Statistical Approach for the Design of Refresh-Free eDRAM with Retention Timing Constraint

ARTURO PRIETO LLORENS
MASTER´S THESIS
DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY

Statistical Approach for the Design of Refresh-Free eDRAM with Retention Timing Constraint

Arturo Prieto Llorens
ar4628pr-s@student.lu.se

Department of Electrical and Information Technology
Lund University

Supervisor: Babak Mohammadi

Examiner: Pietro Andreani

April 16, 2019

Printed in Sweden
Tryckeriet i E-huset, Lund

Abstract

In digital integrated circuits, memories are often a limiter for main performance, power and area. Over the past decade, integrated memories have gained dominance in terms of area and power cost. In applications like machine learning or image processing the area share can be above 80% consuming more than 50% of the total power budget. On-chip memories in CMOS technologies can be categorized as static (SRAM) and dynamic (embedded DRAM). SRAM has been the main choice due to its high-access rates and static data retention feature. eDRAM offers higher area gain over SRAM counterpart. However, dynamic memories require periodic, power-hungry refresh operations. This operations increase the design complexity and have a high energy cost. Furthermore, they lower the access rates due to memory restriction during refresh periods, which makes it less desirable in SoC context.

A wide range of computation intensive applications in today's digital systems need to buffer intermediate data for a short fraction of time. Multiple Input Multiple Output (MIMO) communication and convolutional image processing are two cases that use memories for storing data during short time periods. These are appropriate applications for the use of eDRAM without refresh operations.

The goal of this study is to evaluate and develop an eDRAM memory compiler. This compiler evaluates the design of an eDRAM cell and considers the effect of manufacturing process deviations. These variations are based on statistics, and for their analysis, a new statistical approach 100 times faster than the generally used analysis method is developed. The output of the compiler is a macro layout according to the eDRAM cell and results from the statistical design evaluation. The final result is a tool that allows the automatic generation of eDRAM as a function of user requirements.

Popular Science Summary

Memories have an important role in electronic devices with a strong impact in the performance, power and area of digital integrated circuits. The constant increase in data flow intensifies the effect of memories in the design. For example, image processing applications require the storing of one image at a time for their processing. Once an image has been processed, a new image is overwritten in the memory. The data is stored momentary according to the processing period. The temporary storage of these applications is appropriate for the use of memories based in non-static data retention.

Embedded dynamic memories (eDRAM) are a good alternative for the memory implementation in these applications. They use less number of transistors, which represents an area save compare to other possible solutions. The dynamic character of eDRAM require refresh operations, increasing power and restricting the memory access during these operations. However, in applications like image processing, it is possible to design the memory according to the processing period, been able to remove the power-hungry refresh operations. This thesis aims to implement a memory compiler for the automatic generation of Refresh-Free eDRAM according to the application requirements.

Other requirement for the memory generation is the effect of process variations in the design. They are determined by the manufacturing process and affect to the reliability of digital circuits. The trend of reduction in transistor dimensions provides reduction in area cost of circuit implementations. However, it entails more manufacturing difficulties, increasing the effect of process variations. In order to ensure the reliability of the design, the effect of process variations is part of the compiler for the automatic generation of Refresh-Free eDRAM.

Acknowledgements

I would like to express my gratitude to my supervisor Babak Mohammadi, CEO of Xenergic, for the opportunity of this thesis and the support during this time. I also would like to thank my colleagues Tom, Hemanth, Xiao and Srini for the help and collaboration during the project. Special thanks to Berta and Caty for the help and also for the personal advice. Finally, I would like to thank my professor Joachim Rodrigues for the patience and support since the beginning of the thesis.

Acronyms

eDRAM embedded Dynamic Random Access Memory

CMOS Complementary Metal Oxide Semiconductor

DRAM Dynamic Random Access Memory

DRC Design Rules Check

DRT Data Retention Time

FF Fast Fast

FS Fast Slow

GC Gain Cell

IC Integrated Circuit

IS Importance Sampling

MC Monte Carlo

NMOS N-type Metal Oxide Semiconductor

PMOS P-type Metal Oxide Semiconductor

RBL Read Bit Line

RWL Read Word Line

SA Sense Amplifier

SF Slow Fast

SN Storage Node

SRAM Static Random Access Memory

SS Slow Slow

WBL Write Bit Line

WWL Write Word Line

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction to embedded dynamic memories

1.1 Refresh-Free eDRAM discussion and applications

Today's systems have increased data flow intensifying the effect of memory in the design, which often occupies more than 80% of the average silicon area in digital systems [1]. Static Random Access Memories (SRAM) has been a traditional choice for on-chip applications due to its high-access rates and static data retention with relatively high area density [2]. However, static memory cells are constructed using 4 to 14 transistors [3, 4], making it an area expensive option. An alternative is the use of embedded Dynamic Random Access Memory (eDRAM), typically built with fewer transistors, resulting in smaller area footprint compared to SRAM. For eDRAM, data retention is attained by dynamically stored charge, and thereby requires periodic power-hungry refresh operation [2]. This process increases the design complexity, has a high energy consumption, and lowers the access rates due to memory restriction during refresh periods. The interval of time that the cell is able to retain the data before requiring a refresh operation is denominated Data Retention Time (DRT). An embedded dynamic memory without refresh operation, and which storage is dependant on DRT will be defined as Refresh-Free eDRAM.

Refresh-Free eDRAM offers advantages for certain category of applications, where the time required for storing data is smaller than the retention time. One can envision a lot of signal processing applications with streaming data flow to have short data storage time requirements. Wireless applications is one example wherein there is a constant flow of new data packets, which can overwrite the old processed packets. Also in Convolutional Neural Network (CNN) applications for image classification, the image data just needs to be stored in the memory during the convolutional process, once this has been completed, the data will be overwritten and a new image frame will be processed. The period of time required by the application for doing the processing of an image is the DRT requirement of the Refresh-Free eDRAM.

Similar to other memories, eDRAM are subject to process variations and this can impact the overall yield and the performance of the design [5]. In the manufacturing process of the memory, the large number of components and fabrication imperfections cause deviations in transistors that may affect to the performance

and reliability of the system. The trend of reduction in transistors dimensions over time is shown in Figure 1.1. This sizing trend provides reduction in area cost for the the number of transistors that may be used in a process implementation. However, the manufacturing difficulties produce that the effect of process variations is increased as going to smaller technologies.

This study is addressing the necessary steps required to design and implement a Refresh-Free eDRAM using the constraints from target application.

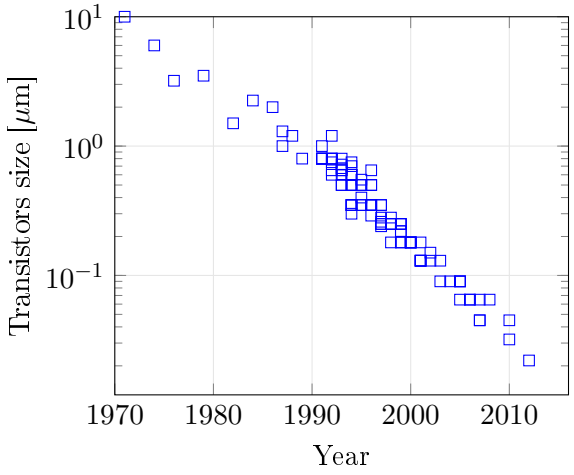

Figure 1.1: Technology scaling in the last 40 years [6]

1.2 Project specifications

The aim of this project is to develop an automatic Refresh-Free eDRAM macro compiler. Where the key idea is to generate an eDRAM cell based on the application requirements. The flow will be technology independent considering retention time and cell failure probability. It should be able to create the memory cell layout considering the effect of variations, and it should include the automatic macro generation for a size requirement.

For the flow evaluation, different cells need to be studied and various examples of data retention time and failure probability will be considered in 28nm Complementary Metal Oxide Semiconductor (CMOS) BULK technology.

