DESIGN AND TEST METHODOLOGIES WITH STATISTICAL ANALYSIS FOR RELIABLE MEMORY AND PROCESSOR IMPLEMENTATIONS

A Dissertation
Presented to
The Academic Faculty

by

Woongrae Kim

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
May 2016

DESIGN AND TEST METHODOLOGIES WITH STATISTICAL ANALYSIS FOR RELIABLE MEMORY AND PROCESSOR IMPLEMENTATIONS

Approved by:

Dr. Linda Milor, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. David E Schimmel
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Abhijit Chatterjee
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Haomin Zhou
School of Mathematics
Georgia Institute of Technology

Dr. Azad J Naeemi
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: 03/17/2016

ACKNOWLEDGEMENTS

I would like to thank Prof. Milor for her professional guidance on my research and dissertation. Since I have joined in her group in 2013, I have learned novel approaches to solve difficult problems and to explore academic achievements.

Also, I would like thank Prof. Chatterjee, Prof. Naeemi, Prof. Schimmel, and Prof. Zhou for serving as the dissertation committee members and the insightful comments to improve my research dissertation.

Finally, I would like to thanks to my colleagues, Dae-Hyun Kim, Soonyong Cha, Chang-Chih Chen, Tazhi Lu, and Kexin Yang in our lab for the co-work and a lot of help to construct my research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

η	Weibull characteristic lifetime
BIST	Built-In Self Test
BTDDB	Backend Time-Dependent Dielectric Breakdown
EM	Electromigration
SIV	Stress-Induced Voiding
GTDDB	Gate Oxide Time-Dependent Dielectric Breakdown
NBTI	Negative Bias Temperature Instability
PBTI	Positive Bias Temperature Instability
HCI	Hot Carrier Injection
TPG	Test Pattern Generator
WE	Write driver enable signals
W-data	Data inputs
SAE	Sense amplifier enable signals
PRE	Precharge circuit enable signals
V_pre	Precharge voltages
P_down	Pull-down control signal
ORA	Output Response Analyzer
SC	Sensing Circuit
I/Os	Inputs and Outputs
SG	Short Group
OG	Open Group
ECC	Error Correcting Codes

SRAM	Static Random Access Memory
DRAM	Dynamic Random Access Memory
STT-MRAM	Spin-Transfer Torque Magnetic Random Access Memory
ReRAM	Resistive Random-Access Memory
3D IC	Three-Dimensional Integrated Circuit
TSV	Through-Silicon Via
CTE	Coefficient of Thermal Expansion

SUMMARY

The main objective of this dissertation is to propose comprehensive methodologies, including design, test, and statistical failure analysis, to handle reliability issues in an embedded cache, processors, and main memory systems. We propose design and test methodologies for the diagnosis of wearout mechanisms in an embedded cache in a processor. The diagnosis results from our proposed methodology are utilized to monitor the system health of the processor. We also propose optimized design solutions for the implementation of an emerging main memory system.

First, we present the detection and diagnosis methodologies for various wearout mechanisms, including backend time-dependent dielectric breakdown (BTDDB), electromigration (EM), stress-induced voiding (SIV), gate oxide time-dependent dielectric breakdown (GTDDB), and bias temperature instability (BTI) in an SRAM array. The built-in self-test (BIST) system and algorithm detect wearout and identify the locations of the faulty cells. Next, the physical location of the failure site within SRAM cells is determined. There are some fault sites for different wearout mechanisms which result in exactly the same electrical failure signature. For these faulty sites, the cause of failure probabilities for each wearout mechanism can be determined by matching the observed failure rate from the BIST system and the failure rate distribution computed by mathematical models as a function of circuit use scenarios. The estimation of wearout distributions in embedded caches is useful in determining the wearout limiting mechanisms in the field and repair schemes.

We also propose to use the embedded SRAM as a monitor of system health. The bit failures are tracked with error correcting code (ECC) and the cause of each bit failure

is diagnosed with on chip built-in self test (BIST) and statistical failure analysis. The wearout model parameters are extracted from the diagnosis results and combined with system wearout simulation to estimate the remaining lifetime of the entire processor dynamically.

For the main memory system, we have studied design methodologies for an emerging main memory to overcome the limitations of device scaling. Among many candidates for emerging memory systems, we have focused on 3D DRAM, where multiple DRAM dies are vertically stacked and connected with through-silicon-vias (TSVs), to increase the total memory capacity. Especially, we present a design solution for 3D DRAM to optimize reliability, power, cost, and performance, given emerging reliability issues induced by TSVs.

