, então independente da frequência analisada sempre terá a falha ou nunca terá. Os mascaramentos podem ocorrer, fazendo com que a falha não seja acusada na saída.

A definição de qual vetor de teste vai ser utilizado, vai depender de qual transistor do circuito a falha está injetada, podendo acontecer de uma mesma porta lógica ter vários vetores de teste diferentes.

O número total de vetores de teste que podem ser gerados (V_{Teste}) para uma porta lógica de E entradas é dado por (2).

$$V_{\text{Teste}} = 2^{(E)} \quad (2)$$

V. FALHAS *STUCK-ON* EM NANOTECNOLOGIAS

A evolução dos circuitos integrados só foi possível devido a redução das dimensões da tecnologia que permitiu a integração de mais transistores em um único *chip*. Com a redução da escala de fabricação de transistores às escalas nanométricas, muitas novas características passaram a influenciar o comportamento dos transistores.

A potência consumida vem crescendo de acordo com a redução da tecnologia e tem sido uma importante restrição de projeto [20].

Sendo assim, quando falamos em falhas *Stuck-On* em nanotecnologias, o principal efeito indesejado que ocorre nos circuitos é o consumo de potência, isto ocorre porque o comportamento da falha é manter o transistor falho sempre conduzindo (podendo gerar um curto-circuito).

VI. AVALIAÇÃO DE PORTAS LÓGICAS COM FALHAS *STUCK-ON*

Os experimentos realizados para avaliar as falhas *Stuck-On* consideraram quatro portas lógicas. As portas lógicas utilizadas foram a NAND de duas entradas (NAND2), NOR de duas entradas (NOR2), XOR de duas entradas (XOR2) e a AOI21, que podem ser vistas na Fig. 22. Cada uma das quatro portas lógicas foi analisada em quatro tecnologias preditivas: 45nm, 32nm, 22nm e 16nm, adotando modelos tecnológicos de *High Performance* (HP) [21].

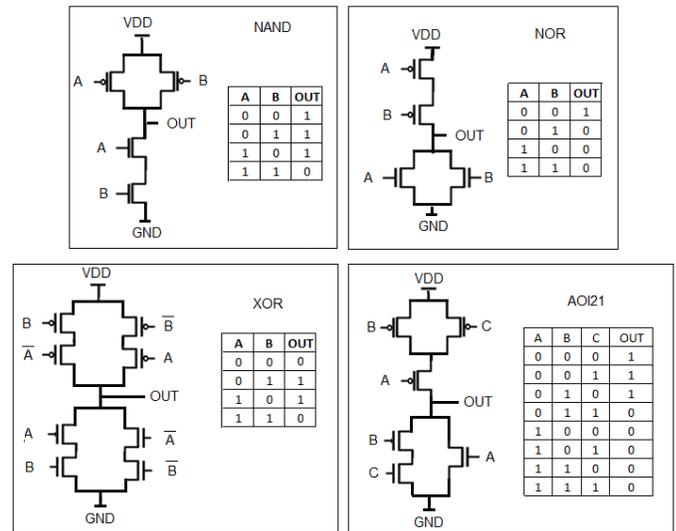


Fig. 22. Circuitos utilizados nos experimentos

A partir da descrição de cada uma das portas lógicas com suas respectivas tecnologias, foram feitos experimentos para avaliar o impacto do consumo de potência total e estático.

Para realizar as simulações foi escolhido o simulador elétrico NGSpice por oferecer suporte suficiente para realização dos experimentos desejados. O simulador elétrico NGSpice permitiu a configuração em relação ao dimensionamento, as ligações entre os transistores e dos valores de voltagem, possibilitando a construção das portas lógicas desejadas. Também permite realizar simulações com diferentes tecnologias de transistores, carregando um arquivo contendo as especificações do modelo a ser utilizado.

A potência total (3) foi calculada através da integral da corrente fornecida pela fonte de alimentação ao circuito, dividida pelo período de tempo analisado e após, multiplicado pela tensão da fonte de alimentação.

$$P_{\text{TOTAL}} = V_{\text{DD}} * \int_0^{\Delta t} \frac{i \cdot dt}{\Delta t} \quad (3)$$

A potência total foi medida junto com os atrasos para que fosse possível a análise de todos os arcos da função lógica utilizada.