1.3 Thesis organization

The report is organized as follows:

- **Chapter 2:** background with reference to statistical analysis and dynamic memories.
- **Chapter 3:** statistical method for the analysis of variations affecting to the design.

- **Chapter 4:** evaluation of different eDRAM cells and their feasibility in relation the requirements.
- **Chapter 5:** eDRAM macro compiler including design, characterization and macro generation.
- **Chapter 6:** results of the design and evaluation of the new statistical method, eDRAM cell architectures and the eDRAM macro compiler functionality.
- **Chapter 7:** conclusions of the project.
- **Chapter 8:** future work for the completion of the necessary blocks for the application of these memories.

Chapter 2
Background

2.1 Statistical analysis

The last decade has seen a potential growth in the amount of manufacturing variations in silicon circuits, as a result, proper manufacturing yield is no longer guaranteed easily, and must be explicitly optimized during the design phase [7]. These deviations affect the behaviour of the integrated circuit components and are present in all designs.

Memories are typically large arrays of cells built by transistors. The continuous technology scalability in addition to the increase in the number of components present in an Integrated Circuit (IC) emphasize the importance of considering variations of transistors parameters during the design. There are two types of variations:

- Process: refers to the variations between different chips on a single waffer and between the waffers themselves.
- Mismatch: refers to the variations of the devices which are located close to each other in the same chip.

Statistical variations are the dominant source of variability of the transistors parameters. It is necessary to know the statistics of them to analyse their effect in the yield. It is observed that these variations follow a Gaussian distribution (see Figure 2.1) [7].

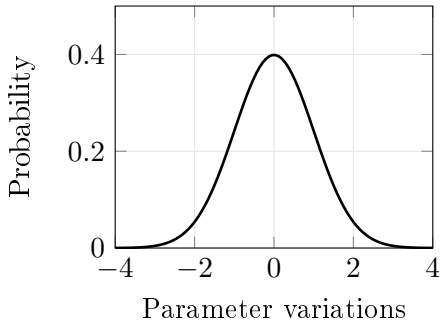

Figure 2.1: Transistor parameter Gaussian distribution

Monte Carlo (MC) method is a statistical approach commonly employed for the analysis of problems which components are defined inside a probability distribution. It relies on the random sampling of the parameters distributions to obtain numerical results [8]. For each simulation, it takes a random sample of each parameter and performs a design evaluation with the given configuration. If the system fails, the failure is accumulated in relation to the number of simulations performed to define the failure probability. Due to the parameter distributions are Gaussian, the sampling is done mainly around the mean, where the probability is higher, than in tails, where the probability is lower.

2.2 Dynamic memories

Dynamic memories are a type of Random Access Memory (RAM). They allow both processes of writing and reading and are a type of volatile memories that hold data while the power supply is provided. eDRAM are formed of CMOS transistors. The gate of a CMOS transistor is isolated from the channel by an oxide and this has a capacitance known as parasitic capacitance (C_{par}). Due to the data keeping is based in storing the writing voltage in a cell parasitic capacitance (see Figure 2.2), leakage produces the corruption of the value. Therefore, this type of memories are dynamic because require refresh operations for data storing [9].

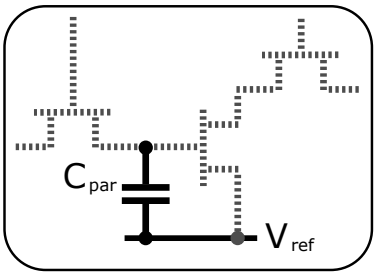

Figure 2.2: Generic DRAM cell with storing parasitic capacitance

Figure 2.3 represents the standard architecture of a dynamic memory cell. The Write Word Line (WWL) is used to control the writing process of the level charged in the Write Bit Line (WBL) to the cell. For the reading process, the Read Bit Line (RBL) is pre-charged to a level, then the activation of the Read Word Line (RWL) controls the swing that the bitcell stored level produces in the bitline.

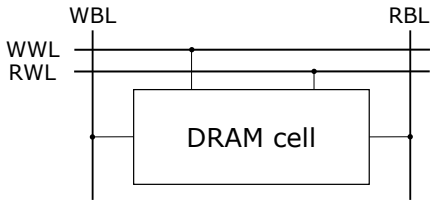

Figure 2.3: Dynamic memory cell architecture

Memory cells are connected in arrays generating macros. The size of a macro will be defined by the number of words and the number of bits per word:

$$\text{Macro size} = \text{words} \times \text{bits per word}$$

The read and write wordlines are shared between bitcells in the same row, and the bitlines are shared between the bitcells over each column (see Figure 2.4). Cells connected to the same bitlines produce parasitic capacitances that affect to the writing and reading processes. It is necessary to limit the number of words per macro to avoid high WBL bitlines parasitics that affects to the bitcell access performance.

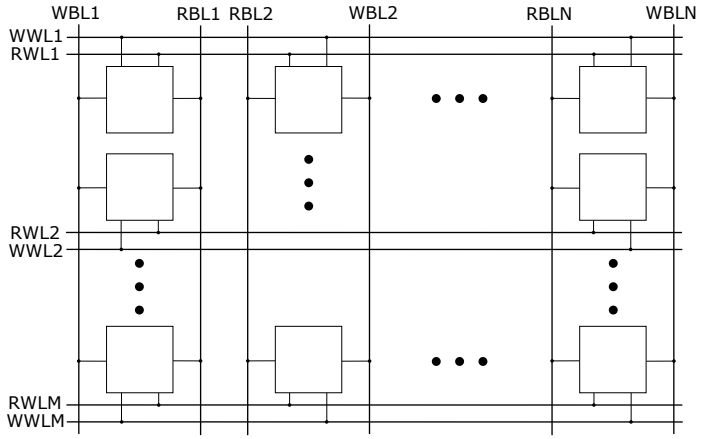

Figure 2.4: Matrix of bitcells in an eDRAM macro with the size of MxN

Macros also contain peripheral circuits merged to the bitcells matrix to perform the write and read operations (see Figure 2.5). The Sense Amplifier (SA) is part of the reading circuitry. It is used to sense the lower voltage swing in the RBL to recognize the logic level stored in the bitcell and serve it outside the memory. SA in DRAM is single-ended because in dynamic memories there is only one bitline for the reading process. It means that the RBL is compared with a reference value in the SA, in contrast with SRAM where the sensing is done using both cell bitlines.

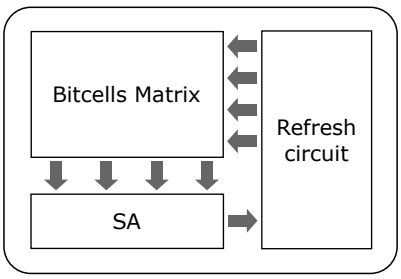

Figure 2.5: Macro architecture

A dynamic memory requires dedicated refresh circuit for the valid data storage (see Figure 2.5). The refresh operation consists of a read of the cell followed by a write. During the refresh period, memory access is blocked for other reading or writing operations. In applications where the memory writing rate is higher than the Data Retention Time, the use of refresh circuit is unnecessary due to the stored value will be overwritten before the refresh process occurs.

Chapter 3

Statistical approach for eDRAM design

As mentioned in section 2.1, it is important to consider the statistical variations of transistor parameters to guarantee a functional silicon circuit. Process corners are used to represent the distant variations of these parameters. The convention for referring to these corners is by the carriers mobility of NMOS and PMOS transistors. Typical (T), Fast (F) and Slow (S) are usually used to refer to normal, high and low carrier mobility, respectively [10]. The combinations of these cases defines the process corners (see Figure 3.1). The first letter refers to NMOS devices and the second to PMOS.