CHAPTER 1

INTRODUCTION

Reliability is becoming more critical because advanced process technology scaling has involved the reduction of interconnect and transistor dimensions without reducing the supply voltage in proportion. Hence, wearout of devices and interconnects is occurring more quickly with aggressive technology scaling. Despite the use of more vulnerable components, SRAM systems in electronic applications, from mobile devices, personal computers, automatic vehicles, to flight controllers, need to be fault tolerant and reliable in order to guarantee safe operations. Among several techniques to ensure fault tolerance is the use of error correcting codes and redundant arrays, together with on-chip test algorithms for automated self-reconfiguration of SRAMs [1].

Despite the use of error correcting codes and memory redundancy, systems can fail in the field. This happens if the system does not have sufficient redundant resources or if the wearout rate is faster than predicted. Under such circumstances, failing chips are returned to the manufacturer, and the manufacturer is expected to diagnose the cause of wearout failures. The standard method is physical failure analysis, which involves deprocessing to visually determine the nature of the defects and failures. The success rate for physical failure analysis is low and the required cost to perform physical failure analysis is too high. Hence, there is a need to develop another method to determine the causes of wearout. In this work, we propose built-in electrical tests with statistical analysis of volume test data based on mathematical models to determine the causes of wearout.

According to the International Technology Roadmap for Semiconductors (ITRS), high performance processors, such as servers, are expected to consist of 82% memory on average. Since SRAMs are designed with the tightest design rules, they can provide an appropriate vehicle to diagnose most wearout failures in a processor. Moreover, since SRAMs use error correcting codes, an SRAM will have many failing cells whose causes of failure can be determined. The use of electrical tests with statistical failure analysis enables efficient diagnosis of the causes of failure of large failing samples, which in turn increases confidence in the results of failure analysis.

To monitor the health of an SRAM array, an SRAM system may be monitored periodically, and the field test data can be combined to determine the separate wearout distributions for each wearout mechanism. Then, we can identify wearout model parameters for each wearout mechanism. These separate wearout models can then be compared with process-level models to determine if lifetime is correctly estimated, and if not, appropriate corrections can be made to improve the manufacturing process.

Firstly, in this thesis, we propose diagnosis methodologies for all possible frontend and backend wearout mechanisms in an SRAM array, namely backend time-dependent dielectric breakdown (BTDDB) and gate oxide time-dependent dielectric breakdown (GTDDB), which result in resistive-bridges in an SRAM array, via/contact voiding due to current stress-dependent electromigration (EM) and temperature-stress-dependent stress-induced voiding (SIV), and threshold voltage shifts due to NBTI and PBTI. Unlike the resistive-open and bridging models presented in [2],[3], the fault model in our thesis includes resistive-bridging defects and resistive-open defects in vias/contacts, considering only the BTDDB, GTDDB, EM, and SIV effects that are

feasible based on a physical layout of an SRAM cell. Moreover, even if it is expected that most failures are due to a smaller set of frontend wearout mechanisms, namely BTI and GTDDB, we have included a much larger set of wearout mechanisms for completeness.

Note that the EM and SIV mechanisms result in exactly the same failure signatures (opens), as do BTDDB and GTDDB (shorts). It is not easy to separate them using electrical tests only. Nevertheless it is important to separately determine the failure rate for each mechanism to estimate the lifetime of the entire chip correctly and to help improve the manufacturing process. Hence, overall, our electrical test methodology not only involves determining if the failure in an SRAM cell is a short or an open, but also identifies the physical location of each voiding via/contact and short site. To determine the cause of faults with the BIST test data, we propose to match the failure rate from BIST using volume data and the failure distribution from a reliability simulator [4]-[10]. We conduct statistical analysis to distinguish GTDDB vs. BTDDB failures and the EM vs. SIV mechanisms to determine separate wearout distributions. For statistical analysis, we also present numerical optimization methodologies that use more test sets with more stress acceleration conditions to make our statistical methodology tolerant to errors from process variations and the statistical analysis.

The extracted wearout distribution from the diagnosis results can also be used to monitor the remaining lifetime of the entire processor dynamically. High-performance processors, such as high-end server processors, are usually designed with tight design constraints and operate with a fast clock frequency. For such high-end systems, the dynamic monitoring of wearout is important to guarantee safe operations [11]-[13].

To do so, components can be monitored periodically with the proposed BIST and statistical failure analysis to detect components that are likely to fail in the near future. Then, by monitoring the remaining life, the components which have a risk of potential failures can be replaced prior to failure.