A potência estática foi analisada para cada um dos estados da tabela verdade. A partir do momento em que todos os valores foram obtidos, é possível definir o maior valor, o menor valor e a média entre eles.

VII. RESULTADOS

Considerando as quatro portas lógicas mostradas na seção anterior, foram feitos experimentos medindo a potência estática e a potência total. Foram determinados os vetores robustos, podendo ocorrer de uma mesma porta lógica ter mais de um vetor robusto. Os vetores de teste robustos e qual transistor apresenta a falha nos experimentos realizados, são mostrados na Tab. 1.

Tab. 1. Vetores de teste robustos utilizados nos experimentos

Porta Lógica	Vetor Robusto	Rede	Transistor com falha
NOR2	00	<i>Pull-down</i>	A
NAND2	11	<i>Pull-up</i>	A
XOR2	00	<i>Pull-up</i>	A_NOT
AOI21	010	<i>Pull-down</i>	C

A potência estática e a potência total consumida pelos circuitos foram analisadas quando este teve a presença da falha e quando não teve. Os resultados obtidos são apresentados nas subseções abaixo.

A. Potência Estática

Na presença de falhas *Stuck-On*, a avaliação estática é importante porque haverá um transistor sempre ligado. Ficou comprovado nos experimentos que apenas o vetor de teste robusto, isto é, que acusa a falha na saída, que possui variações no consumo de potência estática.

A Tab. 2 mostra as quatro portas lógicas utilizadas juntamente com os valores de potência obtidos quando há uma falha *Stuck-On* no circuito e quando eles estão isentos da presença dela.

Como já era previsto, quando há presença da falha *Stuck-On* no circuito, a potência estática aumenta significativamente. A porcentagem desse aumento pode ser visto na coluna ΔP da Tab. 2. Esse aumento ocorre porque existe um transistor que fornece corrente continuamente, não importando o sinal aplicado em seu terminal de *gate*. Isso se torna um fator determinante no consumo total de potência dos circuitos.

Tab. 2. Resultados da potência estática

NOR2	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	14,90	3,96	276,26
22nm	4,96	6,50	1210,48
32nm	2,14	10,80	4946,32
45nm	1,29	17,50	13556,89
NAND2	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	38,60	3,73	934,85
22nm	12,00	5,10	135,29
32nm	3,46	9,95	187,57
45nm	1,94	15,10	678,35
XOR2	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	101,00	7,91	1176,86
22nm	31,20	12,30	1536,50
32nm	10,20	20,30	1890,19
45nm	7,63	30,11	3946,26
AOI21	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	20,00	2,03	985,22
22nm	6,31	3,13	2015,97
32nm	2,49	5,47	2196,78
45nm	1,63	8,85	5429,44

B. Potência Total

Os resultados para as quatro portas lógicas foram semelhantes para a potência total, isto é, com a inserção da falha, o circuito consome muito mais potência. Isto ocorre porque há um transistor sempre consumindo potência.

Para demonstrar o quanto a potência total aumenta em circuitos com falhas *Stuck-On*, a Tab. 3 mostra a diferença em percentual (ΔP) entre os valores de potencia total para os circuitos sem falhas e com falhas.

Quando o circuito não apresenta falha, o consumo de potência aumenta conforme a tecnologia aumenta. Com a inserção da falha, a potência também continua aumentando à medida que a tecnologia também aumenta.

Tab. 3. Resultados da potência total

NOR2	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	64,8	6,83	848,75
22nm	75,4	12,60	498,41
32nm	130,0	23,50	453,19
45nm	237,0	42,60	456,33
NAND2	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	103,0	4,24	2329,24
22nm	127,0	15,0	746,66
32nm	196,0	13,1	1396,18
45nm	388,0	21,5	1704,65
XOR2	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	298,0	6,76	4308,28
22nm	784,0	12,60	6122,22
32nm	774,0	22,20	3386,48
45nm	254,0	38,50	559,74
AOI21	Sem Falha (nW)	Com falha (μ W)	ΔP (%)
16nm	172,0	3,54	4758,75
22nm	256,0	6,53	3820,36
32nm	477,0	12,10	3842,14
45nm	883,0	21,80	3950,45

VIII. APLICANDO TMR PARA TOLERAR FALHAS *STUCK-ON*

Para os experimentos relativos ao TMR utilizamos as mesmas portas lógicas que estávamos analisando anteriormente: NOR2, NAND2, XOR2 e AOI21. Estes circuitos foram triplicados e ligados a um votador. Apenas uma das réplicas possuía a falha *Stuck-On* inserida no transistor indicado pela Tab. 1 para cada circuito.

