Projeto de um Bloco de Memória SRAM em Tecnologias CMOS Nanométricas de 16nm

Iuri A. C. Gomes, Cristina Meinhardt, Paulo F. Butzen

***Abstract*—Memórias SRAM são fundamentais na hierarquia de memória dos computadores atuais. A miniaturização da tecnologia impacta não somente no desempenho como na confiabilidade dessa arquitetura de memória, aumentando a complexidade de seus projetos. Este trabalho tem como objetivos o projeto, validação, análise e comparação de dois blocos de memória SRAM. Um dos blocos será projetado utilizando transistores do tipo *High Performance* (HP) e o outro utilizando transistores do tipo *Low Power* (LP), ambos pertencentes a uma tecnologia preditiva CMOS de 16nm. Cada bloco de memória é composto por quatro circuitos principais: célula de memória, circuito de escrita, circuito de pré-carga e circuito de leitura. Conceitos de dimensionamento, características elétricas e estabilidade são explorados. O trabalho analisa os resultados referentes a desempenho e robustez. A comparação entre os dois blocos mostra que o bloco HP, apesar de ser mais veloz, é menos robusto que o bloco LP.**

***Palavras-chaves* — CMOS, Memória SRAM, projeto, tecnologia nanométrica**

I. INTRODUÇÃO

Computadores pessoais, *notebooks*, celulares, vídeo *games* portáteis, GPS, impressoras, *scanners*, *tablets* e *smartphones* são apenas alguns exemplos na longa lista de inovações tecnológicas modernas. Estas inovações são projetadas em uma tecnologia denominada CMOS (*Complementary Metal-Oxide-Semiconductor*). Grande parte destas inovações são possíveis devido aos avanços alcançados no contínuo desenvolvimento desta tecnologia. A adoção de novos materiais, novas estruturas e, principalmente, a redução das dimensões dos elementos projetados na tecnologia CMOS são a estratégia da indústria de semicondutores para produzir circuitos menores e mais rápidos, possibilitando a criação de produtos com melhor desempenho, menor tamanho e com maior interação e usabilidade com o usuário.

Grande parte das inovações tecnológicas listadas anteriormente é derivada do tradicional computador. Pode-se definir um computador como a organização de um conjunto de dispositivos capazes de armazenar e processar informações. Computadores de propósitos gerais, baseados na arquitetura definida por John Von Neumann [1], possuem quatro componentes principais: a Unidade Lógica Aritmética (ULA), responsável por operações lógicas e aritméticas; a Unidade de Controle, responsável por gerar os sinais de controle e temporização; os Dispositivos de E/S, para entrada e saída de dados; e a Memória Principal, usada para armazenar dados e instruções.

Computadores utilizam diferentes tipos de memória, organizando o sistema de memória de forma hierárquica, de forma a manter as memórias mais velozes próximas ao processador. Dentre os tipos de memória disponíveis, a memória SRAM (*Static Random Access Memory*) destaca-se por estar entre as mais rápidas, sendo altamente utilizada para construção das memórias caches de computadores. Com a redução das dimensões dos dispositivos e, consequente, o aumento da capacidade de integração de transistores em um circuito integrado (CI), a área que os níveis de memória cache ocupam em um processador vem aumentando.

A miniaturização da tecnologia traz diversos benefícios, dentre os quais se pode citar a fabricação de CIs (Circuitos Integrados) menores e com um maior número de funcionalidades. Isso só é possível graças à redução do tamanho do transistor e consequente aumento da densidade desses dispositivos. Outra vantagem é o menor consumo de potência e a maior velocidade dos dispositivos MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*) nanométricos. Atualmente, circuitos integrados já são fabricados em tecnologias menores que 45nm [8][10]-[12], sendo que o tamanho da tecnologia em que um transistor é fabricado é dado pelo comprimento do canal do transistor.

No entanto, a redução na tecnologia vem cada vez mais apresentando novos desafios aos projetistas de circuitos integrados, principalmente quando se trata do projeto de circuitos de memória SRAM [2]. Pela característica essencial que a SRAM ocupa na hierarquia de memória é necessário que esse tipo de memória funcione corretamente nas novas

Iuri A. C. Gomes é graduado em Engenharia de Computação pela Universidade Federal do Rio Grande (Furg) e mestrando em Microeletrônica na Universidade Federal do Rio Grande do Sul (PGMicro/UFRGS). (e-mail: iualbandes@hotmail.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 e mestre em Ciência da Computação pela Universidade Federal do Rio Grande Sul (UFRGS). Atualmente é doutorando do Programa de Pós-graduação em Microeletrônica da Universidade Federal do Rio Grande do Sul (UFRGS) atua como professora no Centro de Ciências Computacionais da Universidade Federal do Rio Grande (C3/Furg) (e-mail: paulobutzen@furg.br)

tecnologias, já que a miniaturização tecnológica causa impacto no desempenho e na confiabilidade dessa arquitetura de memória.

Uma célula de memória SRAM possui três operações básicas ligadas ao seu funcionamento: guardar um valor (*hold*), ler um valor contido na célula (*read*) e escrever um valor na célula (*write*). Além destas, existe uma função muitas vezes abstraída, mas essencial para a SRAM: a operação de pré-carga.

Para o funcionamento correto de um banco de memória SRAM é necessário o projeto de alguns circuitos essenciais. O modo como eles são organizados definem a arquitetura de memória. A arquitetura de memória é definida por seis circuitos: célula de memória, circuito de escrita (*Write Driver*), circuito de leitura (*Sense Amplifier* - SAE), decodificador de coluna (*Column decoder*), decodificador de linha (*Row decoder*), e circuito de pré-carga (*Pre-charge*). Um bloco de memória SRAM é formado por quatro destes circuitos: o circuito de leitura, o circuito de escrita, o circuitos de pré-carga e a célula de memória. Um bloco está diretamente relacionado às funções e operações da arquitetura de memória descrita, onde cada um dos circuitos possui relação direta com uma ou mais destas operações e funções.

Os circuitos decodificador de coluna e decodificador de linha são necessários para endereçar corretamente um determinado bit do banco de memória. Na análise e projeto de células de memória SRAM, estes blocos podem ser omitidos por não serem essenciais para o funcionamento das operações de leitura, escrita e armazenamento de uma palavra.

Atualmente diversos esforços têm sido realizados para o projeto de memórias SRAM em escalas nanométricas. Em [9][15] são apresentadas otimizações da célula de memória e outras técnicas de projeto para SRAMs em tecnologias nanoescalares. A utilização de duas alimentações, uma para o circuito propriamente dito, e outra para as células de memória têm se mostrado uma alternativa para o aumento da robustez da SRAM e também para a maximização da relação entre desempenho e consumo de potência da memória [14][15]. Alguns trabalhos exploram a adoção de transistores *multigates* e seu impacto constatado principalmente na estabilidade e no consumo de potência da memória [10][13]. No que se refere ao consumo de potência, os trabalhos concentram esforços na redução das correntes de fuga existentes nas tecnologias nanométricas [16]. Técnicas para lidar com o impacto de variações de processo de fabricação foram apresentadas para tecnologias de 32nm e 22nm [10][11].