The embedded memory systems and logic blocks are likely to fail at different rates. However, the cache systems are potentially less vulnerable to wearout mechanisms since they can be reconfigured on-line [15] and use error correcting codes (ECCs) [16]. Fig. 1.1 shows the failure distributions for both logic blocks and memory blocks in the open-source LEON3 processor [14]. The memory blocks cover 89% of the layout area, but are much less vulnerable to failure.

The proposed research to estimate the remaining lifetime involves two steps, a backward parameter extraction process, and a forward lifetime distribution prediction process. The backward parameter extraction process involves measurement data from SRAM systems. Specifically, the wearout model parameters are extracted from observed memory bit failures in the field, after the chip has been in operation. We use built-in self test with electrical failure analysis to diagnose and classify the failures and track the failure rate of memory cells for each mechanism. Process-level Weibull parameters for all critical wearout mechanisms are estimated using conversion maps between SRAM cell Weibull parameters (describing the observed failure rate) and process-level Weibull parameters. The conversion maps are generated with lifetime simulation based on the aging simulation framework presented in [4]-[10].

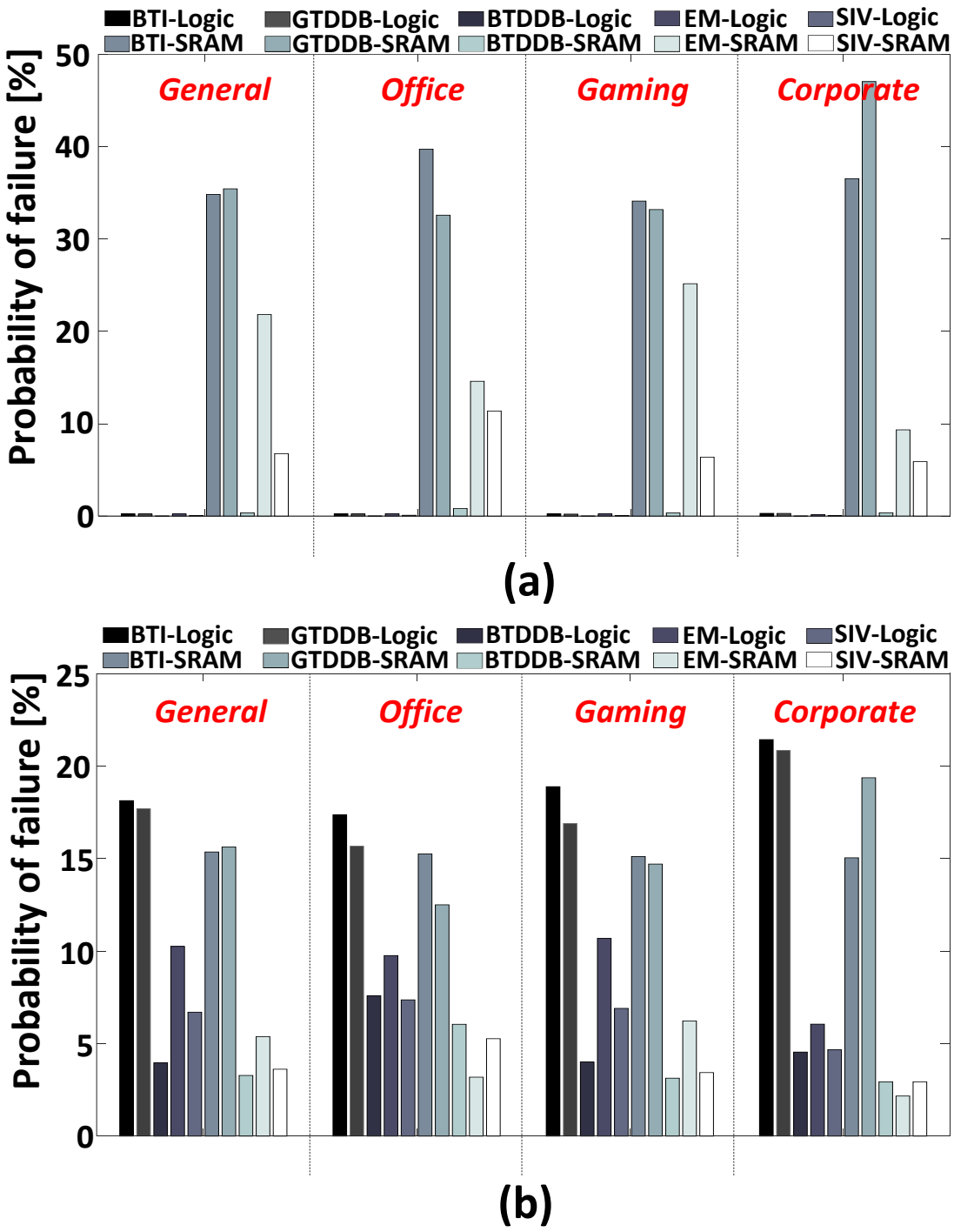

Figure. 1.1 The failure rates of the logic parts and memory parts (226Kb) of the LEON3 processor [14] due to BTI, GTDDB, BTDDB, EM, and SIV for four usage scenarios (a) without ECCs and (b) with ECCs. The logic components consist of the IU, MUL, DIV, and MMU. The memory systems contain the D-Cache, I-Cache, D-tags, I-tabs, and RF.

