

# 國立交通大學

電子工程學系

電子研究所碩士班

碩 士 論 文

嵌入式動態隨機存取記憶體測試方法



Testing Methodology of Embedded DRAMs

研究生:張啓銘

指導教授:趙家佐 教授

中華民國九十七年七月

嵌入式動態隨機存取記憶體測試方法  
Testing Methodology of Embedded DRAMs

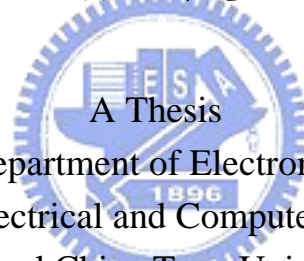
研究生:張啓銘

Student: Chi-Min Chang

指導教授:趙家佐 教授

Advisor: Mango Chia-Tso Chao

國立交通大學  
電子工程學系電子研究所碩士班  
碩士論文



A Thesis  
Submitted to Department of Electronics Engineering  
College of Electrical and Computer Engineering  
National Chiao Tung University  
in partial Fulfillment of the Requirements  
for the Degree of  
Master  
in

Electronics Engineering

July 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年七月

× × × × (嵌入式動態隨機存取記憶體測試方法) × × × ×

學生:張啓銘

指導教授:趙家佐

國立交通大學

電子工程學系電子研究所碩士班



由於嵌入式動態隨機存取記憶體以靜態隨機存取記憶體界面 (所謂的 1T-SRAM) 所構成, 因此嵌入式動態隨機存取記憶體測試混合了動態隨機存取記憶體與靜態隨機存取記憶體的測試, 在這篇論文中, 我們首先針對嵌入式動態隨機存取記憶體測試提出了測試演算法。然後, 對於電閘電晶體漏電機制的理論分析也同樣被提供; 以此為基礎, 我們可以在較高的溫度測試嵌入式動態隨機存取記憶體, 並減少整體測試時間與維持同樣的資料維持時間錯誤涵蓋率。實驗的結果是從一批 16Mb 的嵌入式動態隨機存取記憶體晶片收集得到的。

----- (Testing Methodology of Embedded DRAMs) -----

Student: Chi-Min Chang

Advisor: Dr. Mango Chia-Tso Chao

Department (Institute) of Electronics Engineering  
National Chiao Tung University

## ABSTRACT



The embedded-DRAM testing mixes up the techniques used for DRAM testing and SRAM testing since an embedded-DRAM core combines DRAM cells with an SRAM interface (the so-called 1T-SRAM architecture). In this thesis, we first present our test algorithm for embedded-DRAM testing. A theoretical analysis to the leakage mechanisms of a switch transistor is also provided, based on that we can test the embedded-DRAM at a higher temperature to reduce the total test time and maintain the same retention-fault coverage. The experimental results are collected based on 1-lot wafers with an 16Mb embedded DRAM core.

## 誌 謝

本論文承蒙恩師 趙家佐教授於研究所兩年求學生涯悉心教誨、懇切叮嚀，引領建立記憶體測試正確的觀念及生活態度，並給予學生很大的發展空間。相處二年使得學生在研究及待人處事上得以啟發，師恩浩瀚，永銘於心，特於卷首致衷心之謝意。

論文初成，承蒙晶豪科技副總經理姚忠鼎博士、本校黃俊達教授與江蕙如教授於百忙之中撥冗審閱對論文內容詳加指正，並惠賜寶貴意見，使拙作更臻充實完備，謹此獻上由衷之謝意。

完成論文期間，感謝睿夫學長在記憶體測試及觀念上的傳授；摯友佩娟、曉繼在我遇到挫折時不斷的給我支持和鼓勵及生活上的照顧。

最後更要感謝我的父母多年來的辛苦栽培與呵護，哥哥豪揚與妹妹淑莉對我的支持關懷與包容，使我無後顧之憂，得以順利完成學業，在此由衷感謝。願以此成果與我最親愛的家人及所有關愛我的師長、朋友一同分享。

## 目 錄

中文摘要	.....	i
英文摘要	.....	ii
誌謝	.....	iii
目錄	.....	iv
表目錄	.....	v
圖目錄	.....	vi
一、	緒論	1
二、	嵌入式動態隨機存取記憶體架構	4
三、	嵌入式動態隨機存取記憶體測試方法	8
四、	藉由升溫減少資料維持測試時間	16
五、	結論	25
參考文獻	.....	26

## 表 目 錄

表 1	填入圖3中棋盤式背景的寫入順序與相對應的輸入 .....	7
表 2	不同測試方法下的良率 .....	13
表 3	測試時間的分佈 .....	15
表 4	關於每一種資料維持時間規格與時脈的資料維持測試時間 與總測試時間的比率 .....	15
表 5	計算的等價資料維持時間與相對於 16ms 資料維持時間規 格與溫度 85 度 C 的減少率 .....	22
表 6	關於每一種溫度與資料維持時間的良率 .....	23
表 7	關於每一種資料維持時間規格、時脈與溫度的測試時間的 減少率 .....	24



## 圖 目 錄

圖 1 不同溫度下汲極電流與閘極電壓的關係 .....	3
圖 2 嵌入式動態隨機存取記憶體架構 .....	4
圖 3 陣列擾亂的範例 .....	6
圖 4 動態隨機存取記憶胞的漏電來源 .....	17





## I. INTRODUCTION

Due to the advantages of high density, structure simplicity, low-power consumption, and low cost, DRAM has been the mainstream of the commodity-memory market since its invention by Dr. Dennard [1]. With the continually growing need to an effective and economic embedded-memory core in the SoC era, researchers attempt to carry DRAM's advantages from a commodity memory into a SoC. In the past decade, a lot research effort has been put into the embedded-DRAM (*eDRAM*) technologies, such as deep-trench capacitor with bottle etch [2], planar capacitor [3] [4], shallow trench capacitor [4], and metal-insulator-metal (MIM) capacitor [3] [5], to reduce the process adds to the CMOS process, where the eDRAM is embedded in. The eDRAM technologies are now available in the IC-foundry industry [6][7] and its application includes the products of networking, multimedia handheld devices, gaming consoles, high definition televisions, and so forth.

Unlike integrating a bare DRAM die within a system-in-package or a packaged DRAM on a system board, where the responsibility of testing the commodity DRAM itself is on the memory design company, the responsibility of testing the eDRAM is transferred to the system integrator. Testing embedded-memory cores has been a big challenge for SoC testing due to the difficulty of test isolation and test accessibility [8]. By reducing the tester requirement and enabling the parallel testing of different memory cores, memory built-in-self-test (BIST) circuit is seemed to be the best solution to the embedded memory testing in common consensus today [9][10][11]. Several BIST schemes are proposed for the embedded DRAM testing [12][13][14] [15]. However, these previous works mainly focus on the architecture and the automatic generation of the BIST circuitry. Few discussions on the test algorithms and the test-time overhead resulted from the retention test can be found in the literature for the eDRAM testing.