O objetivo principal deste trabalho é o projeto de blocos de memória SRAM em tecnologia nanométrica, explorando dois modelos de transistores preditivos para a tecnologia de 16nm. Os tipos de transistores adotados são o *Low Power* (LP) e o *High Performance* (HP). Este trabalho apresenta as decisões de projeto adotadas para a criação dos dois blocos de memória do tipo SRAM, cada um utilizando um tipo de transistor. Também apresenta a comparação dos dois projetos quanto ao desempenho das operações de escrita e leitura e quanto à robustez (SNM – *Signal Noise Marging*) durante as operações de *hold* e leitura.

O projeto de cada bloco de memória envolve o desenvolvimento dos quatro circuitos principais: célula de memória, circuito de escrita, circuito de leitura e circuito de pré-carga. A estrutura do bloco proposto e os sinais de controle necessários para as operações estão ilustrados na Fig.1. O projeto considerou 256 células por coluna no bloco de memória. As 256 células de memória foram implementadas através de uma célula real SRAM de 6 transistores e simulação das demais 255 células de memória através de duas células de memória de grande dimensionamento a fim de gerar as capacitâncias e correntes de fuga de proporções adequadas.

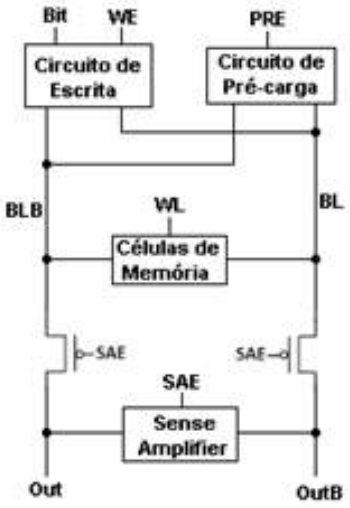

Fig. 1: Bloco de memória proposto e sinais de controle

A Seção II apresenta os conceitos de projeto de uma célula SRAM. A Seção III descreve o projeto dos blocos de memória SRAM, ou seja, a metodologia abordada no trabalho explicitando características do projeto como dimensionamento, tensão nominal, características dos circuitos utilizados, restrições de projeto, métodos para validar a arquitetura e cálculos para avaliar o desempenho. Na Seção IV são apresentados os resultados de validação, desempenho e estabilidade fazendo comparações entre as arquiteturas e operações. Finalmente, a Seção V apresenta as conclusões observadas no projeto.

II. Célula de Memória SRAM: Circuito, Funcionamento e Estabilidade

As SRAMs são o tipo mais comum de memória utilizada nos circuitos integrados atualmente [3]. Existem diversas propostas de circuitos para uma célula de memória SRAM. Cada uma delas é normalmente nomeada de acordo com o número de transistores que formam o circuito.

A. Célula de Memória SRAM de seis transistores

A célula de SRAM é a componente chave para o armazenamento da informação de um bit. A SRAM 6T, apresentada na Fig. 2, é o tipo mais comum de célula de

memória [3]. Como o nome indica, ela é formada por seis transistores. Por ser a arquitetura mais empregada, ser robusta e simples no arranjo dos transistores, esta será a célula de SRAM adotada neste trabalho.

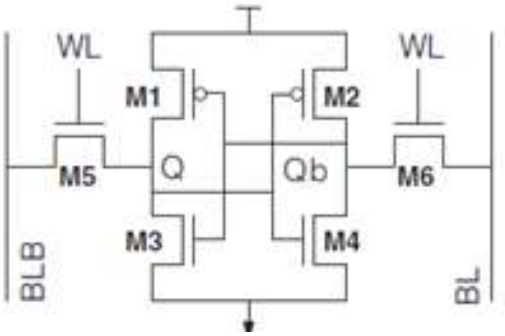

Fig. 2: A arquitetura SRAM 6T [3]

No arranjo dos transistores da SRAM 6T, os quatro transistores (M1 – M4) criam dois inversores CMOS cruzados da célula e os transistores M5 e M6, ambos do tipo NMOS (*N-type MOSFET*), definem o momento em que a célula pode ou não ser acessada por alguma operação. Durante as operações de leitura e escrita, ocorre a ativação do sinal *wordline* (WL) e os transistores de acesso conectam os nodos internos Q e Qb da célula à *bitline* (BL) e à *bitline* complementar (BLB). Ao desativar o sinal WL, os transistores de acesso se tornam responsáveis pelo isolamento da célula durante a operação de espera (*hold*).

Uma SRAM é projetada de modo a prover um acesso de leitura não destrutiva, uma operação capaz de trocar o valor na célula de memória e o armazenamento de dados enquanto a arquitetura esteja sendo alimentada. A seguir são apresentadas as características associadas a cada operação da SRAM.

1) *Operação de leitura*

Como o próprio nome indica, uma operação de leitura tem como objetivo ler o valor de *bit* contido em uma célula de memória sem que o valor guardado seja alterado. Normalmente três sinais de controle, ilustrados na Fig. 1, estão envolvidos nesta operação:

- WL (*wordline*): para permitir acesso a célula
- SAE (*sense-amplifier enable*): para ligar o circuito que percebe a diferença gerada nas *bitlines* durante a operação
- PRE (*pré-carga*)*:* para retornar o valor das *bitlines* para a tensão de alimentação após a operação.

Neste trabalho, o método considerado para se obter o valor do bit contido na célula consiste nos seguintes passos:

i) Pré-carregar as *bitlines* até a tensão de alimentação nominal (V_{dd}) através do circuito de pré-carga (sinal PRE = '0'). Após as *bitlines* estarem carregadas, o próprio circuito de pré-carga (sinal PRE = '1') desconecta a alimentação destas.

ii) Desativar a pré-carga e ativar o sinal WL. Isso conecta os nodos Q e Qb, internos da SRAM nas *bitlines*, fazendo com que a *bitline* conectada ao nodo com valor lógico zero tenha uma queda na tensão, enquanto a outra *bitline*, conectada ao nodo com valor lógico alto, continue com a tensão próxima de V_{dd}.

iii) Esperar que a diferença de tensão entre as *bitlines* BL e BLB, $\boldsymbol{V_{bl}}$ e $\boldsymbol{V_{blb}}$ respectivamente, atinja um valor que o *sense-amplifier* consiga detectar. Neste momento, o sinal de controle SAE é ativado e o *sense-amplifier* amplifica a diferença entre as tensões $\boldsymbol{V_{bl}}$ e $\boldsymbol{V_{blb}}$ para um valor digital, ou seja, uma das tensões passa para '1' lógico e outra para o '0' lógico.

Alguns estudos mostram que uma tensão no valor de $V_{dd}/2$ também pode ser usada como valor de pré-carga [4][5]. Por vezes, não é necessário que as *bitlines* sejam completamente carregadas, todavia, é indispensável que elas estejam equalizadas em certo valor [4].