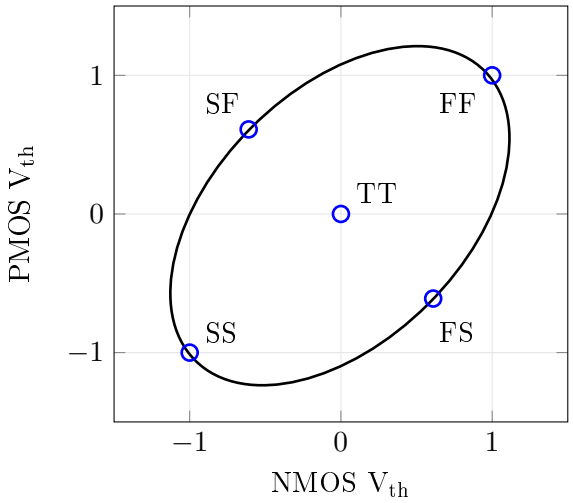

Figure 3.1: Process corners distribution

For circuits with large amount of transistors, like a memory, the use of process corners overestimates the design. Consequently, it is usual to establish a target failure criteria that defines the probability of failing for the design. The failure probability of the bitcell ($F_{\text{f-cell}}$) is dependant on the desired failure criteria of the yield ($F_{\text{f-mem}}$) and the memory size (N) [11]:

$$F_{f-cell} = 1 - [1 - F_{f-mem}]^{\frac{1}{N}}$$

In Table 3.1 are three examples of the required failure probability of the bitcell for different constraints of memory yield and size. For all cases, the failure probability design requirement is lower than one in a million.

Size [MB]	$F_{f\text{-}mem}$ [%]	$F_{f\text{-}cell}$
4	50	2.07×10^{-8}
8	20	3.33×10^{-9}
16	1	7.49×10^{-11}

Table 3.1: Targeted cell probability failure for different memory sizes and yield requirements

3.1 Importance Sampling

Monte Carlo is an accurate method for calculating failure probability. However, in designs with high replication and low error rate it is not an efficient method. This is due to the fact that the sampling is done mainly around the mean than in tails of the parameters distributions. For evaluating rare failure events, they can be found in tails of the Gaussian distributions. Consequently requires a large number of samples to find the failure probability of the design. For this reason, new sampling methods were evaluated in order to speed up the statistical analysis trying to find rare failure events clearly and calculate the $F_{f\text{-}cell}$ with less number of samples.

Importance Sampling (IS) is a statistical approach similar to MC that also relies on the random sampling of the parameters distributions. However, IS does not take samples from the original Gaussian p(x), it defines a new distribution g(x) to sample more in the targeted region than around the mean. A simple approach is to shift the mean of the natural distribution in the failure region (see Figure 3.2) [12].

Independent of the sampling method, every sampling point f(x) is associated with an indicator function I(x), where f_0 is the fail-pass criteria of the design [7]:

$$I(x) = 0 \text{ (pass)}, f(x) < f_0$$

$$I(x) = 1 \text{ (fail)}, f(x) > f_0$$

Each random sample generated from the shifted distribution needs a conversion to the probability in the original Gaussian. The weight function W(x) is the relation between the probabilities of a sampling point in the new and in the original distributions:

$$W(x) = \frac{p(x)}{g(x)}$$

For the calculation of the failure probability (P_f), it is necessary to define a function y(x) that applies I(x) to consider only the probabilities of the failing points in a function that disregards the others:

$$y(x) = W(x) \cdot I(x)$$

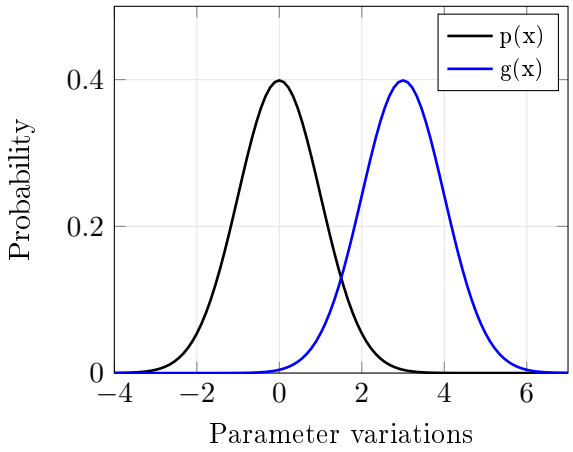

Figure 3.2: Original Gaussian and mean shifted distributions

$$P_f = \frac{\sum_{i=1}^{\infty} y(x_i)}{\sum_{i=1}^{\infty} W(x_i)} = \frac{\overline{y(x)}}{\overline{W(x)}}$$

In [12], it is proposed a mixture distribution combining the original Gaussian, a uniform U(x) and a shifted distribution (see Figure 3.3). This allows to generate random variables focusing on the failure region without leaving any cold spots. Other option is proposed in [8] based on using a wider Gaussian density function for sampling (see Figure 3.4). The result is a larger range compared to the standard Monte Carlo analysis.

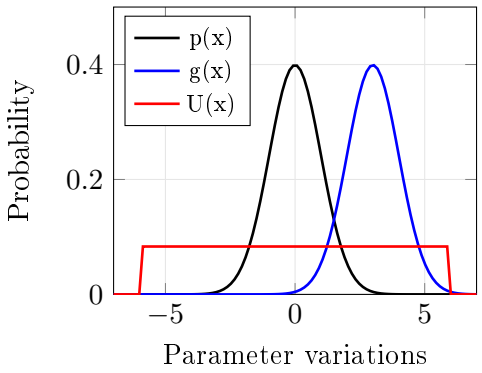

Figure 3.3: Mixture IS

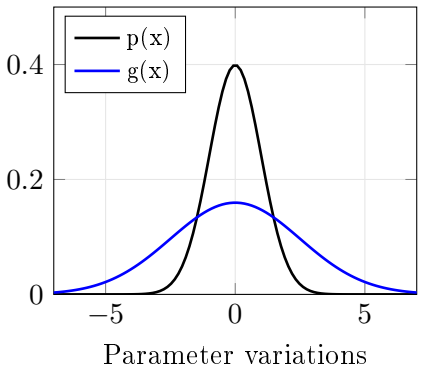

Figure 3.4: Wide sampling

When generating a new shifting distribution, besides the form, it is also important to define accurately the shifting position. The basis is being able to find the failure region without leaving any important area of the original Gaussian distribution. Study in [13] proposes a new technique to identify the optimal shift-vector using the results of doing an initial analysis of all parameters distributions in order to find the failure region. This process is implemented in different steps

to accurately sample each region of the parameters distributions. Then, the samples that have the smaller Euclidean norm have assigned the larger importance weight and will define the shifting position. This method lacks in the possibility of having more than one failure region in the same distribution due to the new distribution will only cover the shifted area. In [14] the proposed method covers different failure regions in the same distribution by considering a shifting function that will be a mixture of them.

After the evaluation of these alternatives, the method implemented in this study is based in four steps:

1. An initial uniform distribution U(x) used to sample the complete range of the transistor parameters distributions in order to find failing points.
2. The arrange of these points generating failure areas that determine the shifting position of the new IS distribution. In case more than one failure region is found, the IS distribution will also cover them.
3. Random sampling using the new distribution to evaluate the design.
4. Calculation of the failure probability of the design for the established fail-pass criteria with the indicator function I(x) and the weights calculation W(x).

Chapter 4
eDRAM cell

eDRAM technology arises from DRAM basis but adapted to be compatible with CMOS fabrication processes [15]. This makes the memory integrated on-chip and eliminates expensive inter-chip communications. It offers higher density and lower leakage compared to SRAM, and with the removal of the refresh circuit is a good alternative for applications that need to buffer data for a short fraction of time. The eDRAM compiler is independent of the cell architecture, and any type of dynamic cell can be used. However, different eDRAM cell architectures were evaluated as consideration for the memory generation flow.