O que podemos perceber na saída destes circuitos é que a técnica de TMR tolera 100% das falhas do tipo *Stuck-On*, desde que seja apenas uma das réplicas atingida pela falha. Os resultados referentes ao consumo de potência e ao desempenho dos circuitos sem a presença de falha são mostrados na Tab. 4 para a tecnologia de 16nm. Nas demais tecnologias, o comportamento de aumento na potência estática, potência total e nos atrasos é equivalente. Escolhemos mostrar a tecnologia de 16nm porque ela é a mais recente atualmente.

Tab. 4. Resultados aplicando o TMR sem a presença de falha

Portas Lógicas		Potência Estática (nW)	Potência Total (nW)	Atraso Médio (ps)	Número de Transistores
NOR2	Sem TMR	14,90	64,80	3,73	4
	Com TMR	79,29	348,59	28,26	26
	Comparação	5,32	5,37	7,57	6,50
NAND2	Sem TMR	38,60	103,00	4,92	4
	Com TMR	133,52	446,13	35,20	26
	Comparação	3,45	4,33	7,15	6,50
XOR2	Sem TMR	101,00	298,00	13,79	12
	Com TMR	306,09	1302,15	47,11	50
	Comparação	3,03	4,36	3,41	4,16
AOI21	Sem TMR	20,00	172,00	5,96	6
	Com TMR	86,10	991,50	33,98	32
	Comparação	4,30	5,70	5,70	5,33

A Tab. 4 mostra que quando a técnica de TMR é aplicada o consumo estático aumenta consideravelmente, impactando no consumo total que também sofre um grande aumento. Tanto na análise da potência estática quanto da potência total, foi verificado que os circuitos apresentam um aumento de consumo de no mínimo três vezes mais e no máximo de cinco vezes mais com a aplicação do TMR.

Quanto ao desempenho, quando utilizamos a técnica de TMR, os atrasos sofrem um aumento tornando o circuito mais lento comparado aos circuitos originais sem TMR, entretanto, garantem o comportamento lógico esperado do circuito.

A área de um circuito corresponde ao número de transistores existentes na descrição do circuito. Já que a potência está diretamente relacionada com a área, o circuito que tem a maior área terá também a maior potência. Isso pode ser observado na Tab. 4 com a porta lógica XOR2.

Na Tab. 5 é mostrado o consumo de potência quando uma das réplicas está com falha aplicando o TMR e também quando o circuito não está triplicado. O consumo estático e o consumo total aumentam ligeiramente, podendo ser considerados equivalentes ao consumo das portas sem TMR com falhas.

Tab. 5. Resultados aplicando o TMR com a presença de falha

Portas Lógicas		Potência Estática (μW)	Potência Total (μW)
NOR2	Sem TMR	3,96	6,83
	Com TMR	4,03	6,97
	Comparação	1,01	1,02
NAND2	Sem TMR	3,73	4,24
	Com TMR	3,84	4,57
	Comparação	1,02	1,07
XOR2	Sem TMR	7,91	6,76
	Com TMR	8,14	7,82
	Comparação	1,02	1,15
AOI21	Sem TMR	2,03	3,54
	Com TMR	2,10	4,32
	Comparação	1,03	1,22

Além disso, vale lembrar que o votador na técnica de TMR é um ponto único sensível a falha, e que se vier a falhar, o resultado poderá ser incorreto. Quando mais de uma réplica é atingida pela falha *Stuck-On*, o circuito apresentará falha na saída.

IX. CONCLUSÕES

Este trabalho teve por objetivo mostrar o comportamento de um tipo de falha que ocorre em transistores MOS em tecnologias nanométricas. Os resultados mostram que os circuitos na presença de falhas *Stuck-On* impactam significativamente o consumo de potência.

O consumo de potência sofre impacto devido a um transistor que fica conduzindo de forma permanente, sendo a principal característica para detecção da presença da falha *Stuck-On*.