A ativação do SAE faz com que o circuito de escrita se isole do resto dos circuitos e comece a amplificar essa pequena diferença de voltagem em um valor alto ou baixo, conforme o armazenado na célula de memória lida. Simultaneamente à ativação do SAE, os sinais WL e PRE voltam para suas situações iniciais. A mudança nos sinais WL e PRE antes da leitura terminar permite que não se perca tanto tempo com a recarga das *bitlines*, já que é possível manter a pré-carga ligada sem afetar o funcionamento da operação, visto que o circuito de leitura está isolado e funcionando. A Fig. 3 elucida o funcionamento dos sinais de controle e o impacto da operação nas *bitlines*.

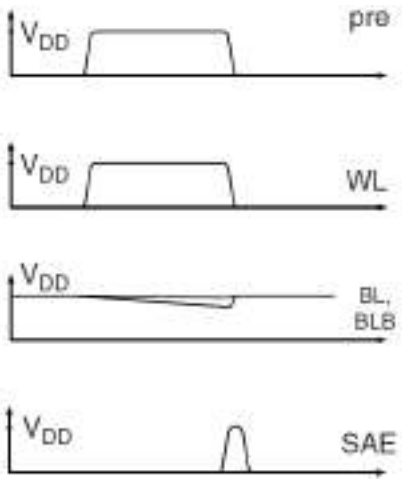

Fig. 3: Temporização dos sinais durante a operação de leitura [6]

2) *Operação de escrita*

O procedimento de escrita na memória SRAM 6T é ilustrado na Fig. 4 e funciona da seguinte forma:

i) A pré-carga é desligada (PRE = '1') e o circuito de escrita define os valores das *bitlines* de acordo com o bit que se deseja gravar. No exemplo da Fig. 4, BLB mantêm valor lógico '1' oriundo da pré-carga enquanto BL é descarregada para assumir o valor lógico '0'.

ii) A seguir o acesso aos nodos internos da célula de memória é permitido ativando o sinal WL. Como as *bitlines* estão com sinais complementares definidos pelo circuito de escrita, este valor é armazenado na célula de memória.

iii) Após efetivado o armazenamento, o sinal WL é desativado, impedindo qualquer mudança do valor armazenado, e o circuito de escrita passa ao circuito de pré-carga o controle do valor das *bitlines*.

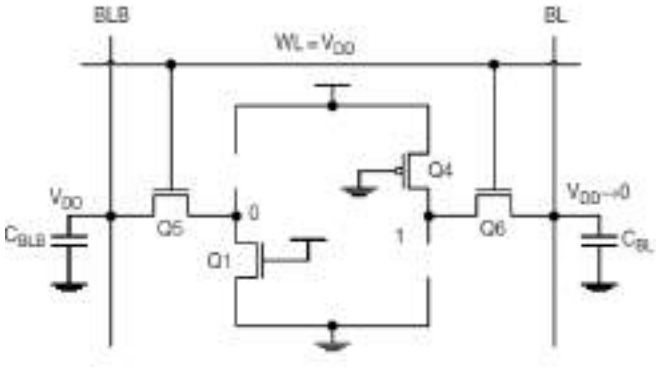

Fig. 4: Modelo simplificado de uma célula SRAM 6T durante a operação de escrita [6]

3) *Estabilidade da SRAM: SNM – Static Noise-Margin*

A confiabilidade em uma memória é quantificada pelo que chamamos de margem de ruído estático, ou simplesmente SNM. A SNM é o máximo valor de ruído, medido em volts, que um dispositivo tolera sem que suas operações sejam afetadas. No caso da SRAM, a SNM é o valor máximo de ruído que a célula suporta sem que ocorra a mudança do valor armazenado (também conhecido como *flip*).

O cálculo da SNM na SRAM é feito utilizando um método gráfico que envolve o espelhamento da curva de transferência de voltagem dos inversores que formam a célula de seis transistores [7]. Além do método de quantificação da SNM, também será analisado o quanto o dimensionamento da célula afeta essa margem de ruído e de que forma é possível melhorar a estabilidade quando esta for considerada no projeto da SRAM.

a) *Método Gráfico para o Cálculo de SNM*

A SNM da SRAM é definida como o máximo valor de ruído elétrico que a célula de seis transistores tolera sem que ocorra a troca de estado entre os inversores [7]. Os ruídos elétricos podem ser motivados por inúmeros fatores como, por exemplo, raios, interferências de rádio freqüência e radiação. O método para cálculo da SNM foi primeiramente descrito por Hill em 1968 no artigo *Definitions of noise margin in logic systems* [7]. Uma vantagem importante desse método é que ele pode ser automatizado utilizando simulador elétrico através de uma simulação quiescente (DC).

A estabilidade e robustez de uma dada célula de memória são analisadas durante suas operações de leitura e espera. Nesse método gráfico são utilizadas as curvas de transferência de voltagem dos inversores. O valor da SNM é definido pelo lado do máximo quadrado desenhado entre a curva de transferência de voltagem (VTC) e a VTC espelhada. A composição das duas curvas é chama de curva borboleta (*butterfly curve*) e está ilustrada na Fig. 5.

b) *SNM: Dependências*

Parâmetros como tensão de alimentação, tensões nas *bitlines* e na *wordline* e dimensionamento dos transistores, causam impacto nas estabilidades de uma SRAM. A seguir é discutido como o dimensionamento pode aumentar a robustez da célula de memória. Como neste trabalho será explorado o dimensionamento dos transistores da célula de memória para aumentar a robustez da mesma, a seguir é discutido como este dimensionamento deve ser realizado. Para isso são utilizadas duas razões, **α** e **β,** definidas de acordo com as equações (1) e (2), onde W_{nmos} é a largura utilizada nos transistores tipo-n dos inversores da célula de memória, W_{acs} é a largura dos transistores NMOS de acesso, e W_{pmos} é a largura utilizada nos transistores tipo-p dos inversores da célula de memória.

$$\alpha = \frac{W_{nmos}}{W_{acs}} \quad (1)$$

$$\beta = \frac{W_{pmos}}{W_{acs}} \quad (2)$$

Para manter a área da célula com valores aceitáveis, os valores de **α** e **β** são restritos entre 1 e 2.5. A Fig. 6 ilustra as variações na SNM de acordo com os parâmetros **α** e **β**. A partir da Fig. 6 observa-se que o dimensionamento tem muita importância quanto à robustez da célula na operação de leitura. Na operação de espera quase não existe diferença. Outra característica que se observa é que a SNM tem uma melhora maior quando se aumenta **α** do que quando se aumenta **β**. Apesar da conclusão que aumentar **α** e **β** melhore a SNM, deve se levar em conta que isso gera impactos em outros parâmetros como, por exemplo, o aumento na área ocupada pela memória e no consumo estático [5].