The classical DRAM testing contains two main steps: the functional test and

the retention test. In the functional test, each functionality of DRAM cells and DRAM's peripheral circuitry are verified. In the retention test, we check whether the data retention time, which is in the order of milliseconds, of each DRAM cell can meet its specification. An industrial test set for DRAM's functional test requires a series of different test algorithms to ensure its complete functionality and coverage [16]. Those algorithms include checkerboard, address complement, March, row/column disturb, self-refresh, XMOVI, butterfly, etc. Applying all of the above test algorithms is time-consuming, thus commodity-DRAM testing heavily relies on the parallel testing capability provided by the memory testers to shorten the average test time of each DRAM chip. In fact, the architecture and functions of most current eDRAM cores use the interface of SRAM (1T-SRAM architecture), which consists of no address multiplexer and can auto-refresh, are simpler than commodity-DRAM. Therefore testing the functionality of eDRAM is simpler than that of commodity DRAM, and hence requires only a shorter test algorithm.

However, testing eDRAM is not completely the same as testing SRAM. Applying only the SRAM test algorithm for eDRAM testing is not sufficient due to the following reasons. First, testing eDRAM needs to consider word-line coupling faults and bit-line toggling faults, but testing SRAM does not. It is because the power/ground shielding technique is commonly used in modern SRAM designs to eliminate the signal disturbance between word-lines or bit-lines, but eDRAM does not have this mechanism. Second, the eDRAM has the functionality of auto-refresh and self-refresh but SRAM does not. Similar to DRAM, eDRAM need to test the retention time, which takes a significant portion of the overall eDRAM test time.

The specification of eDRAM's data-retention time is a constant. As a result, the ratio of this retention test time over the eDRAM test time increases when the clock frequency of the eDRAM increases. It implies that the retention-test time may dominate the eDRAM test time for high-performance eDRAM designs. The data-retention time of an eDRAM cell depends on the leakage current of the

switch transistor in the cell, which is sensitive to the temperature [17][18]. Figure 1 shows that a transistor's leakage current increases dramatically with the increase of temperature [18]. Therefore, by properly increasing the test temperature, the retention test time can be significantly reduced.

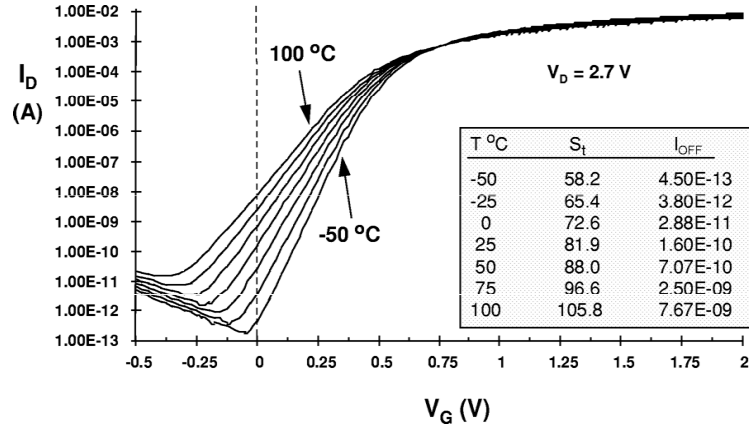


Fig. 1. Relation between  $I_D$  and  $V_G$  associated with different temperatures [18].

In this thesis, we would like to share the experience obtained from testing an industrial eDRAM core. We first discuss the test algorithms used for the eDRAM testing and compare the corresponding yields of different test algorithms through wafer-test results. We then analyze the test time of eDRAM retention test and its ratio to total eDRAM test time. Next, we study the leakage mechanisms of a switch transistor and theoretically compute the leakage-charge equivalence between different temperatures. Based on this leakage-charge equivalence, we can obtain the equivalent retention time used for retention test at different temperatures. We also report the test-time reduction by increasing tester's temperature and validate the equivalent retention-fault coverage through wafer-test results. All reported wafer-test results are collected from 1-lot test wafers. The remaining of this paper is organized as follows. Section II first introduces the embedded DRAM architecture in use. Section III presents a reduced, effective test algorithm for eDRAM. Section IV discusses the leakage mechanism of a switch transistor and analyzes the retention-test time at different temperatures. The conclusion is given in Section V.

## II. OVERVIEW OF EMBEDDED DRAM

Figure 2 shows the block diagram of the 16Mb eDRAM core on our test chips. We will use this eDRAM core as the target instance throughout the rest of this paper. This eDRAM core utilizes a 65nm low-leakage logic process. The size of the eDRAM core is around 4 mm<sup>2</sup>, which contains two symmetric eDRAM arrays with 8Mb data on each. Each array contains 128 banks, and each bank contains 64 word-lines and its own local sense amplifier (LSA). Each word-line on each array is connected to 64 half-words, and the data-width of each half-word is 16 bits. When a word is accessed, its first 16 bits are contributed from the first eDRAM array, and its last 16 bits are from the second array. Note that the layout topology of the eDRAM array utilizes the distributed folding scheme, where the  $i$ th bit of the  $j$ th word is adjacent to the  $i$ th bit of the  $(j+1)$ th word, not the  $(i+1)$ th bit of the original  $j$ th word. Between the two eDRAM arrays is the address decoder including word-line drivers. The control circuit (CTL) and global sense amplifier (GSA) are on the bottom of the eDRAM core.

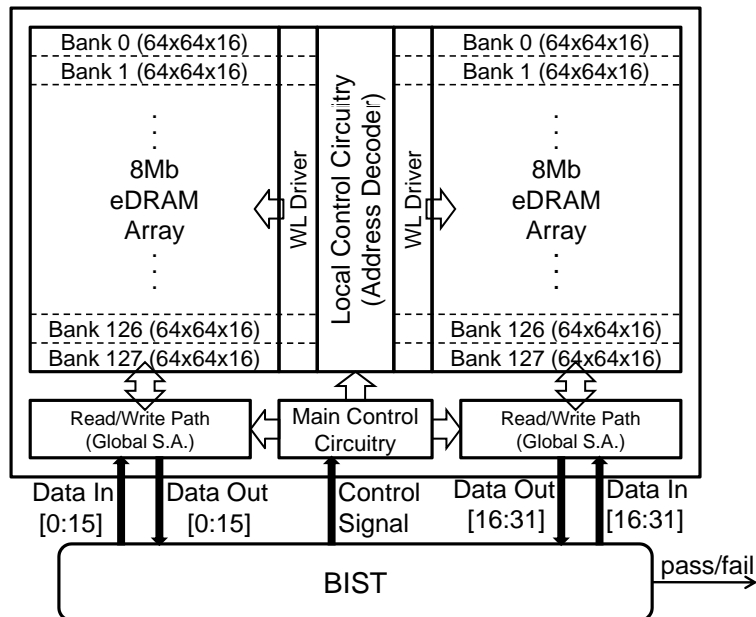


Fig. 2. Embedded-DRAM architecture.