4.1 Cell architecture

As mentioned in section 1.2, the removal of the refresh circuit makes DRT together with process variations the main specifications for the cell design in this project. Figure 4.1 shows a common implementation based on three transistors of the same type. The parasitic capacitance of M2 is used to store the bitcell value. M1 is the pass transistor for writing in the cell. In the writing process, WWL activates M1 to store in Storage Node (SN) the pre-charged voltage in WBL. In the reading process, M3 is the transistor for the reading activation and the SN voltage produces M2 to be active and to discharge RBL through M2 and M3.

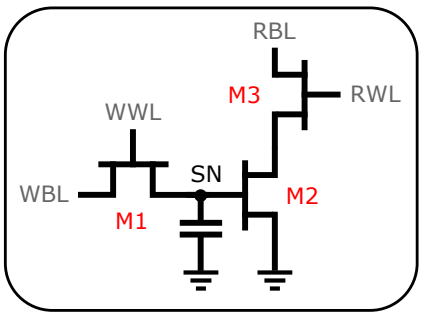

Figure 4.1: N-type 3T eDRAM

When writing a logic value 1, the maximum voltage that can pass through M1

depends on its threshold voltage:

$$V_{SN} = V_{WWL} - V_{th_{M1}}$$

When reading a logic value 1, the maximum time after writing to activate RWL has to guarantee that V_{SN} is high enough to switch on M2 and discharge RBL:

$$V_{SN} \geq V_{th_{M2}}$$

Analysing these processes and dependencies, we define the components that affect to DRT of the cell and need to be considered during the design:

- Leakage through the gate of M2 (I_{G2}) and through the pass transistor M1 (I_{DS1}).
- Dimension of the parasitic capacitance for storing the logic level.
- Maximum voltage that can pass through M1.
- Reading speed, which is determined by the discharge of RBL through M2 and M3.
- Sense Amplifier specifications that will define the range between reading logic levels 0 or 1.

We have evaluated other proposed architectures for the improvement of these components. Gain Cell (GC) is a technique that can boost the storage voltage via capacitive coupling [16]. Figure 4.2 shows an alternative 3T gain cell. Similar to previous design in Figure 4.1, this cell is composed of three transistors. However, the drain of the storage device M2 is connected to RWL. For reading, first RBL is pre-charged to 0 V and then RWL switches high to activate M3. If the storage level is high and M2 is activated, the SA will detect the RBL charging to V_{RWL}.

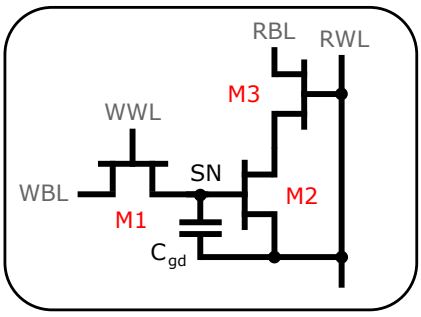

Figure 4.2: 3T NMOS eDRAM Gain Cell

The idea of this technique is to improve the cell's data retention capability by boosting the SN voltage having connected RWL to the drain of the storage device. When the stored level is high, the gate-to-drain coupling capacitance (C_{gd}) is larger compare to when the stored level is low, because M2 in inversion mode makes the entire oxide capacitance act as coupling capacitance [16]. This increases the signal

difference during read which allows the storage node voltage to decay further with level high in the cell. This translates into higher DRT.

Other option proposed in [17] is a Gain Cell structure with two transistors (see Figure 4.3). This cell uses the same DRT improvement technique as the cell with three transistors. However, during a read operation, RBL and RWL are pre-charged high at the same time, then RWL switches to low and if the stored voltage is high, RBL is discharged through M2.

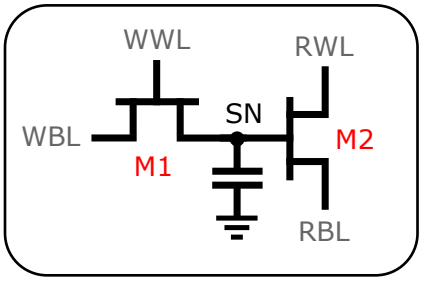

Figure 4.3: 2T NMOS eDRAM Gain Cell

In order to achieve a significantly higher DRT, it was evaluated a 4T Gain Cell topology presented in [18]. The proposed bitcell includes two additional NMOS devices (NB and NF), which form a feedback loop to improve the retention time of the cell (see Figure 4.4). Both transistors are low-V_{th} in order to increase the strength of the feedback loop and minimize the effect of the voltage decay through NB in the writing process.

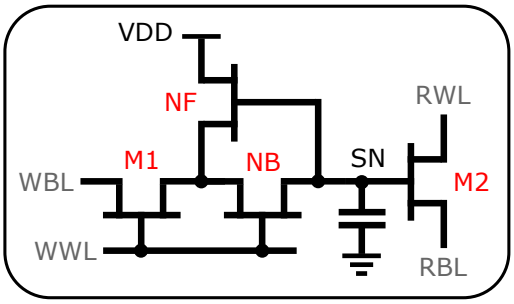

Figure 4.4: 4T NMOS eDRAM Gain Cell

A 5T Gain Cell topology and other Gain Cell architectures are presented in [19] and [20]. However, they were not evaluated due to the use of five transistors does not represent a potential area reduction compare to a SRAM cell.

The different architectures presented were simulated with minimum dimensions of all components and evaluating the DRT for the reading of logic level 1. Table 4.1 summarizes the results of the cells evaluation normalized to the 3T eDRAM cell results.

Cell	DRT [a.u.]	Write delay [a.u.]	Read delay [a.u.]
3T	1	1	1
3T-GC	1.13	1	0.87
2T-GC	1	1.01	0.59
4T-GC	16.84	2.19	0.61

Table 4.1: eDRAM cells evaluation results

The 2T-GC eDRAM cell properties are good for these applications with same retention time as 3T cell and only two transistors. However, this cell has a lot of variation in the storage value [21]. In order to reduce the effect of variations in the design, the 3T eDRAM bitcell in [9] was chosen for the study evaluation. It offers reasonable DRT in relation to the cell density. However, the cell was designed using PMOS devices since they tend to have lower leakage than NMOS resulting in better retention time characteristics [22].

4.2 Retention time relation with cell

After evaluating the different cell architectures, the device properties which effected the DRT were analysed. The concept was to set the parameters of cell components and establish one that relates the retention time with the cell design. The main parameters which effected the retention time after performing evaluations are listed below:

- Transistors V_{th}
- Transistors width (W)
- Transistors length (L)
- SA reference voltage
- SA reading time
- WWL and WBL voltages

L or V_{th} of the transistors have important effect in the retention time of the cell. However, they are very limited in the possible range of use and resolution for being function of DRT. Similar cases are the SA reference voltage and reading time, and the WWL and WBl voltages. They are more dependant of the circuit specifications and it is inefficient to design them also as function of DRT for the cell requirements.

As mentioned in Figure 4.1, the capacitance generated in the storage node SN of an eDRAM cell, is mainly the effect of the parasitic capacitance of the transistor M2, which is directly dependent on the width of this transistor. This parameter will be used to determine the design of the bitcell as function of the input DRT for this application.

For calculating the function that relates these two parameters, we simulated different cases for a given SA specifications of reading time and reference voltage.

Figure 4.5 represents the function obtained that relates DRT with the width of the storage transistor M2. This function is normalized to the retention time obtained for a cell design of M2 width 100 nm. There is a linear relationship between these two parameters because the parasitic capacitance is linear dependant with the transistor width.