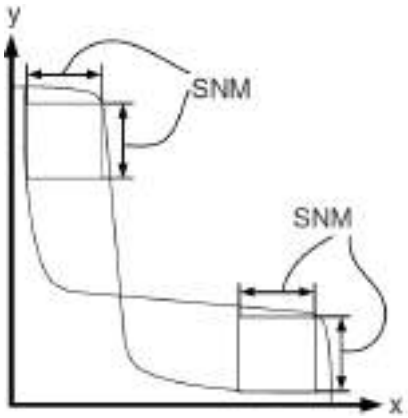

Fig. 5: Curva borboleta usada para estimativa de SNM baseada no método do máximo quadrado na operação de leitura[6].

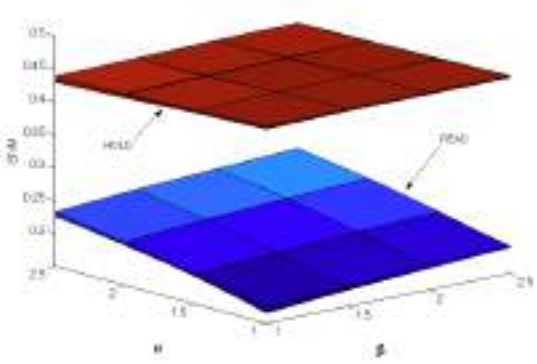

Fig. 6: Variação na SNM em volts aplicando o dimensionamento dos transistores. [5]

B. Circuitos do Bloco de Memória

1) Célula de Memória

A célula de memória utilizada neste trabalho é a SRAM 6T ilustrada na Fig. 2 (a). Suas principais características foram descritas na seção anterior. Para alcançar uma boa margem de ruído SNM, o tamanho dos transistores que fazem parte da célula de memória foram definidos com larguras W_{acs} = 40nm, W_{pmos} = 60nm e W_{nmos} = 80nm, conforme análise apresentada na seção II.3.b.

2) Circuito de Escrita (Write Driver)

Como a célula de memória SRAM já foi discutida e apresentada na seção II, aquelas informações não serão repetidas nesta seção

O circuito de escrita tem como função carregar ou descarregar as *bitlines* de forma que seja escrito o bit desejado na célula de memória. O esquemático do circuito de escrita projetado neste trabalho é apresentado na Fig. 7.

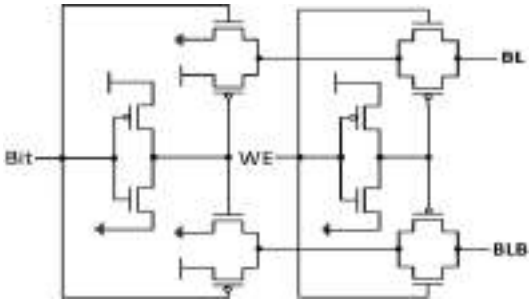

Fig. 7: Circuito de escrita

Este circuito possui dois sinais de entrada, o sinal *Bit* representando o valor a ser escrito na célula de memória, e o sinal de controle *write enable* (WE). O sinal de controle WE tem como função permitir, ou não, o acesso do circuito de escrita às *bitlines*. Quando WE está ligado, o circuito de escrita impõe nas *bitlines* os valores de tensão necessários para a escrita do *bit* na célula de memória. Como este circuito tem a função de carregar as *bitlines* com o valor do bit a ser armazenado, os transistores desse circuito devem ser projetados para terem a capacidade de carregar toda a capacitância associada às *bitlines*. Neste sentido, os transistores deste circuito foram definidos com uma largura W = 1 µm.

3) Circuito de Pré-carga das Bitlines

O circuito de pré-carga das *bitlines* tem grande importância para correto funcionamento da arquitetura proposta. Durante a operação de escrita, uma das *bitlines* é descarregada, gerando uma diferença entre as *bitlines* após o termino da operação. Da mesma forma, na operação de leitura, uma das *bitlines* também é descarregada. Essa diferença não deve existir em uma próxima operação de leitura. Portanto se faz necessário a recarga das *bitlines* após as operações. A Fig. 8 representa o circuito utilizado para a equalização e pré-carga das *bitlines*. Da mesma forma que o circuito de escrita, os transistores foram definidos com largura W = 1µm para terem capacidade de carregar rapidamente a *bitlines*.

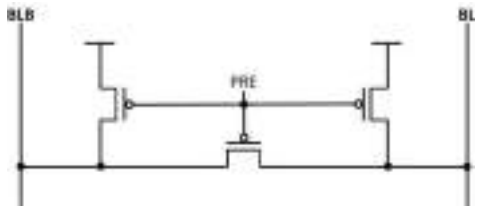

Fig. 8: Circuito de pré-carga e equalização

4) Circuito de Leitura (Sense Amplifier)

O *sense-amplifier* (SA) é um circuito importante na arquitetura da SRAM, pois ele define boa parte do desempenho da operação de leitura. O circuito de leitura mostrado na Fig. 9 é um dos tipos mais comuns, também conhecido na literatura como *SA* do tipo *latch* [6] e é o circuito adotado neste projeto. Por ser essencial ao desempenho da operação de escrita, e considerando que as *bitlines* são isoladas da saída conforme mostrado na Fig. 1, seus transistores forem definidos com largura W = 500 nm.

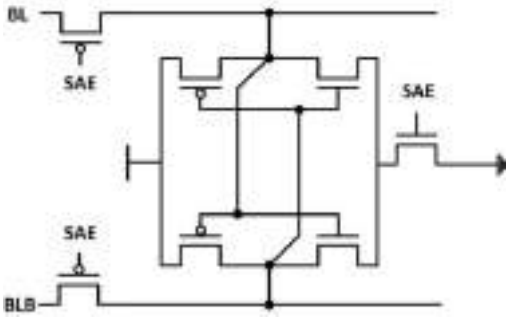

Fig. 9: Sense Amplifier do tipo latch.

C. Validação dos Blocos

Esta seção explica como os circuitos foram validados e apresenta as abstrações utilizadas. Normalmente, memórias SRAM são organizadas no formato de linhas e colunas, como

mostrado na Fig. 10. Se considerarmos somente uma célula de memória, estaríamos simplificando o projeto e desconsiderando fatores importantes que impactam o desempenho individual de cada célula de memória. Normalmente, 256 células de memória são dispostas na mesma coluna, compartilhando as *bitlines* e os circuitos de leitura, escrita e pré-carga. Esta será a quantidade de célula de memória utilizadas no projeto do bloco proposto.

Considerando que dados extraídos de uma célula de memória são muito semelhantes ,independente da célula escolhida na coluna, apenas uma célula será avaliada e as demais 255 serão substituídas por células de memória com tamanho equivalente a 128 e 127 células. A escolha por utilizar duas células de memória de tamanho equivalente a 128 e 127, ao invés de uma única com tamanho equivalente a 255, deve-se ao fato de considerar a probabilidade de metade delas estarem armazenando o valor lógico '0' e a outra metade o valor lógico '1'.