The CTL controls all operations of eDRAM, including read, write, self-refresh, auto-refresh, and any application-dependent operation such as burst-mode read/write or byte read/write. After pre-charge and charge redistribution, the data is first differentiated by LSA, then passed to GSA, and read out through the read/write path. The refresh operation in this eDRAM core can be finished by using the LSA so that refreshing all the words on one word-line (64 words in total) requires only one cycle. Therefore, total 64x128 cycles are required for one refresh operation. When operating at 100 MHz, the bandwidth of this eDRAM core is 3.125 Gb/s (32 bits x 100 MHz).

During the eDRAM testing, the data background written into or read from the memory core should represent cell's physical value instead of its logical value. Therefore, when designing the BIST circuitry, we should consider the physical layout of the word-oriented eDRAM array [19]. The technique of address and data scrambling is commonly used in current memory designs, which can optimize memory's layout geometry, address decoder, cell area, performance, yield, and I/O pin compatibility [19]. The forms of scrambling include folding, address decoder scrambling, contact and well sharing, and bit-line twisting [19].

Figure 3 shows an exemplary scrambling used in current popular eDRAM designs, where the ordering of word-lines in this example are arranged according to the least significant bits of the address. With an SRAM interface, eDRAM utilizes both bit-lines and bit-line-bars to distinguish the data value stored in an eDRAM cell, but a cell's data is only connected to either one of the corresponding bit-line and bit-line-bar. In this example, each word-line connects to two 4-bit words. The first word on a word line uses the 0th, 2nd, 4th, and 6th pairs of the bit-line and bit-line-bar, and the second word uses the 1st, 3rd, 5th, and 7th pairs. By proper arrangement, half of eDRAM cells are connected to bit-line, and the other half to the bit-line-bar. This balances the capacitor of the data-lines and improves the efficiency of eDRAM. As a result, the physical value of those cells connected to

a bit-line-bar is inverse to their logical value. The bit-line twisting shown in the middle of Figure 3 can reduce the coupling capacitance between the bit-line of a cell and the bit-line-bar of the next cell [19]. Each bit-line twist for a given column reverses the physical-value/logical-value relation of the cells below that twist.

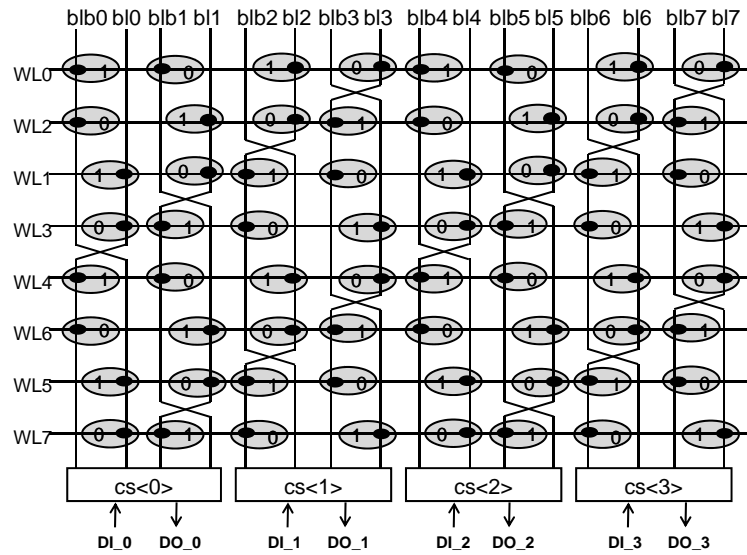
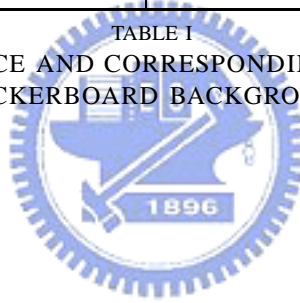


Fig. 3. An exemplary array scrambling.

In the BIST circuitry, a scramble table maps the physical value described in the test algorithm to its corresponding logical value for a given address [19][20]. Those logical values then form the functional test patterns or expected responses during testing. This scramble table can be implemented by a simple two-level logic, whose inputs contains few least significant bits and most significant bits of an address. In addition, when performing Y-direction March algorithm, the sequence of the activated word-lines also needs to follow the physical sequence, not logical address sequence. Thus, the BIST requires another physical-address-mapping circuitry to handle this address scrambling. For instance, Figure 3 shows a checkerboard background for cells' physical values. To fill such a background with an X-direction March algorithm, the sequence of write operations and the corresponding functional inputs are listed in Table I.

write sequence	word-line	word on word-line	functional input
1	0	1st	0101
		2nd	1010
2	2	1st	1010
		2nd	1111
3	1	1st	1111
		2nd	0000
4	3	1st	0000
		2nd	1010
5	4	1st	1010
		2nd	0101
6	6	1st	0101
		2nd	0000
7	5	1st	0000
		2nd	1111
8	7	1st	1111
		2nd	0101

TABLE I  
WRITE-OPERATION SEQUENCE AND CORRESPONDING FUNCTIONAL INPUTS FOR  
FILLING THE CHECKERBOARD BACKGROUND IN FIGURE 3.



### III. THE EDRAM TEST APPROACH

#### A. Current SRAM Test Approach

In this section, we use the March C- algorithm as the basic skeleton of our eDRAM-testing algorithm. March C- algorithm is currently the most widely used test algorithm for SRAM in industry, which can detect stuck-at faults (SAFs), transition faults (TFs), address decoder faults (AFs), inversion coupling faults (CFins), idempotent coupling faults (CFids), and state coupling faults (CFst) [1]. Below shows the element sequence of the March C- algorithm. The complexity of the March C- algorithm is  $10N$ , where  $N$  is the density of the array.