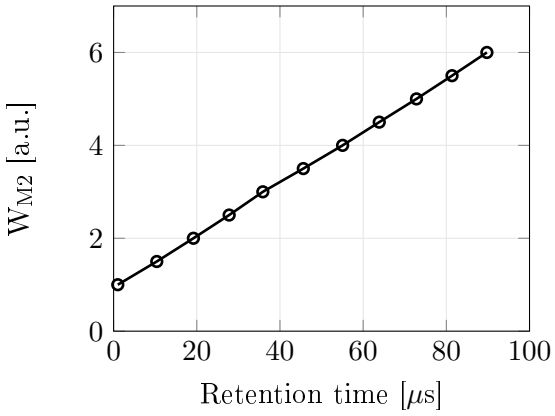

Figure 4.5: DRT relation with eDRAM cell design

It was also necessary to introduce the effect of variations in the bitcell design considerations. The intention was to model their effect as a function of the M2 width by an element that specifies the required width increment (ΔW). To the relation in Figure 4.5, it was necessary to include the effect of deviations for the accomplishment of retention time, this effect is set as ΔW. Figure 4.6 shows the relation between the design width and the ΔW necessary to add for the accomplishment of the requirements. Increasing the design width, the ΔW necessary for the accomplishment of the requirements is higher due to the effect of variations in wider transistors.

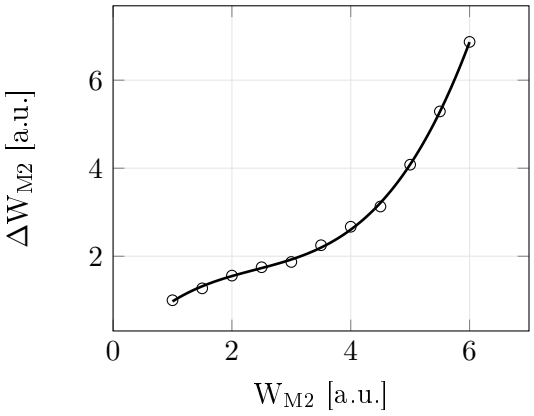

Figure 4.6: ΔW_{M2} relation with W_{M2} for 2σ

The required ΔW increases as function of W_{M2} due to the variations effect produced in the cell. Leakage current from gate (I_{gs}) deteriorates the storage voltage and it is directly dependant of the storage transistor width. Figure 4.7 shows the behaviour of I_{gs} as function of the transistor width for three corners: Typical (T), Fast (F) and Slow (S). Increasing the width, the effect of the leakage current becomes dominant and determines the increase in the required ΔW for the accomplishment of the applications requirements.

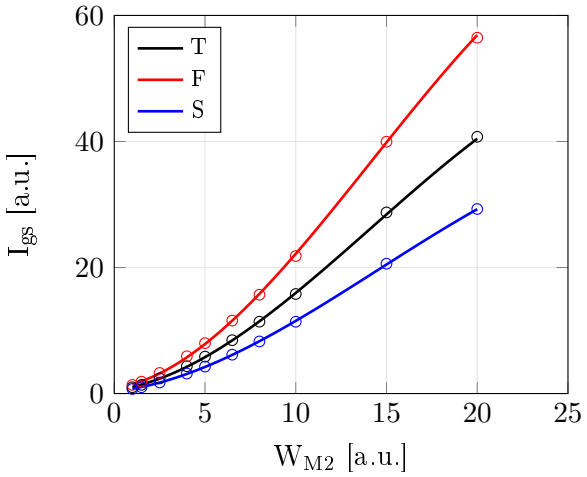

Figure 4.7: I_{gs} relation with W_{M2}

4.3 Layout

After the completion of the transistor level design, it is necessary to move to the creation of the layout. It was designed with the intentions of area reduction and to facilitate the process of generation with the storage transistor width dependency. For an efficient layout design is essential to take into consideration the Euler path of the circuit. This provides the most adequate routing when the IC components share drain or source [23]. In order to establish the Euler diagram, it is necessary to abstract the circuit logic diagram and find the transistor connections sharing drain and source (see Figure 4.8).

After obtaining the Euler path, it can be identified how the bitcell components need to be drawn in order to increase the density of the layout. Figures 4.9 and 4.10 show two examples of bitcell layout results for different retention times. As mentioned in section 4.2, all parameters have the same configuration except the width of the storage transistor. The reading transistor (M3 in Figure 4.1) needs to have the same width as M2 to ensure a Design Rules Check (DRC) clean layout.

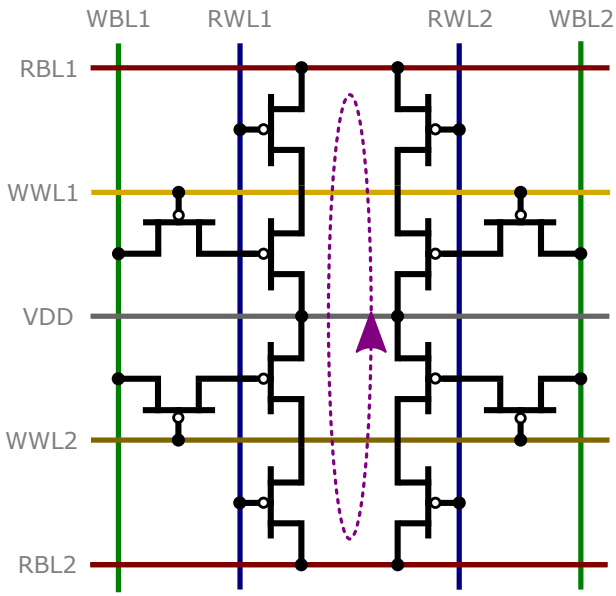

Figure 4.8: eDRAM bitcells Euler path

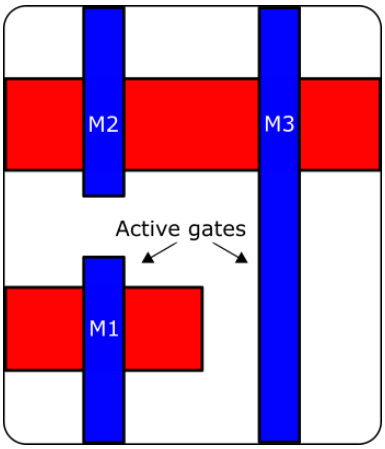

Figure 4.9: 3T-PMOS eDRAM layout with storage transistor width 125nm

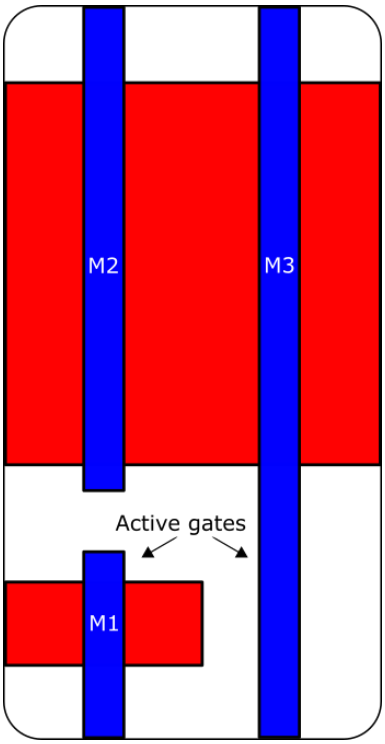

Figure 4.10: 3T-PMOS eDRAM layout with storage transistor width 600nm

4.4 Sense Amplifier Integration

The design of the bitcell has to be consistent with the SA. The sense amplifier has a sensing range where is not possible to ensure the reading value as high or low level. Due to the cell is P-type, the RBL is pre-charged to 0 and connects to VDD in the reading process. There is a maximum RBL voltage after read for stored value high and a minimum voltage for stored value low that defines this range.

Chapter 5

Refresh-Free eDRAM compiler

After the set up of the bitcell design and layout, it was necessary to automatize the memory generation by creating a memory compiler. This includes the cell evaluation for finding the necessary cell configuration according to the initial conditions, and the macro generation for the correspondent bitcell layout.