The forward lifetime distribution process is also conducted with the aging simulation framework presented in [4]-[10], which involves simulating microprocessors with standard benchmarks [17] on an FPGA to extract the activity and temperature profiles. Since the lifetime depends on workload, different use scenarios labeled as corporate, gaming, office work, and general usage are utilized for our research [18]. These use scenarios presented in Fig. 1.2 represent fractions of time in operation, standby, and off states. We combine simulation data from the forward simulation process with the extracted process-level parameters to estimate the remaining life of the entire processor which is a function of the memory bit failures, which are tracked with ECC failures.

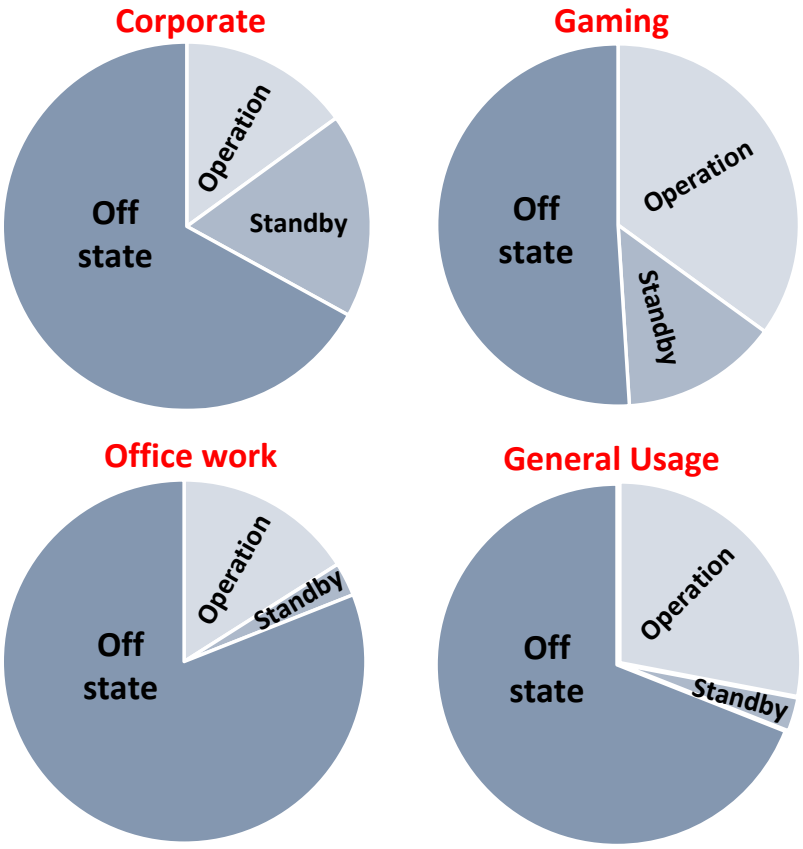

Figure. 1.2 Use scenarios provided by Intel [18].

The main memory system is considered as one of the critical components of computing systems, such as in servers, embedded, desktops, and mobile [21]. It is important to scale memory capacity, power, cost, and performance as we scale the size of the computing system [21]. However, scaling is difficult [21].

Hence, many emerging memory systems, such as STT-MRAM [22] and ReRAM [23], have been proposed. Among the proposed candidates, 3D DRAM is believed by many to be capable of becoming a commercial product in the mainstream market. The total memory capacity in a single DRAM chip increases linearly with the number of tiers stacked with the same footprint. In addition, the recently announced wide-I/O standards increase the memory bandwidth for communication with CPUs, GPUs, and application processors stacked together [24]. These benefits enable 3D DRAMs to be a promising solution in both the mobile and computing areas as they promise massively parallel computing at low power consumption [25],[26].