March C- ( $10N$ ):

$\{\updownarrow(wa); \uparrow(ra,wb); \uparrow(rb,wa); \downarrow(ra,wb); \downarrow(rb,wa); \updownarrow(ra)\}$

The notations are defined as follows.

$\updownarrow$ : address direction do not care

$\uparrow$ : address increase

$\downarrow$ : address decrease

a: data background

b: complement background

r: read

w: write



#### B. Embedded-DRAM Test Strategies

Even though the interface of our eDRAM is the same as that of SRAM, applying only the SRAM test algorithm for eDRAM testing is not sufficient. Therefore, on top of this March C- algorithm, we need to add more elements to cover the faults which may not be considered in current SRAM testing but should be considered



in the eDRAM testing, such as data-retention faults, word-line coupling faults, bit-line toggling faults, and stuck-open faults. We also need to test the functionality which eDRAM has but SRAM does not, such as auto-refresh and self-refresh. In the following subsections, we provide the corresponding test strategy for each of the above uncovered faults and functions in the March C- algorithm.

1) *Auto-Refresh and Self-Refresh*: Auto-fresh and self-refresh are two functionalities which eDRAM has but SRAM does not. When the auto-refresh is activated, all eDRAM cells are refreshed after every period of retention-time specification. When the self-refresh is activated, all eDRAM cells are refreshed and the retention-time counter for auto-refresh is reset. Therefore, in the eDRAM testing, the auto-refresh must be always on since the beginning and a self-refresh operation must be performed right before a "ra" element and a "rb" element individually to check the correctness of refreshing both "0" and "1".

2) *Retention Faults*: The retention faults are caused by the cells which can not hold their charge for the specification-defined retention time. To test retention faults, we need to perform a self-refresh followed by a delay element, which will delay the next operation for the specification-defined retention time. At the same time that the self-refresh is performed, the counter of the auto-refresh is reset. So after the delay element ends, the data will be auto-refreshed again. Then a read operation is performed to check if any retention fault occurs during the delay element. During this retention test, the checkerboard background should be applied because this background can exacerbate the leakage and help to catch a retention fault. Note that the checkerboard background here refers to data's physical values, not logical values. Also, we need to perform this retention test to both the data-background and its complement.

3) *Word-line-Coupling Faults*: In modern SRAM designs, the power/ground shielding technique is used to eliminate the signal disturbance between word-lines or bit-lines, and hence we seldom consider the word-line-coupling faults in SRAM

testing. However, for eDRAM design, such technique cannot be applied due to its high-density requirement. In addition, the capacitive loading of a word-line in eDRAM is relatively large because more words are connected to a word-line in eDRAM than in SRAM. Word-lines are made of polysilicon that has much higher resistance than metal line. When a word-line is turned off too slowly due to its large RC delay, the voltage of the neighboring word-line might couple capacitively a voltage to the original word-line, resulting in a wrong state on the original word-line. In this case, a wrong data would be read from or write into the cells if a cell's data on the original word-line is different from that of its adjacent word-line, such a scenario is easier to happen and test by the checkerboard background. Therefore, to detect word-line-coupling faults, a Y-direction MATS algorithm with a checkerboard background may be utilized. The sequence of a MATS algorithm is shown as follows. Its complexity is  $4N$ .



MATS ( $4N$ ):

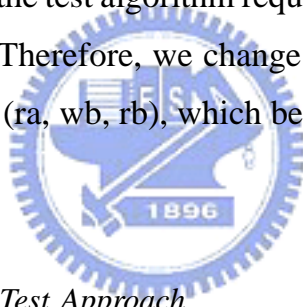
$\{\uparrow\downarrow(wa); \uparrow(ra,wb); \downarrow(rb)\}$

Note that the Y-direction sequence refers to the physical word-line sequence, not the logical address sequence. For example in Figure 3, the physical word-line sequence is "WL 0, 2, 1, 3, 4, 6, 5, 7", not "WL 0, 1, 2, 3, 4, 5, 6, 7". This address scrambling in Y-direction needs to be considered in the BIST circuitry.

4) *Bit-line Toggling Faults*: Testing SRAM needs not consider the bit-line toggling because of its power/ground shielding mechanism. A bit-line-toggling fault occurs when the bit-line or bit-line-bar of a cell is close to the bit-line or bit-line-bar of its adjacent cell, and these two adjacent lines have opposite data values. Because of higher density, one cell's bit-line or bit-line-bar is closer to its adjacent cell's bit-line or bit-line-bar than that in SRAM, resulting higher probably of bit-line toggling fault. In order to create this scenario for each pair of adjacent cells, we need to perform the solid data-background because of the array scrambling as

shown in Figure 3. Therefore, the testing algorithm for eDRAM testing needs to cover bit-line-toggling faults, meaning that the proposed algorithm have to apply the solid data-background.

5) *Stuck-Open Faults*: Stuck-open fault (*SOF*) occurs when the resistance between bit-line and switch transistor, switch transistor and storage capacitor, or storage capacitor and ground is large. In this case, the data is hard to write into or read out from cells. Modern SRAM designs do not have this problem but some eDRAM cores do. SOFs can be detected at the same time as SAFs are detected when the sense amplifier is transparent to stuck-open faults. It means that the second element in March C- algorithm, (ra, wb), can already detect the SOFs in this case. When the sense amplifier is latch-based and thus not transparent to stuck-open faults due to the presence of the data latch, the test algorithm requires an element of (read, write, read) to detect the SOFs [1]. Therefore, we change the second element in March C- algorithm from (ra, wb) to (ra, wb, rb), which becomes the extended March C- algorithm.



### C. Proposed Embedded-DRAM Test Approach

In this section, we summarize the test strategies discussed in Section III-B to form the final test approach for an eDRAM core. This test approach applies an X-direction extended March C- algorithm with solid data-background as well as a Y-direction MATS algorithm with checkerboard data-background. Also, we test the self-refresh operation in the extended March C- algorithm and the retention faults in the MATS algorithm. The auto-refresh is always on in both algorithms. The detail steps of the March C- and MATS algorithms are described as follows.

X-direction Extended March C- with solid background (11N):

$$\{\uparrow(wa); \uparrow(ra, wb, rb); (SR); \uparrow(rb, wa); \downarrow(ra, wb); \downarrow(rb, wa); (SR); \downarrow(ra)\}$$

Y-direction MATS with checkerboard background (4N):

$\{\uparrow(wa);SR;del;\uparrow(ra,wb);SR;del;\downarrow(rb,wa)\}$

SR: self-refresh.

del: delay element which stops for the period of the retention time defined in the specification.