5.1 Automatic Refreh-Free eDRAM generation flow

The automatic memory generation flow covers three processes (see Figure 5.1). The netlist provided with the cell architecture specifications is evaluated in order to find the parameters configuration that accomplish the requirements of DRT and fail probability. Then, the bitcell layout is generated according to the results obtained from the characterization. Finally, it is generated a macro for the requested size along with its parasitics.

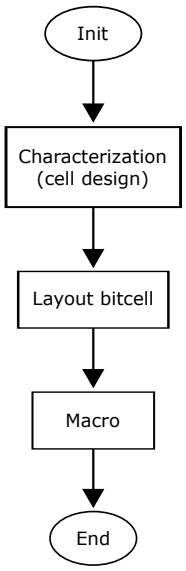

Figure 5.1: Memory compiler flow diagram

5.2 Cell characterization

The first step in the compiler is the bitcell characterization. The process flow diagram is shown in Figure 5.2. Before doing any analysis, it checks in the database if the configuration for the requirements has been already done before. If not, the characterization of the bitcell starts finding the parameters values for the defined DRT. In the base case of the design exposed in section 4, it evaluates the process of writing a logic level low and reading after the retention time delay. The leakage makes this the critical case. In Figure 5.2, the SA minimum voltage for stored value low exposed in section 4.4 is used for the definition of the cell. Then, this design is evaluated with variations and the calculated W_{M2} is adjusted with ΔW_{M2} for the required failure probability.

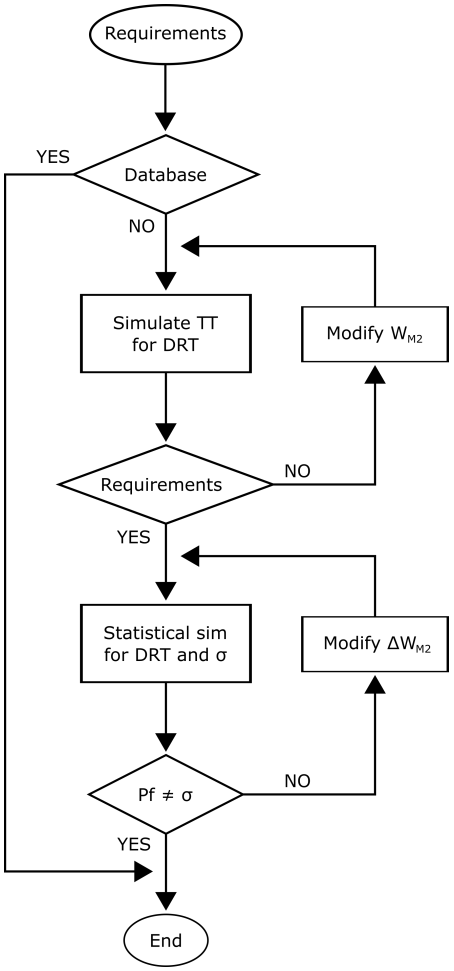

Figure 5.2: Characterization flow diagram

In this example, the final width of M2 for the layout generation is determined by the two values calculated from the flow:

$$W_{M2layout} = W_{M2} + \Delta W_{M2}$$

5.3 Macro generation

The size of the macro is critical for the writing and reading delays. The number of words determines the number of bitcells sharing the same WBL and RBL. The cells connected to these lines produce parasitic capacitances that affect to the bitcell access speed (see Figure 5.3). For that reason, it is important to limit the number of words in a macro to avoid problems in read and write operations.

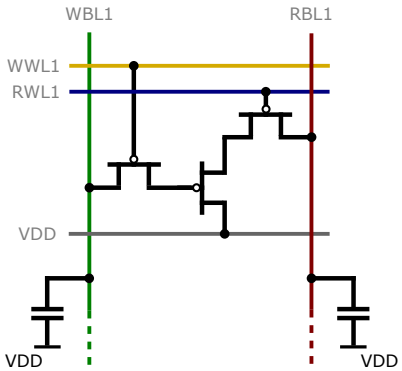

Figure 5.3: Bitcell with parasitic capacitance effect in bitlines. The size of C_{BL} depends on the number of words in the macro

With the results of the bitcell characterization, the layout is generated following the criteria described in section 4.3. Due to it has been designed considering the replication in a macro by the use of the Euler path, it was only necessary to mirror the bitcell layout as function of the macro size. However, it is also necessary to merge the endcells to the generated layout. Endcells are at the boundary cells of the macro layout and their purpose is to make the design DRC clean. They surround the bitcells matrix as shown in Figure 5.4. All memory macros need endcells to ensure the manufacturing plausibility.

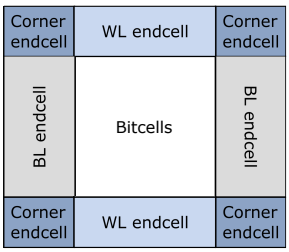

Figure 5.4: Endcells surrounding bitcells matrix layout

Figure 5.5 shows an example of the layout of a macro with a bitcell matrix of 64 words and 32 bits per word including endcells. From this circuit, the parasitic capacitances for the post-layout simulations are also extracted.

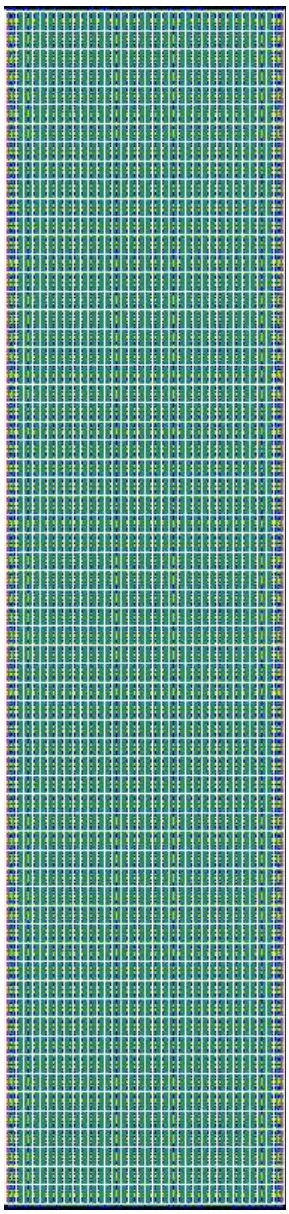

Figure 5.5: 64x32 macro layout

Chapter 6
Results

This chapter is the summary of the results obtained from the implementation of the selected bitcell in the flow implemented in this project. First, the Importance Sampling approach is analysed and compared with Monte Carlo simulation method. Then, the evaluation of the cell architecture and the results obtained in terms of area effect is presented.

6.1 Importance Sampling

In order to validate the IS method, a cell configuration was evaluated simulating both MC and this approach. Figure 6.1 shows the failure probability as function of the number of simulations. The conventional MC method does not find any fail until 10^4 simulations have been completed, and requires high number of samples to converge due to the low failure rate. IS is capable of finding fails since the beginning and is able to converge 100 times faster than Monte Carlo. The proposed IS are evaluated by repeating the simulation 7 times with different seed values.The results show that is able to converge to the same failure probability, which makes it a stable statistical analysis method.

6.2 Cell architecture simulation

This section includes the analysis necessary to perform when selecting a new cell architecture for the use in the eDRAM macro compiler. For this example, a PMOS cell makes the process of writing low level the critical case due to the leakage in a dynamic memory. However, it is necessary to evaluate both processes of storing level high and low.

Considering a bitcell with the architecture of 3T and PMOS transistors, in Figure 6.2 is shown the effect in the SN voltage after writing 0 or 1. It is considered the worst case for each situation. When writing a logic level low, the WBL is set to high and with level high, WBL is set to low. This deteriorates the storage value, and represents the situation when the logic written in the other bitcells connected to the same bitline is the opposite to the stored value in the given cell. Furthermore, the inclusion of variations illustrates the different retention characteristics that a cell may have.