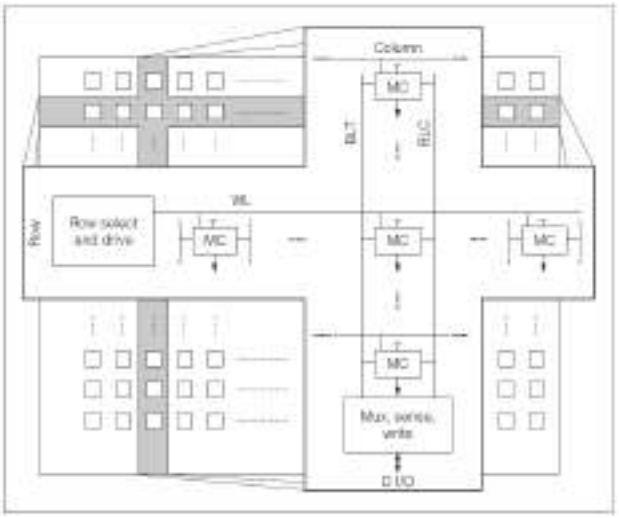

Fig. 10: Arquitetura de um bloco de memória SRAM

1) *Validação da Célula de Memória*

Para validar este circuito devemos analisar duas operações, a escrita e a espera. A Fig. 11 apresenta o comportamento esperado. No instante inicial o nodo Q contém o valor lógico '1', e seu nodo complementar Qb valor lógico '0'. Desejamos que a escrita do valor '0' lógico seja feita. Para isso, deixamos a BL com '0' lógico e a BLB com valor '1' lógico. Após as *bitlines* alcançarem o valor desejado, ligamos o sinal *wordline* (WL = '1'), conectando a célula de memória as *bitlines*. A análise das formas de onda dos sinais Q, Qb e WL permite identificar se a escrita ocorreu com sucesso, verificando que o valor no nodo Q passou de '1' lógico para '0' lógico. Logo após a mudança no nodo Q, desligam-se os transistores de acesso e espera-se um tempo para verificar a operação de espera. O bloco de escrita terá sido validado ao se verificar que o nodo Q não muda de valor durante a operação de espera.

Uma característica importante para os testes da SRAM usada é que os nodos Q e Qb são virtualmente idênticos já que ambos são conectados a inversores e transistores de mesma função e dimensionamento. Por essa razão pode-se analisar os nodos Q e Qb em uma operação de escrita para um valor e afirmar que a escrita do valor complementar também funcionará.

2) *Validação do Circuito de Escrita (Write Driver)*

Para validar o circuito, foram realizados dois testes. Primeiramente, foi testado se o circuito consegue impor os valores corretos nas *bitlines*. Depois, foi verificado se o circuito de escrita consegue modificar o valor dentro da célula de memória.

A Fig. 12 é referente aos testes de carga e descarga das *bitlines*. O gráfico demonstra a queda de tensão nas *bitlines* esperada durante o processo de escrita. Já a Fig. 13 é referente ao teste de escrita, ou seja, ele analisa se o circuito de escrita consegue trocar o valor na célula de memória.

3) *Validação do Circuito de Pré-carga*

Neste caso testa-se a capacidade de carregamento e equalização das *bitlines* após uma operação de escrita. A Fig. 14 representa o comportamento esperado após uma operação de escrita. A operação de escrita inicia no momento em que o sinal WE passa para o '1' lógico, e juntamente a pré-carga é desligada (PRE = '1'). Durante o tempo em que WE está ligado, uma das *bitlines* passa a descarregar rapidamente. Quando a operação de escrita termina, WE passa para '0' e a pré-carga retorna a operar levando as *bitlines* para a tensão nominal do circuito.

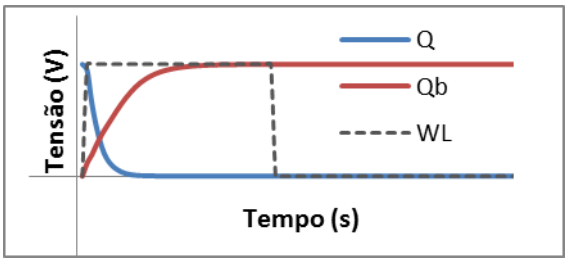

Fig. 11: Comportamento esperado para validação da célula de memória. Operação de escrita e espera.

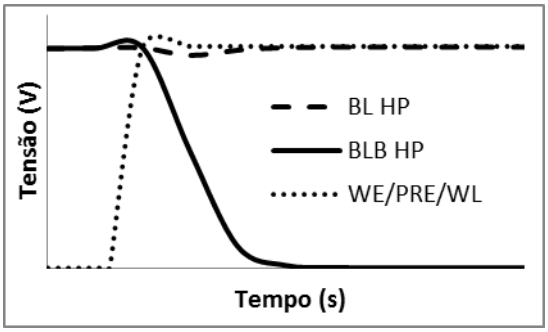

Fig. 12: Comportamento esperado nas bitlines durante a operação de escrita.

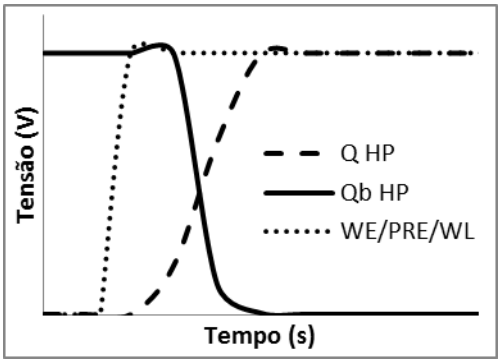

Fig. 13: Comportamento esperado no nodos da célula de memória durante a operação de escrita.

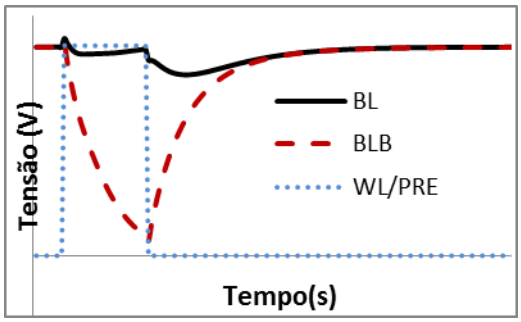

Fig. 14: Teste de pré-carga e equalização após escrita

4) Validação do Circuito de Leitura (Sense-Amplifier)

São necessários dois testes para validar o circuito de leitura. Primeiramente, é verificado se o circuito destrói o bit guardado na célula durante a leitura. No segundo teste, é analisado se o SA, quando ligado (SAE = '1'), consegue amplificar a diferença de tensão gerada no inicio da leitura entre as *bitlines*.

Os gráficos na Fig. 15 e na Fig. 16 definem comportamentos esperados durante os testes para a validação do circuito. No início da operação de leitura WL passa para '1' lógico e os nodos da célula de memória conectam com as *bitlines*. A conexão entre a *bitline* e o nodo com sinal baixo causa uma pequena descarga na *bitline*. Esta pequena diferença entre as *bitlines* será amplificada pelo *Sense Amplifier*..