The above X-direction extended March C- algorithm covers the stuck-open faults by the element (ra,wb,rb). It also tests the functionality of self-refresh and auto-refresh. The above Y-direction MATS algorithm tests the word-line-coupling faults by the Y-direction elements and checkerboard data-background. It also tests the retention faults by inserting the sequence of SR and del twice. The bit-line-toggling faults are covered by the solid-background operations in the extended March C- algorithm and the checkerboard-background operations in the Y-direction MATS algorithm.

From coverage's point of view, the two self-refresh operations in the extended March C- algorithm seem redundant since two self-refresh operations are also performed in the MATS algorithm for the retention test. However, we keep the first two self-refresh operations in our first tape-out to differentiate the detection of self-retention faults from that of the data-retention faults. These two self-refresh operations in the extended March C- algorithm can be further removed to speed up the test time if the diagnosis requirement is low.

#### *D. Experimental Results*

We apply the test set of the following three test approaches individually to the same eDRAM cores on 1-lot wafers through external testers, not BIST circuitry.

1. The proposed test approach

2. X-direction March C- with solid background plus  
Y-direction MATS with CHK background

3. X-direction March C+ with solid background plus  
Y-direction MATS with CHK background

The detail of March C+ (14N) is as follow:

$$\{\uparrow(wa); \uparrow(ra, wb, rb); \uparrow(rb, wa, ra); \downarrow(ra, wb, rb); \downarrow(rb, wa, ra); \uparrow(ra)\}$$

The difference between proposed approach and the others is on their March algorithms in use. Approach 2 uses the basic March algorithm described in Section III-A and approach 3 uses the default March algorithm generated by a commercial memory-BIST tool, *Memory BIST Architecture* [21]. Note that we turn off the retention test in this experiment to save its test time. The experimental results containing the retention test will be discussed later in the Section IV.

Table II lists the yield of the above three test approaches. Our proposed approach and Approach 3 result in the same yield while the Approach 2 results in a higher yield. This result implies that only applying March C- may miss certain faults and lead to higher test escape. The proposed approach can achieve the same level of fault coverage with Approach 3. However, the proposed approach only requires a 11N extended March C- algorithm but Approach 3 requires a 14N March C+ algorithm. This result shows that the general SRAM algorithm, March C- (10N), cannot provide sufficient fault coverage, and the default March algorithm generated by a commercial tool, March C+ (14N), is redundant in our eDRAM testing.

Test Approach	proposed	2	3
yield (%)	96.9	97.8	96.9

TABLE II  
YIELD OF DIFFERENT TEST APPROACHES.

### E. Test Time Analysis for Proposed Test Approach

The total test time of the proposed test approach ( $T_{test}$ ) is the summation of the test time on retention test ( $T_{RT}$ ), read/write operations ( $T_{R/W}$ ), self-refresh ( $T_{SR}$ ), and auto-refresh ( $T_{AR}$ ).

$$T_{test} = T_{RT} + T_{R/W} + T_{SR} + T_{AR} \quad (1)$$

where

$$T_{RT} = 2 \times T_{del} \quad (2)$$

$$T_{R/W} = N_{WORDS} \times N_{R/W} \times T_{CYCLE} \quad (3)$$

$$T_{SR} = N_{WL} \times N_{SR} \times T_{CYCLE} \quad (4)$$

$$T_{AR} = N_{WL} \times N_{AR} \times T_{CYCLE} \quad (5)$$

$T_{del}$  : time of one (del) element

$T_{CYCLE}$  : cycle time

$N_{WORDS}$  : number of words

$N_{R/W}$  : number of reads and writes

$N_{WL}$  : number of word-lines

$N_{SR}$  : number of self-refreshes

$N_{AR}$  : number of total auto-refreshes

$T_{del}$  is equal to the retention-time specification, and  $N_{AR}$  is equal to the runtime divide by the specified retention time.

Table III lists the test time spent in each component of the proposed approach, given a 50MHz clock frequency and a 16ms retention-time specification. In this case, the ratio of retention-test time to total test time is 16.5%.

In current eDRAM designs, the target clock frequency can be higher than the 50MHz used in Table III. Table IV shows the ratio of the retention-test time to total eDRAM-test time for different clock frequencies and different retention-time

	retention	read & write	self-refresh	auto-refresh	total	retention ratio
test time (ms)	32	160	0.6	1.3	193.9	16.5%

TABLE III  
TEST TIME DISTRIBUTION OF THE PROPOSED TEST APPROACH.

specifications. As the results show, the ratio of the retention-test time increases when the clock frequency increases, and gradually dominates the total eDRAM-test time. If the retention time is defined longer in the specification, this ratio would be even higher. For the case that clock frequency is 200MHz and the defined retention time is 32ms, this retention-test-time ratio can be up to 61.4%. Therefore, reducing the retention-test time can significantly reduce the total eDRAM-test time. In Section IV, we will attempt to increase the temperature to further reduce the retention-test time.

retention time in spec. (ms)	retention test time (ms)	clock rate (MHz)	total test time (ms)	ratio of retention-test time to total test time
16	32	50	193.9	16.5%
		100	112.5	28.4%
		200	72.2	44.3%
32	64	50	224.9	28.5%
		100	144.3	44.4%
		200	104.2	61.4%

TABLE IV  
RATIO OF RETENTION-TEST TIME TO TOTAL TEST TIME W.R.T. EACH RETENTION-TIME SPECIFICATION AND CLOCK RATE.

Another way to further reduce the total test time is to apply the burst mode operation, if the eDRAM core supports, for a single-operation March element, such as the  $\uparrow(wa)$  and  $\uparrow(ra)$  in the extended March C- algorithm. However, this reduction is still limited since most elements contain more than one operations.

#### IV. REDUCING RETENTION-TEST TIME BY INCREASING TEMPERATURE

For an eDRAM cell, its data-retention time is determined by the leakage of its switch transistor, which increases along with the increase of the temperature. In the eDRAM testing, we attempt to raise the temperature to increase transistor's leakage current, which shortens the data-retention time of a cell. Therefore, at a higher temperature, the delay element used for retention test can be specified shorter since a retention fault can be detected within a shorter period of time than that at the original reference temperature. However, if the new specified retention time is too low, some retention faults may be able to escape, resulting in a higher defect level. On the contrary, if it is too high, the retention time of an eDRAM cell is over-tested, resulting in a yield lost.

In order to specify an appropriate retention time for the delay element at a higher temperature, we need to calculate the time at a given temperature during that the leakage of a switch transistor is equivalent to the leakage during the specified retention time at the reference temperature, which is defined as 85°C in our specification. This time is defined as the *equivalent retention time* for a given temperature, which implies that a eDRAM cell loses its data after the specified retention time at 85°C if and only if this cell will lose its data after the equivalent retention time at the given temperature.