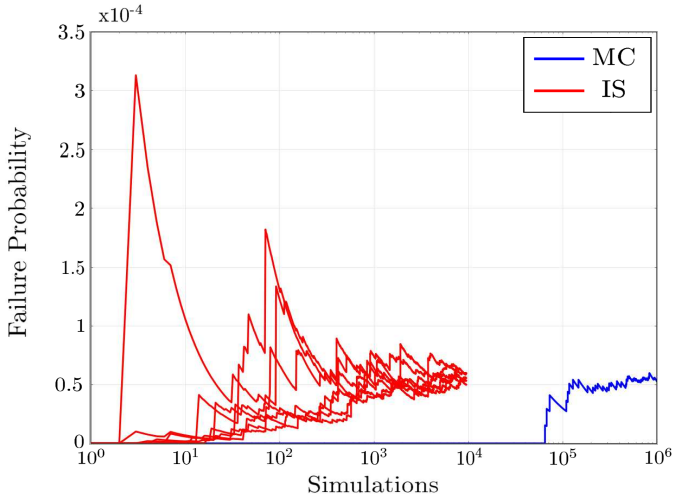

Figure 6.1: MC and IS method applied to the same design evaluation

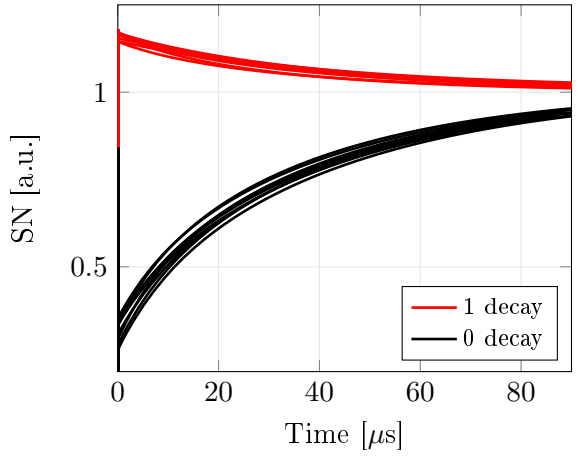

Figure 6.2: Data degradation curves

The voltage in WBL is the parameter that highly deteriorates the stored level in SN. The value of this bitline during idle state determines the leakage of the stored value. Figure 6.3 shows the relation between WBL voltage and leakage current of the SN bitcell. For WBL 0.6 V the leakage current is minimum, the further increase in WBL voltage makes the leakage current to switch its direction.

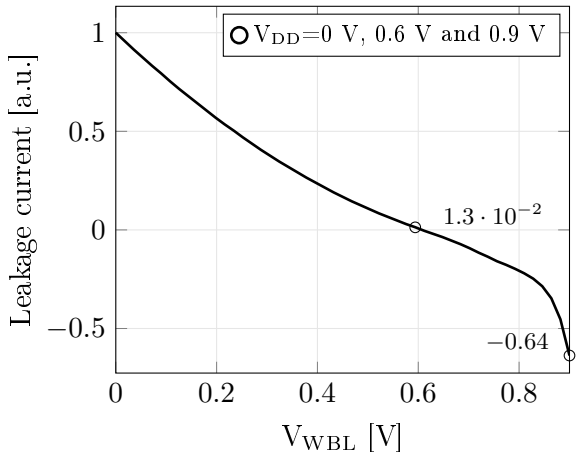

Figure 6.3: SN leakage function of WBL voltage in idle state

6.3 Sense Amplifier specifications

As mentioned in section 4.4, the design of the bitcell as function of DRT depends also on the capabilities of the Sense Amplifier to perform the reading operation. It is necessary that the RBL voltage is outside the SA uncertainty input range after completing the read process. Figures 6.4 and 6.5 show the reading process of stored low and high levels, respectively. The bitcell requirements were the same as described in Table **??**, for DRT of 30 μs. It can be seen in Figure 6.4 that the RBL voltage of reading 0 is higher than the minimum 275 mV. On the other hand, in Figure 6.4, the RBL voltage is lower than the maximum 135 mV when reading 1. This means that the cell is inside the SA range operation specifications.

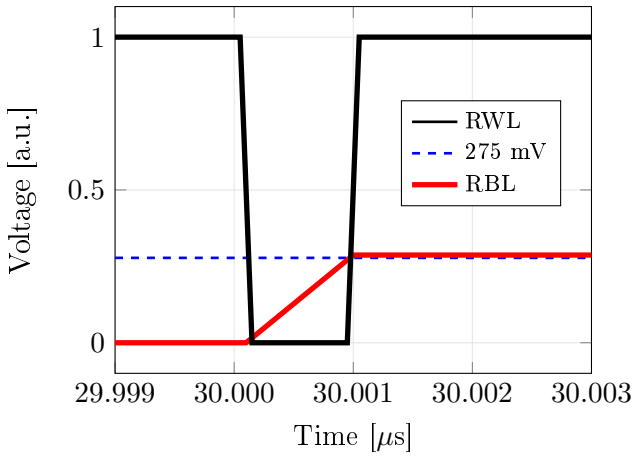

Figure 6.4: Read process for low level stored

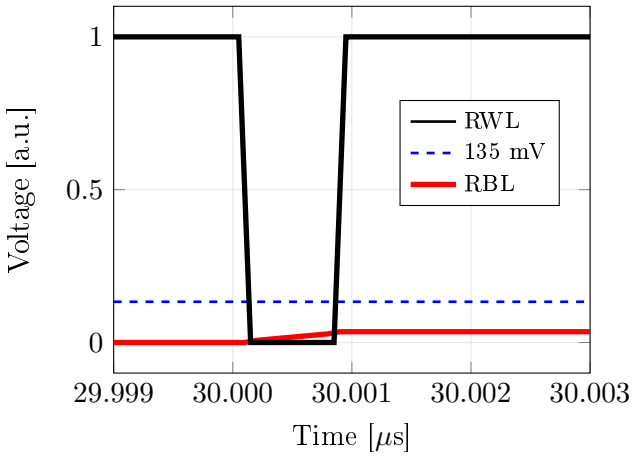

Figure 6.5: Read process for high level stored

6.4 Area comparison

The last step in the flow is the macro generation. As mentioned in section 5.3, the number of words per macro is an important consideration for the writing and reading operations due to the parasitic capacitances generated in WBL and RBL. Figure 6.6 shows the bitline parasitic capacitance effect in relation with the number of words. It is normalized as function of the minimum number of words. The size of the macro is function of 2^N, for that reason the increment of $C_{parasitic}$ can be seen as an exponential dependency with macro size.

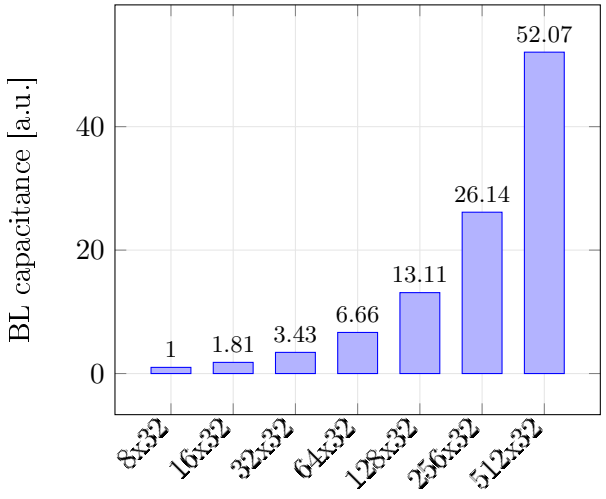

Figure 6.6: Bitline parasitic capacitance dependency with macro size

The comparison in area effect of an eDRAM with DRT 5.6 μs and a SRAM as function of the macro size is shown in Figure 6.7. The use of push rules in the case of the SRAM cell allows to highly minimize its area. For that reason, the main area difference is in the endcells. Increasing the size of the macro, the effect of the endcells is minimized.