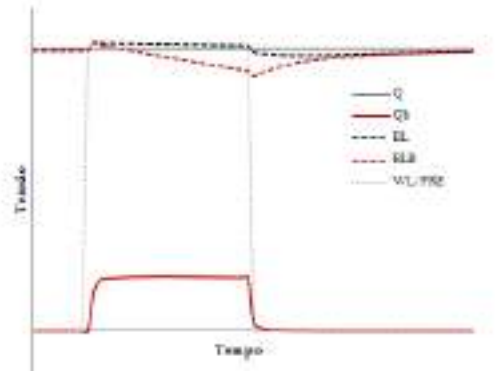

Fig. 15: Primeiro comportamento esperado para o teste do circuito de leitura. Descarga pequena em uma das bitlines sem trocar valor nos nodos Q e Qb.

O segundo gráfico, apresentado na Fig. 16, analisa a capacidade do SA de amplificar a diferença de tensão entre BL e BLB. No momento que o sinal SAE passa para '1' lógico, o circuito de leitura isola-se juntamente com os nodos Out e OutB. A tensão V_{dif} é amplificada para valores digitais, representando em Out o valor lido, e em OutB o valor complementar contido na célula.

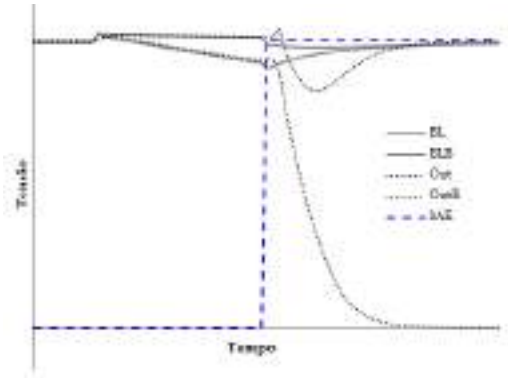

Fig. 16:Segundo comportamento esperado para o teste do circuito de leitura. Amplificação de sinal entre Out e OutB

D. Integração dos Circuitos e Validação

O último passo é integrar todos os circuitos propostos no decorrer deste trabalho. Para a integração do bloco *High Performance* (HP) os quatro circuitos, célula de memória 6T, circuito de escrita, circuito de pré-carga e circuito de leitura, foram projetados utilizando transistores *High Performance* na tecnologia de 16nm.

Para a integração do bloco *Low Power* (LP) os mesmos quatro circuitos foram projetados utilizando transistores *Low Power* na tecnologia de 16nm.

A Fig. 17 apresenta o circuito resultante da integração dos circuitos de leitura, escrita, pré-carga e células de memória. Esta figura detalha a simulação das 256 células de memória. A célula zero será a célula a ser avaliada e as demais 255 serão substituídas por células de memória com tamanho equivalente a 128 (1-128) e 127 (129-255) células. A escolha por utilizar duas células de memória de tamanho equivalente a 128 e 127, ao invés de uma única com tamanho equivalente a 255, deve-se ao fato de considerar a probabilidade de metade delas estarem armazenando o valor lógico '0' e a outra metade o valor lógico '1'.

Para validar o bloco foram realizadas as seguintes operações:

i) Escrever valor lógico 0
   a. Sinais de controle utilizados: Bit, PRE, *word-line* e *write-enable*
   b. Nodos analisados: BL, BLB, Q e Qb
   c. Comportamento esperado: BL ser descarregada, Q passar para 0 lógico

ii) Deixar o circuito em espera durante a recarga das *bitlines*
   a. Sinais de controle utilizados: PRE, WL e WE

b. Nodos analisados: BL, BLB, Q e Qb
c. Comportamento esperado: BL e BLB serem carregadas até Vdd, Q continuar em 0 lógico

iii) Ler o valor contido na célula
a. Sinais de controle utilizados: PRE, WL e SAE
b. Nodos analisados: BL, BLB, Q, Qb, Out e OutB
c. Comportamento esperado: tensão em BLB ter uma pequena diminuição, Q continuar em 0 lógico, Out ser totalmente descarregado.

E. Cálculo de Desempenho e Estabilidade

O desempenho do bloco de SRAM será medido através da velocidade mínima necessária para o funcionamento correto das operações de leitura e escrita, ou seja, pelos tempos de escrita e tempos de leitura do bloco SRAM. Também serão computados os valores de SNM para as operações de leitura e espera. As próximas subseções detalham como estes dados são calculados.

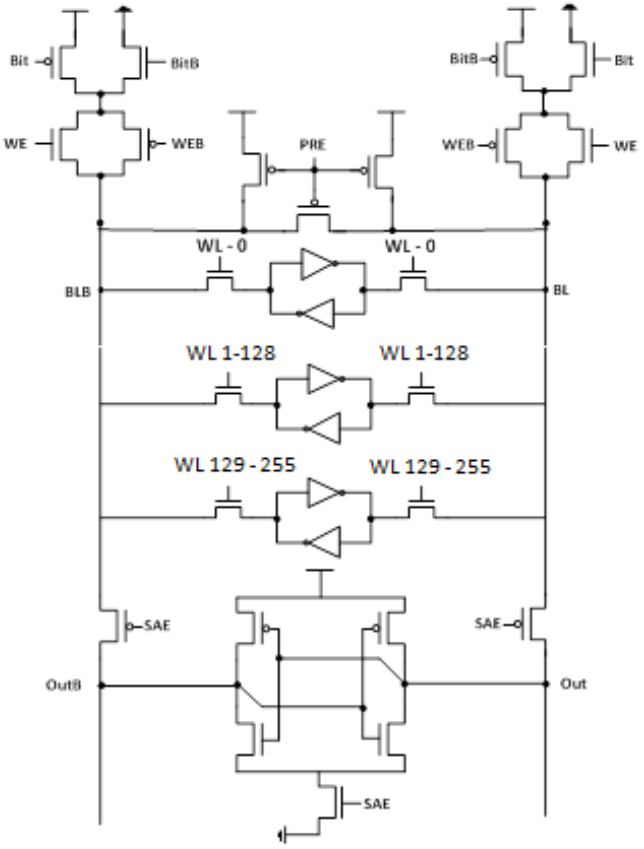

Fig. 17: Integração final dos circuitos

1) Tempos de Escrita

Para as análises de desempenho da operação de escrita utilizou-se o mesmo sinal para WE, WL e PRE. Essa abstração condiz com o funcionamento ideal de escrita da SRAM, onde durante a operação a pré-carga deve estar desligada, enquanto WL libera acesso à célula de memória e WE permite a modulação correta das *bitlines* pelo circuito de escrita. Dessa forma chegamos a seguinte equação:

$$T_{write} = T_{sinal \to 1} + T_{nodo \to 0} + T_{sinal \to 0} + T_{BL/BLB} \quad (3)$$

Onde $T_{sinal \to 1}$ é o tempo de transição dos sinais WE, WL e PRE para '1' lógico. $T_{nodo \to 0}$ representa o tempo necessário para troca do valor contido na célula. $T_{sinal \to 0}$ é o tempo de transição dos sinais WE, WL e PRE para '0' lógico. E $T_{BL/BLB}$ é o tempo necessario para a recarga das *bitilines* até V_{dd}.