In the following of this section, we first study different leakage mechanisms of a switch transistor and their sensitivity to the temperature. Based on this leakage analysis, we then calculate the equivalent retention time. Last, the experimental results of using different equivalent retention time at different temperatures are presented. We will also compare the total test-time reduction by increasing the temperature.



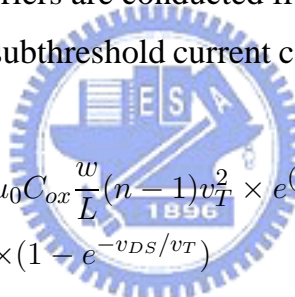


where  $A = \frac{\sqrt{2m^*}q^3}{4\pi^3\hbar^2}$ , and  $B = \frac{4\sqrt{2m^*}}{3q\hbar}$ ;  $m^*$  is the effective mass of electron;  $E_g$  is the energy-band gap;  $V_{app}$  is the applied reverse bias;  $E$  is the electric field at the junction;  $q$  is the electronic charge; and  $\hbar$  is the reduced Planck's constant. The electric field at the junction is

$$E = \sqrt{\frac{2qN_aN_d(V_{app} + V_{bi})}{\epsilon_{si}(N_a + N_d)}} \quad (7)$$

where  $N_a$  and  $N_d$  are the doping in the p and n side, respectively;  $\epsilon_{si}$  is permittivity of silicon;  $V_{bi}$  is the built in voltage across the junction.

2) *Subthreshold leakage*: Subthreshold leakage occurs when gate voltage is below  $V_{th}$ . In the weak inversion, the diffusion current occurs in the subthreshold conduction when the minority carriers are conducted from channel region and exist in channel depletion layer. This subthreshold current can be expressed as follow [22]:



$$I_{sub} = \mu_0 C_{ox} \frac{w}{L} (n-1) v_T^2 \times e^{(V_g - V_{th})/nv_T} \times (1 - e^{-v_{DS}/v_T}) \quad (8)$$

where

$$n = 1 + \frac{3t_{ox}}{W_{dm}} \quad (9)$$

where  $V_{th}$  is the threshold voltage;  $v_T = K\theta/q$  is the thermal voltage,  $\theta$  is temperature;  $C_{ox}$  is the gate-oxide capacitance;  $\mu_0$  is the zero-bias mobility;  $n$  is the subthreshold swing coefficient (also called body effect coefficient);  $W_{dm}$  is the maximum depletion-layer width;  $t_{ox}$  is the gate-oxide thickness.

3) *Gate tunneling current*: The high electric field coupled with low oxide thickness causes tunneling of electrons both from substrate to gate and from gate to substrate, resulting in the gate-oxide-tunneling current. The direct tunneling mechanism occurs in more advanced devices because the potential drop across the oxide

is smaller than the barrier height of Si-SiO<sub>2</sub>. The current density of direct tunneling can be expressed as follows [22]:

$$J_{DT} = AE_{ox}^2 \exp\left(-\frac{B(1 - (1 - \frac{V_{ox}}{\phi_{ox}})^{3/2})}{E_{ox}}\right) \quad (10)$$

where  $A = \frac{q^3}{16\pi^2\hbar\phi_{ox}}$  and  $B = \frac{4\sqrt{2m^*}\phi_{ox}^{3/2}}{3\hbar q}$ ;  $E_{ox}$  is the electric field across the oxide.

When a DRAM cell stores "1", its bias condition is illustrated in Figure 4.  $V_G = 0$  and  $V_{DS} = V_{DD}$  induce a subthreshold current.  $V_{DB} = V_{DD}$  means that drain-substrate is reverse-biased, which induces BTBT leakage. In addition, the direct tunneling current also may occur because the voltage across the intersection of drain and gate is equal to  $V_{DD}$ . Hence, the total leakage current  $I_{leak}(\theta)$  of the switch transistor for a given temperature  $\theta$  can be expressed as

$$I_{leak}(\theta) = I_{sub} + J_{DT} \times A_{DT} + J_{BTBT} \times A_{BTBT} \quad (11)$$

where  $A_{DT}$  and  $A_{BTBT}$  is the tunneling area of direct tunneling and BTBT.

Note that this leakage is actually a function of temperature. The following subsection discusses those temperature-dependent parameters in the above leakage equations. In addition, the leakage for the storage capacitor itself is small when using a high- $k$  material and hence can be omitted in our analysis.

### B. Temperature-Dependent Parameters in Leakage

Different leakage-current sources have different temperature dependence. In the following, we list the temperature-dependent parameters in above three leakage equations and discuss the magnitude of their dependency to the temperature  $\theta$ .

1) *Energy-band gap*( $E_g$ ): The energy-band gap may be narrowed by the increase of temperature within an order of  $10^{-4}\theta^2$ .

2) *Junction electric field*( $E$ ): The junction electric field coupled with the doping concentration may be influenced by the temperature, but it is more dependent on the junction voltage.

3) *Mobility*( $\mu_0$ ): The increase of temperature results in the reduction of mobility. The degradation of mobility is direct proportional to  $\theta^{1.5}$ .

4) *Thermal voltage*( $V_T$ ): The thermal voltage is linearly proportional to the temperature, which results in an exponential growth of the subthreshold leakage.

5) *Threshold voltage*( $V_{th}$ ): The increase of temperature causes more carriers on the channel, which reduces the threshold voltage and hence increases the subthreshold leakage.

6) *Barrier height*( $\phi_{ox}$ ): The barrier height decreases when temperature increases, which is proportional to  $10^{-4}\theta$ .

In summary, the direct-tunneling current is invariant to the temperature since the barrier height and potential drop across oxide are invariant to the temperature. The BTBT leakage may vary with the temperature but only in a small order. The subthreshold leakage increases significantly along with the increase of the temperature due to the decrease of  $V_{th}$  and the increase of thermal voltage. Even though the direct-tunneling current and BTBT current are not sensitive to the temperature, both of them should still be considered in our leakage analysis since they contribute a significant portion of the total leakage at the normal temperature especially in advanced process technologies [17].