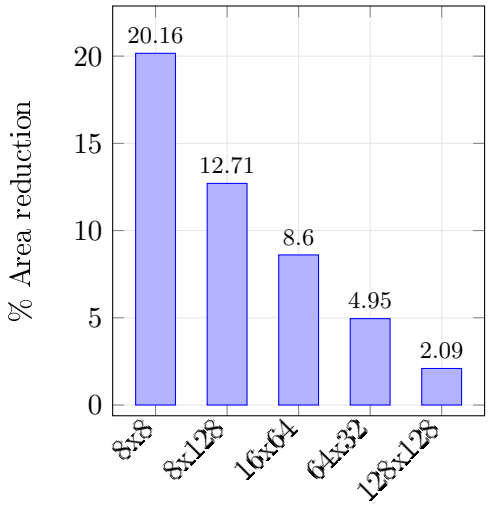

Figure 6.7: Area savings of eDRAM with DRT = 5.6 μs vs SRAM for different macro sizes

Chapter 7
Conclusion

The design flow of Refresh-Free eDRAM for retention timing constraint was analysed in this project. The design process was focused in the inclusion of variations effect in the cell. For an efficient study, different methods for speeding up the statistical analysis were evaluated creating an IS approach for the cell simulation. This approach showed an improve in the speed of 100 times compared to the original Monte Carlo analysis method.

For the evaluation of the created memory macro compiler, an eDRAM cell architecture was tested. With this test case, it was possible to show the steps in the compiler and the required characterization processes until the macro generation. Results section include the considerations required in the cell for the integration with the SA. Furthermore, it is demonstrated with this example the area comparison with an example SRAM implementation.

In conclusion, it can be considered as an efficient approach for the generation of dynamic memories as function of the retention timing constraint. This is an area efficient alternative compared to the traditional SRAM for applications with requirements for data storage for a short duration of time.

Chapter 8

Future work

For the implementation of a Refresh-Free eDRAM, it is important to investigate and evaluate different cell architectures and work on the netlist definition as input for the macro compiler obtained from this project. The bitcell needs to be optimized for DRT and density representing an area reduction compared to traditional SRAM implementation for its manufacturing feasibility.

Another future task is the design of peripheral components for the complete generation of the memory. From this project is extracted the macro of bitcells, it is necessary to include peripheral components of pre-charge, decode and sense circuits for the memory generation.

References

[1] A. Goel, R.K. Sharma and A.K. Gupta, *Nanometer Variation-Tolerant SRAM: Circuits and Statistical Design for Yield*, USA: Springer Science+Business Media New York, 2013, pp. 5-6

[2] R. Giterman, A. Teman, P. Meinerzhagen, L. Atias, A. Burg, and A. Fish, "Single-Supply 3T Gain-Cell for Low-Voltage Low-Power Applications", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol: 24, Issue: 1, January 2016*

[3] C.B.C. Chan, F.R.G. Cruz and W. Chung, "A Single Ended Zero Aware Asymmetric 4T SRAM Cell", *2017, IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*

[4] C. Peng, J. Huang, C. Liu, Q. Zhao, S. Xiao, X. Wu, Z. Lin, J. Chen and X. Zeng, "Radiation-Hardened 14T SRAM Bitcell With Speed and Power Optimized for Space Application", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol: 27, Issue: 2, February 2019*

[5] R. Kanj, R.Joshi, JB Kuang, J.Kim, M. Meterelliyoz, W.Reohr, S. Nassif and K. Nowka, "Statistical Yield Analysis o Silicon-On-Insulator Embedded DRAM", *2009, 10th International Symposium on Quality Electronic Design*

[6] A. Danowitz, K. Kelley, J. Mao, J.P. Stevenson, M. Horowitz, O. Azizi, J.S. Brunhaver II, R. Ho, S. Richardson, O. Shacham and A. Solomatnikov, "Stanford University's VLSI Research Group's CPU Database" [Online], Available: http://cpudb.stanford.edu/

[7] A. Singhee and R.A. Rutenbar, *Extreme Statistics in Nanoscale Memory Design*, USA: Springer Science+Business Media LLC, 2010, pp. 12-15, 72-85

[8] T.S. Doorn, E.J.W. ter Maten, J.A. Croon, A. di Bucchianico and O. Wittich, "Importance Sampling Monte Carlo simulations for accurate estimation of SRAM yield", *2008, 34th European Solid-State Circuits Conference*

[9] J.M. Rabaey, A. Chandrakasan and B Nikolic, *Digital Integrated Circuits: A Design Perspective*, USA: Pearson Education, 2003, pp. 664-669

[10] N.H.E. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective, 3rd Ed.*, Addison-Wesley, 2005, pp. 231-235

[11] B. Mohammadi, *Ultra-low Power Design Approaches in Memories and Assist Techniques*, Lund University, 2017, pp. 26-28

[12] R. Kanj, R. Joshi and S. Nassif, "Mixture Importance Sampling and Its Application to the Analysis of SRAM Designs in the Presence of Rare Failure Events", *2006, 43rd ACM/IEEE Design Automation Conference*

[13] T. Date, S. Hagiwara, K. Masu and T. Sato, "Robust Importance Sampling for Efficient SRAM Yield Analysis", *2010, 11th International Symposium on Quality Electronic Design (ISQED)*

[14] M. Wang, C. Yan, X. Li, D. Zhou and X. Zeng, "High-Dimensional and Multiple-Failure-Region Importance Sampling for SRAM Yield Analysis", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol: 25, Issue: 3, 2017*

[15] N. Jing, L. Jiang, T. Zhang, C. Li, F. Fan and X. Liang, "Energy-Efficient eDRAM-Based On-Chip Storage Architecture for GPGPUs", *IEEE Transactions on Computers, Vol: 65, Issue: 1, 2016*

[16] K.C. Chun, P. Jain, J.H. Lee and C.H. Kim, "A 3T Gain Cell Embedded DRAM Utilizing Preferential Boosting for High Density and Low Power On-Die Caches", *IEEE Journal of Solid-State Circuits, Vol: 46, Issue: 6, 2011*

[17] Y. Xie, K. Cheng and Y. Lin, "A logic 2T gain cell eDRAM with enhanced retention and fast write scheme", *2012, IEEE 11th International Conference on Solid-State and Integrated Circuit Technology*

[18] R. Giterman, A. Fish, N. Geuli, E. Mentovich, A. Burg and A. Teman, "An 800Mhz Mixed-VT 4T Gain-Cell Embedded DRAM in 28nm CMOS Bulk Process for Approximate Computing Applications", *2017, 43rd IEEE European Solid State Circuits Conference*

[19] O. Maltabashi, H. Marinberg, R. Giterman and A. Teman, "A 5-Transistor Ternary Gain-Cell eDRAM with Parallel Sensing", *2018, IEEE International Symposium on Circuits and Systems (ISCAS)*

[20] A. Teman, P. Meinerzhagen, A. Burg and A. Fish, "Review and Classification of Gain Cell eDRAM Implementations", *2012, IEEE 27th Convention of Electrical and Electronics Engineers in Israel*

[21] R. Giterman, A. Teman, P. Meinerzhagen, A. Burg and A. Fish, "4T Gain-Cell with Internal-Feedback for Ultra-Low Retention Power at Scaled CMOS Nodes", *2014, IEEE International Symposium on Circuits and Systems (ISCAS)*

[22] R. Giterman, L. Atias and A. Teman, "Area and Energy-Efficient Complementary Dual-Modular Redundancy Dynamic Memory for Space Applications", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol: 25, Issue: 2, 2017*

[23] S. Yan, D. Li, L. Wang, Y. Xiao and M. Tang, "A Novel Methodology of Layout Design by Applying Euler Path", *2010, 10th IEEE International Conference on Solid-State and Integrated Circuit Technology*

Printed by Tryckeriet i E-huset, Lund 2019

LUND
UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2019-690
http://www.eit.lth.se