2) Tempos de Leitura

A operação de escrita tem seu tempo definido da seguinte forma:

$$T_{read} = T_{WL \to 1} + T_{Vdiff} + T_{sinal} + T_{amp} + T_{SAE \to 0} + T_{rec} \quad (4)$$

Onde $T_{WL \to 1}$ representa o tempo para a transição de WL de '0' para '1' lógico. T_{Vdiff} é o tempo necessário para que a diferença de voltagem entre BL e BLB alcance o valor desejado. T_{sinal} é o tempo de transição dos sinais SAE (vai para '1' lógico) e WL (vai para '0' lógico) de forma que o *Sense Amplifier* passe a funcionar. T_{amp} é o tempo que o SA leva para amplificar a diferença de voltagem entre BL e BLB para valores lógicos.

3) Cálculo de SNM

Para o cálculo de SNM é necessário espelhar a curva VTC e encontrar o maior quadrado possível entre a VTC original e a VTC espelhada. A Fig. 18 apresenta os passos necessários para obter a curva borboleta: (a) representa a curva de transferência de voltagem normal, no eixo x possuímos V_{in} e no eixo y V_{out} dos inversores da SRAM 6T; (b) representa a VTC espelhada. Esta curva é facilmente obtida trocando os valores entre os eixos x (V_{in}) e o eixo y (V_{out}); (c) é a união das duas curvas anteriores e denominada de curva borboleta.

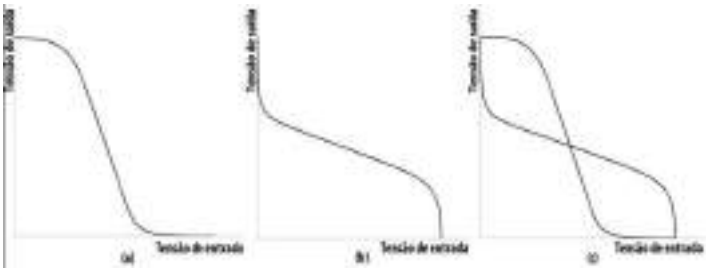

Fig. 18: (a) VTC. (b) VTC espelhada. (c) Curva borboleta.

Após o espelhamento das curvas, deve-se calcular o maior quadrado possível entre a VTC original e a VTC espelhada. Para calcular o maior quadrado possível, calculou-se a maior reta com declividade de 45 graus que interceptava as duas curvas, esse passo pode ser visto na Fig. 19 (a). Esta reta representa a diagonal do quadrado máximo que representa a SNM. Com a diagonal do quadrado podemos gerar os lados do quadrado, como demonstra a Fig. 19 (b).

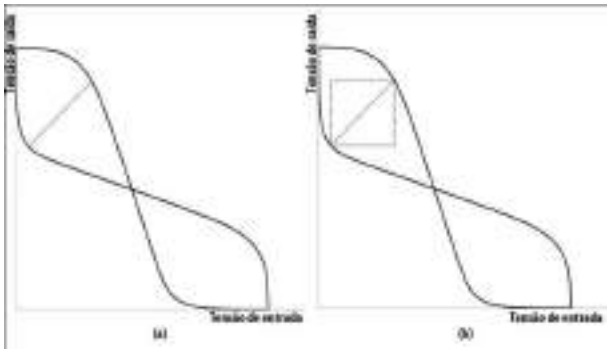

Fig. 19: (a) Diagonal do quadrado máximo. (b) Lados do quadrado máximo.

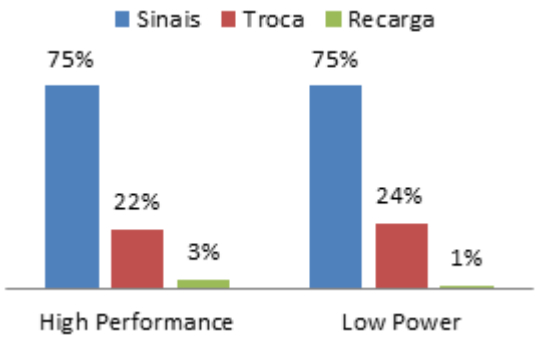

Fig. 20: Análise de percentual dos tempos de troca e recarga em relação ao tempo total durante a operação de escrita.

III. Resultados

Esta seção apresenta os resultados de validação, desempenho e robustez dos dois blocos de SRAM desenvolvidos nesse trabalho.

Cada circuito projetado, nas duas tecnologias, foi validado individualmente de acordo com os procedimentos descritos na seção III. Os dois blocos, após integrados, foram novamente validados e sua funcionalidade comprovada.

O desempenho do bloco de SRAM será medido através da velocidade mínima necessária para o funcionamento correto das operações de leitura e escrita. A Tabela 1 apresenta os valores para cada arquitetura segundo as equações apresentadas na seção III.

Tabela 1 – Resultados de Desempenho

Arquitetura	Escrita	Leitura
HP	128 ps	106 ps
LP	2629 ps	2348 ps

Comparando os tempos entre os dois blocos, percebe-se claramente que o bloco HP leva grande vantagem no desempenho de escrita e leitura. Os tempos T_{write} e T_{read} para o bloco HP são mais de 20 vezes mais rápido que os tempos observados para o bloco LP. A vantagem da arquitetura HP é inerente as características dos transistores HP. A tensão de limiar do transistor HP é mais baixa e isso faz com que todos os transistores tenham maior capacidade de corrente e que os inversores contidos tanto na célula de memória e no circuito de leitura troquem de estado mais rapidamente. O inverso se aplica ao transistor *Low Power*, ou seja, a tensão de limiar alta faz com que a troca de estado nos inversores seja mais lenta.

Pode se perceber, analisando a Fig. 20, que, apesar da grande diferença entre desempenho das arquiteturas, o tempo de recarga é o principal fator de tempo para ambos os blocos durante a escrita.

A Tabela 2 mostra os resultados de SNM encontrado. Observa-se que o bloco *Low Power* teve um melhor desempenho no quesito de robustez, sendo a SNM do bloco LP 4,25 vezes maior que a SNM do bloco HP durante a operação de espera e 2,9 vezes maior durante a operação de leitura.

Tabela 2 – Resultados de Estabilidade (SNM)

Arquitetura	SNM Leitura	SNM HOLD
HP	41,4 mV	77,4 mV
LP	176,1 mV	224,3 mV

A vantagem do bloco LP advém da alta tensão de limiar do transistor Low Power. Isso faz com que o ruído elétrico tenha que ser maior para causar o *flip* da célula de memória, tanto para a operação de leitura quanto de espera. O contrário se aplica ao bloco HP, com transistores com baixa tensão de limiar.

IV. Conclusões

A SRAM ocupa um espaço muito importante nas hierarquias de memória. Além de sua importância no desempenho dos computadores, a SRAM vem ocupando cada vez mais espaço físico nos processadores, por vezes ocupando mais de 60% do espaço do chip. Existem diversos tipos de arranjos de transistores para a célula de memória SRAM. Dentre os diversos tipos, o mais comumente utilizado é chamada de SRAM 6T. Fazendo uso de seis transistores ela é conhecida por sua robustez e simplicidade do arranjo dos transistores.