### C. Analysis of Equivalent Retention Time

To calculate the equivalent retention time for a target temperature, we first calculate the total amount of charge ( $Q_{total}$ ) leaked from the storage capacitor during the retention-time specification ( $T_{ref}$ ) at the reference temperature ( $\theta_{ref}$ ), i.e., 85°C. Then the leakage during the equivalent retention time ( $T_{eqv}$ ) at the target temperature ( $\theta_{tgt}$ ) has to be equivalent to  $Q_{total}$ , which is expressed in Equation 12.

$$Q_{total} = I_{leak}(\theta_{ref}) \times T_{ref} = I_{leak}(\theta_{tat}) \times T_{eqv} \quad (12)$$

Therefore, the equivalent retention time  $T_{eqv}$  at the target temperature  $\theta_{tgt}$  can be obtained by Equation 13.

$$T_{eqv} = \frac{I_{leak}(\theta_{ref}) \times T_{ref}}{I_{leak}(\theta_{tat})} \quad (13)$$

The parameters used in the leakage calculation are listed as follows, which are provided by the IC foundry and may vary from different process technologies.

Mobility ( $\mu_0$ ) :  $(230 \sim 250) \times 10^{-4}(m^2/V \times s)$

Oxide Capacitance ( $C_{ox}$ ) :  $(1.1 \sim 1.3) \times 10^{-2}(F/m^2)$

Oxide Thickness ( $T_{ox}$ ) :  $(2 \sim 3) \times 10^{-9}(m)$

Channel Width ( $W$ ) :  $(0.8 \sim 1) \times 10^{-7}(m)$

Channel Length ( $L$ ) :  $1.3 \times 10^{-7}(m)$

Subthreshold Swing ( $n$ ) :  $1.1 \sim 1.5$

Thermal Voltage ( $V_T$ ) :  $K/11600(V)$

Threshold Voltage ( $V_{th}$ ) :  $0.4 \sim 0.6(V)$

Supply Voltage ( $V_{DD}$ ) :  $1.2(V)$

Barrier Height ( $\phi_{ox}$ ) :  $3.1 \sim 3.2(eV)$

Energy Band-gap ( $E_g$ ) :  $1.17 - \frac{4.73 \times 10^{-4} \times K^2}{K+636}(eV)$

Doping Concentration: about  $10^{24}(m^{-3})$

Table V lists the calculated equivalent retention time and its reduction ratio to the original specification-defined retention time associated with each given temperature. The retention-time specification ( $T_{ref}$ ) is 16ms at the reference temperature

( $\theta_{ref}$ ) 85°C. As the results shows, the retention-time reduction is close to 50% when raising the temperature to 105°C, and 65% when 120°C, respectively. It implies that the retention-test time can be significantly reduced by raising the temperature.

	90°C	95°C	100°C	105°C	110°C	115°C	120°C
retention time (ms)	13.57	11.55	9.87	8.47	7.29	6.30	5.47
reduction ratio	15.2%	27.8%	38.3%	47.1%	54.4%	60.6%	65.8%

TABLE V  
THE CALCULATED EQUIVALENT RETENTION TIME AND ITS REDUCTION TO THE RETENTION-TIME SPECIFICATION 16MS AT 85°C.

#### D. Experimental Results

In the following experiment, we apply our proposed test algorithm (described in Section III) on the eDRAM cores of 1-lot test wafers repeatedly with different retention-time specifications at different temperatures. In each time of the eDRAM testing, the delay element needs to match the retention-time specification. Table VI shows the corresponding yield for each retention-time specification and temperature. As the results show, the yield reaches 86.5% with 16ms retention time at 85°C. Also, the same yield is first-reached with 12ms retention time at 95°C and 8ms retention time at 105°C. This result implies that the eDRAM cells which hold their charge for 16ms at 85°C can hold their charge for 12ms at 95°C and for 8ms at 105°C, respectively. This result approximately matches the calculated equivalent retention time listed in Table V, where the equivalent retention time for 95°C and 105°C is 11.55ms and 8.49ms, respectively.

Note that we are not suggesting to directly use the calculated equivalent retention time during the eDRAM testing. The equivalent retention time used in practice should be verified through real silicon experiments. For the IC foundry providing eDRAM cores, a table of equivalent retention time associated with different temperatures can be built through a similar experiment as shown in Table VI. However,

retention time (ms)	85°C	95°C	105°C
16	86.5%	83.1%	77.5%
14	86.5%	84.3%	82.0%
12	86.5%	86.5%	83.1%
10	86.5%	86.5%	83.1%
8	86.5%	86.5%	86.5%
6	86.5%	86.5%	86.5%
4	86.5%	86.5%	86.5%

TABLE VI  
YIELD W.R.T. EACH TEMPERATURE AND RETENTION-TIME SPECIFICATION.

it may take weeks or even longer to build a complete yield table with respect to each temperature and each retention-time specification. The cost of repeatedly testing the same wafers should be considered. This cost limitation is also the reason why the resolution of the retention time in Table VI is in 2ms, not in a smaller, more accurate unit of time. Therefore, our theoretical calculation of the equivalent retention time can be used as an efficient guideline during the above process of searching the equivalent retention time with silicon experiments, which can save the high cost of repeatedly testing a significant number of test wafers.

Table VII further shows the total eDRAM-test-time reduction which can be achieved by increasing the testing temperature. In Table VII, Column 4, Column 5, and Column 6 lists the equivalent retention time, retention-test time, and total eDRAM-test time, respectively, associated with each retention-time specification at 85°C, clock frequency, and temperature. Column 7 list the total eDRAM-test-time reduction achieved by using the equivalent retention time at each temperature compared to the total test time at 85°C. As the results show, this total eDRAM-test-time reduction increases when the temperature, clock frequency, or retention-time specification increases. The reduction ratio can be up to 37.2% by increasing 30°C at temperature when the retention-time specification and clock frequency are 32ms and 200MHz, respectively.

Note that at a higher temperature, its equivalent retention time decreases, which

retention time in spec.	clock rate (MHz)	temp. (°C)	equivalent retention time (ms)	retention test time (ms)	total test time (ms)	test-time reduction to 85°C
16ms	50	85	16	32	193.9	-
		95	11.55	23.1	185.3	4.4%
		105	8.47	16.94	180.2	7.1%
		115	6.3	12.6	177.0	8.7%
	100	85	16	32	112.5	-
		95	11.55	23.1	103.6	7.9%
		105	8.47	16.94	97.8	13.1%
		115	6.3	12.6	93.7	16.7%
	200	85	16	32	72.2	-
		95	11.55	23.1	63.3	12.3%
		105	8.47	16.94	57.1	20.9%
		115	6.3	12.6	53.1	26.5%
32ms	50	85	32	64	224.9	-
		95	23.1	46.2	207.3	7.8%
		105	16.94	33.88	195.8	12.9%
		115	12.61	25.22	187.4	16.7%
	100	85	32	64	144.3	-
		95	23.1	46.2	126.6	12.3%
		105	16.94	33.88	114.4	20.7%
		115	12.61	25.22	105.7	26.7%
	200	85	32	64	104.2	-
		95	23.1	46.2	86.4	17.1%
		105	16.94	33.88	74.1	28.9%
		115	12.61	25.22	65.4	37.2%

TABLE VII  
TEST TIME REDUCTION W.R.T. EACH RETENTION-TIME SPECIFICATION, CLOCK RATE, AND TEMPERATURE.

results in more frequent auto-refresh operations. Fortunately, the time consumed by a refresh operation is short and does not affect test-time reduction too much. In addition, the temperature discussed here is for wafer testing. If we want to test the data retention after package, the temperature under consideration should be the temperature inside the package, not just tester's temperature. The temperature inside the package is higher than that outside the package. The table to map package's outside temperatures to its insides temperature can be obtained from the package providers.