Quando fizemos as análises considerando a técnica de TMR podemos perceber que quando a falha está em apenas uma das réplicas, os circuitos se apresentam 100% robustos. Esta

tolerância total a falhas únicas garantida pela TMR compensa os aumentos de potência e de atrasos inseridos pela triplicação dos circuitos, quando as aplicações projetadas exigem um alto grau de confiabilidade.

REFERÊNCIAS

- [1] G. E. Moore, "Cramming more components onto integrated circuits", 1965, pp. 114-117.
- [2] L. Anghel e M. Nicolaidis, "Defects tolerant logic gates for unreliable future nanotechnologies", Lecture Notes in computer science, v. 4507, Springer Berlin, Heidelberg, 2007, pp. 422-429.
- [3] D. T. Franco, "Du Signal des Circuits Logiques Combinatoires sous Fautes Simultanées Multiples", Doctoral Thesis, l'Ecole Nationale Supérieure des Télécommunications, França, 2008.
- [4] J. Mattos, L. Rosa Jr., M. Pilla, "Desafios e avanços em computação: o estado da arte", Pelotas, 2009, pp. 221-240.
- [5] P. F. Butzen e R. P. Ribas, "Leakage current in Sub-Micrometer CMOS Gates. In: Ricardo Reis. (Org.)", Advanced Topics on VLSI Design. Ied, Porto Alegre, Instituto de informática da UFRGS, 2009, p. 211-240.
- [6] N. H. Weste e D. M. Harris, "CMOS VLSI design a circuits and systems perspective", 4ª edição, Editora Pearson, 2011.
- [7] A. Neutzling, "Avaliação de circuitos somadores focando baixo consumo de potência", monografia, 2011, pp. 53-57.
- [8] I. E. Sutherland e R. F. Sproull, "Logical effort designing for speed on the back of an envelope", IEEE Advanced Research in VLSI, 1991, pp. 1-16.
- [9] S.R., Nassif "Design for variability in DSM technologies [deep submicron technologies]", ISQED.IEEE 2000 First Int. Symp. on Quality Electronic Design., pp.451-454, 2000.
- [10] Y. Cao and L.T. Clark. Mapping Statistical Process Variations Toward Circuit Performance Variability: An Analytical Modeling Approach. In Proc. of DAC, pages 658-663, 2005.
- [11] C. Meinhardt e R. Reis, "Evaluation of Process Variability on Current for Nanotechnologies Devices", IEEE 3rd Latin American Symposium on Circuits and Systems (LASCAS), 2012.
- [12] S.K. Saha. Modeling Process Variability in Scaled CMOS Technology. IEEE Design and Test of Computers, Vol.27:pp.8-16, March/April 2010.
- [13] S. R. Nassif, "Process variability at the 65nm node and beyond., IEEE Custom Integrated Circuits Conference 2008. San Jose, USA, pp 1-8.
- [14] Y. Cao, J. Tschanz e P. Bose, "Guest Editors' Introduction: Reliability Challenges in Nano-CMOS Design", Design for Reliability at 32nm and Beyond, IEEE Design & Test of Computers, 2009.
- [15] L. Chang, Y. Choi, D. Ha, P. Ranade, S. Xiong, J. Bokor, C. Hu e T. King, "Extremely Scaled Silicon nano-CMOS devices", Proceedings of the IEEE, Vol. 91, Issue 11, pp. 1860-1873, 2003.
- [16] T. S. Weber, "Um roteiro para exploração dos conceitos básicos de tolerância a falhas", Textos didáticos em tolerância a falhas, Porto Alegre, 2002.
- [17] B. W. Johnson, "An introduction to the design and analysis of fault-tolerant system", Prentice Hall, New Jersey, 1996.
- [18] C. Metra, M. Favalli e B. Ricco, "Compact and low power on-line self-testing voting scheme", Workshop on defect and fault-tolerance in VLSI systems, 1997.
- [19] Y. Cao, "New paradigm of predictive MOSFET and interconnect modeling for early circuit simulation", Custom Integrated Circuits Conference, EUA, 2000.
- [20] L. Benini, G. Micheli e E. Macii, "Desingning Low-Power circuits: practical recipes", IEEE Circuits and Systems Magazine, Vol. 1, Number 1, First Quarter, 2001.
- [21] PTM, Predictive Technology Model. [Online; acessado em 15 de julho], <http://ptm.asu.edu/>.