A redução constante na tecnologia e a necessidade de dispositivos que consumam menos energia trazem novos desafios aos projetistas de SRAM. A miniaturização da tecnologia melhora diversos aspectos dos circuitos integrados. No entanto, causa um impacto negativo na confiabilidade da SRAM. Dessa forma, é indispensável o estudo dos impactos das novas tecnologias na memória SRAM.

Os principais objetivos desse trabalho foram o projeto, validação, análise e comparação de dois blocos de SRAM desenvolvidos em tecnologias nanométricas de 16nm. Um bloco foi projetado com transistores voltados para alto desempenho (*High Performance*) e outro, com transistores de baixo consumo de potência (*Low Power*). O primeiro passo para realização deste trabalho foi a análise geral do bloco de memória SRAM, suas características elétricas, de dimensionamento, projeto e estabilidade. O projeto dos blocos foi dividido em quatro circuitos, um para leitura, um para escrita, outro para equalização e pré-carga da arquitetura e por

fim a célula de memória.

A validação dos dois blocos demonstrou que é possível projetar células de memória na tecnologia de 16nm que tenham suas operações de escrita, espera e leitura funcionando corretamente, tanto com transistores HP como com transistores LP para criar esse tipo de memória.

A análise dos resultados de desempenho demonstrou a larga vantagem de desempenho das operações de leitura e escrita do bloco HP em relação ao bloco LP. Tanto a operação de leitura quanto a de escrita da arquitetura HP foram 20 vezes mais rápidas que as operações da arquitetura LP. Outra elucidação feita foi que, nos dois blocos projetados, a recarga das linhas de bit é uma funcionalidade crítica para as operações.

A análise dos resultados de robustez elucidou um ganho de SNM na ordem de 4,25 vezes maior na operação de leitura e 2,9 vezes maior na operação de espera do circuito LP em relação ao bloco HP. O HP levou vantagem nos resultados de desempenho ao custo de uma baixa robustez quando comparada à SNM do circuito LP. A análise também confirmou a degradação da SNM de leitura em relação à SNM da operação de espera. Essa característica se manteve em ambas as arquiteturas.

Diversos passos podem ser dados para dar continuidade ao trabalho, dentre eles, três podem ser considerados essenciais para uma análise mais completa da SRAM. Foram analisamos dois quesitos da SRAM, desempenho e robustez. Desta forma faltaram analisar consumo de potência e área ocupada. Esses dois quesitos também são de extrema importância ao se projetar uma arquitetura de SRAM. Outro passo para continuar o trabalho é a redução gradual das tensões do circuito, dentre elas a tensão nominal, a tensão de pré-carga e a tensão da *wordline*. A tensão nominal pode ser reduzida com o objetivo da redução do consumo de potência. A tensão de pré-carga e da *wordline*, com o objetivo de tornar o circuito mais robusto, ou seja, aumentar a SNM. Por fim, outra questão a ser tratada é a análise de outros arranjos dos transistores para a células de memória SRAM, como por exemplo, as células de memória SRAM 7T e 8T.

REFERÊNCIAS

[1] Neumann, “First Draft of a Report”, 1945.

[2] Hiroyuki Yamauchi, “A Discussion on SRAM Circuit Design Trend in Deeper Nanometer-Scale Technologies”, IEEE Transactions on Very Large Integration (VLSI), vol. 18, No.5, Maio 2010

[3] Masood Qazi, Mahmut E. Sinangil, Anantha P. Chandrakasan. “Challenges and Directions for Low-Voltage SRAM.”, 2011.

[4] Volnei A. Pedroni, Eletrônica Digital Moderna e VHDL, 2010

[5] Alorda, B., Torrens, G., Bota, S., Segura, J, “Static-Noise Margin Analysis during Read Operation of 6T SRAM Cells.”, 2009

[6] Andrei Pavlov, Manoj Sachdev, CMOS SRAM Circuit Design and Parametric Test in Nano-Scaled Technologies. Springer, 2008.

[7] Frans J List, Jan Lohstroh, Evert Seevinck, “Static-Noise Margin Analysis of MOS SRAM Cells”, IEEE Journal of Solid-State Circuits, outubro, 1987.

[8] S. Natarajan, et al., “A 32nm Logic Technology Featuring 2nd - Generation High-k + Metal-Gate Transistors, Enhanced Channel Strain and 0.171µm2 SRAM Cell Size in a 291Mb Array”. IEEE Int. Electron Devices Meeting, Dec, 2008.

[9] Fatih H., et al., “Bit Cell Optimizations and Circuit Techniques for Nanoscale SRAM Design”, IEEE Design & Test of Computers, vol. 28, Issue:1, 2011, pp. 22 - 31

[10] Karl, Eric; et al., “A 4.6GHz 162Mb SRAM design in 22nm tri-gate CMOS technology with integrated active VMIN-enhancing assist circuitry”, IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers, 2012.

[11] Kolar, P.; Karl, E.; Bhattacharya, U.; Hamzaoglu, F.; Nho, H.; Yong-Gee Ng; Yih Wang; Zhang, K. “A 32 nm High-k Metal Gate SRAM With Adaptive Dynamic Stability Enhancement for Low-Voltage Operation”, IEEE Journal of Solid-State Circuits, vol. 46 , issue: 1, 2011, pp. 76 - 84

[12] Kelin Kuhn, "Moore's law past 32 nm: Future challenges in device scaling", Proc. Int. Workshop Comput. Electron., 2009

[13] Balwinder Raj, A.K. Saxena e S. Dasgupta, "Nanoscale FinFET Based SRAM Cell Design: Analysis of Performance Metric, Process Variation, Underlapped FinFET and Temperature Effect", IEEE CIRCUITS AND SYSTEM MAGAZINE, 3º trimestre de 2011.

[14] Yen Huei Chen, et al., "A 0.6 V Dual-Rail Compiler SRAM Design on 45 nm CMOS Technology With Adaptive SRAM Power for Lower VDD_min VLSIs", IEEE JOURNAL OF SOLID-STATE CIRCUITS, Abril 2009.

[15] Hiroyuki Yamauchi, "A Discussion on SRAM Circuit Design Trend in Deeper Nanometer-Scale Technologies", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, vol. 18, n. 5, Maio 2010.

[16] Yih Wang, et al., "A 4.0 GHz 291 Mb Voltage-Scalable SRAM Design in a 32 nm High-k + Metal-Gate CMOS Technology With Integrated Power Management", IEEE JOURNAL OF SOLID-STATE CIRCUITS, vol. 45, n. 1, Janeiro 2010.

[17] W. Zhao and Y. Cao. New Generation of Predictive Technology Model for Sub-45 nm Early Design Exploration. IEEE Transactions Electron Devices, vol.53, pp.2816–2823, November 2006.

[18] NGSPICE http://ngspice.sourceforge.net/

[19] Phyton. http://www.python.org/