## V. CONCLUSION

Even though an SRAM interface is used in an eDRAM core, testing an eDRAM core is more than just testing a SRAM core. In this thesis, we have discussed the testing strategies to detect the faults which may not be considered in SRAM testing but should be covered in eDRAM testing. We then proposed an eDRAM-testing approach to target those uncovered faults on top of a SRAM testing approach. Also, we analyze the relation between switch transistor's leakage and temperature. Based on that, we can theoretically calculate the equivalent retention time for different temperatures which can be adopted to reduce the retention-test time. The results were validated through the experiment of 1-lot test wafers.



## REFERENCES

- [1] A. J. van de Goor, "Testing Semiconductor Memories, Theory and Practice," Gouda, The Netherlands: ComTex, 1998.
- [2] G. Wang, et al., "A  $0.127 \mu\text{m}^2$  High Performance 65nm SOI Based embedded DRAM for on-Processor Applications," *International Electron Devices Meeting*, 11-13 Dec. 2006, pp. 1-4.
- [3] E. Gerritsen, et al., "Evolution of Materials Technology for Stacked-Capacitors in 65 nm Embedded-DRAM," *Solid-State Electronics*, vol. 14, 2005, pp. 1767-1775.
- [4] M.-E. Jones, "1T-SRAM- $Q^{TM}$ : Quad-Density Technology Reins in Spiraling Memory Requirements," *Mosys, Inc.*, Retrieved on 2007-10-06.
- [5] A. Berthelot, C. Caillat, V. Huard, S. Barnola, B. Boeck, H. Del-Puppo, N. Emonet, F. Lalanne, "Highly Reliable TiN/ZrO<sub>2</sub>/TiN 3D Stacked Capacitors for 45 nm Embedded DRAM Technologies," *Proceeding of Solid-State Device Research Conference*, Sept. 2006, pp. 343-346.
- [6] "TSMC Embedded High Density Memory," <http://www.tsmc.com/>.
- [7] "0.13 Micron SoC Process Technology," <http://www.umc.com/>.
- [8] "A D&T Roundtable: Testing Mixed Logic and DRAM Chips," *IEEE Design & Test of Computers*, vol. 15, no. 2, Apr. June 1998, pp. 86-92.
- [9] C. Cheng, C.-T. Huang, J.-R. Huang, C.-W. Wu, C.-J. Wey, and M.-C. Tsai, "BRAINS: A BIST compiler for embedded memories," *Proceedings of IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, Yamanashi, Oct. 2000, pp. 299-307.
- [10] J.-F. Li, R.-S. Tzeng and C.-W. Wu, "Diagnostic Data Compression Techniques for Embedded Memories with Built-In Self-Test," *J. Electronic Testing: Theory and Application*, vol.18, no.4, Aug. 2002, pp. 515-527.
- [11] B. Nadeau-Dostie, A. Silburt, V.K. Agarwal, "Serial Interfacing for Embedded Memory Testing," *IEEE Design & Test of Computers*, vol. 7, no. 2, Apr 1990, pp. 52-63.
- [12] C.-T. Huang, J.-R. Huang, C.-F. Wu, C.-W. Wu, and T.-Y. Chang, "A Programmable BIST Core for Embedded DRAM," *IEEE Design & Test of Computers*, vol. 16, no. 1, Jan.-Mar. 1999, pp. 59-70.
- [13] J. E. Barth, et al., "Embedded DRAM Design and Architecture for the IBM 0.11- $\mu\text{m}$

- ASIC Offering," *IBM Journal of Research and Development*, vol. 46, no. 6, Nov. 2002, pp. 675-689.
- [14] S. Miyano, K. Sato, K. Numata, "Universal Test Interface for Embedded-DRAM Testing," *IEEE Design & Test of Computers*, vol. 16, no. 1, Jan.-Mar. 1999, pp. 59-70.
- [15] N. Watanabe, F. Morishita, Y. Taito, A. Yamazaki, T. Tanizaki, K. Dosaka, Y. Morooka, F. Igaue, K. Furue, Y. Nagura, T. Komoike, T. Morihara, A. Hachisuka, K. Arimoto, and H. Ozaki, "An Embedded DRAM Hybrid Macro with Auto Signal Management and Enhanced-on-Chip Tester," *Proceedings of the IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, 2001, pp. 388-389.
- [16] A. J. van de Goor, "An Industrial Evaluation of DRAM Tests," *IEEE Design & Test of Computers*, vol. 21, no. 5, Sept.-Oct. 2004, pp. 430-440.
- [17] S. Mukhopadhyay, A. Raychowdhury, K. Roy, "Accurate Estimation of Total Leakage in Nanometer-Scale Bulk CMOS Circuits Based on Device Geometry and Doping Profile," *IEEE Transaction Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 3, March 2005, pp. 363-381.
- [18] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand, "Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits," *Proceeding of the IEEE*, vol. 91, No. 2, Feb. 2003, pp. 305-327.
- [19] A.J. van de Goor and I. Schanstra, "Address and Data Scrambling: Causes and Impact on Memory Tests," *Proc. 1st IEEE Int'l Workshop on Electronic Design, Test and Application (DELTA 02)*, IEEE Press, 2002, pp. 128-136.
- [20] K.-L. Cheng, M.-F. Tsai, and C.-W. Wu, "Neighborhood pattern-sensitive fault testing and diagnostics for random-access memories," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, no. 11, Nov. 2002, pp. 1328-1336.
- [21] MBIST Architech Reference Manual, V8, Mar. 2003.
- [22] Y. Taur and T. H. Ning, "Fundamentals of Modern VLSI Devices," New York: Cambridge Univ. Press, 1998.