When a DRAM system is to be implemented using 3D stacking technology, designers should first decide how to partition the system and memory architectures into individual dies. For the two notable designs proposed, each die in a stack has all of the basic components, including DRAM cell arrays, decoders, multiplexers, sense-amps, and peripheral circuits [19]. In this so-called cell/logic-mixed 3D DRAM design, DRAM cell arrays are mixed with logic so that all dies have identical designs, except for the bottom die that contains additional components to handle the interface with packages as presented in Fig. 1.3(a). The pros of cell-mixed design include easy design and a smaller TSV count. The cons include a mechanical reliability issue due to thermal mechanical

stress induced by TSVs and a larger chip size mainly due to the presence of cells, logic components, and I/O pads/circuits in each die.

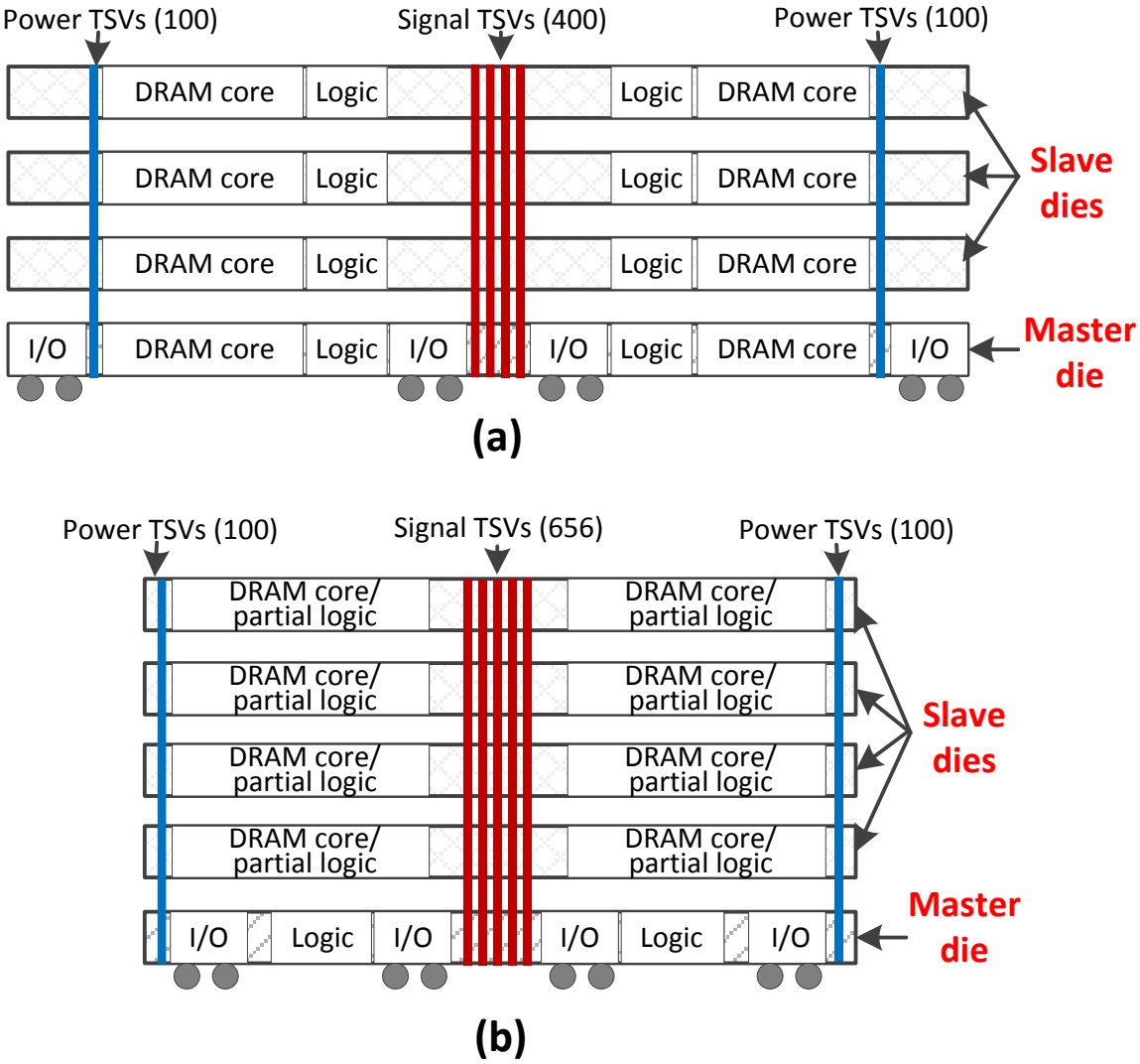

Figure. 1.3 Vertical drawing of (a) 4-tier cell/logic-mixed design [19], (b) our 5-tier cell/logic-split design [20].

In this thesis, we propose another 3D DRAM design style called cell/logic-split to provide design guidelines for a 3D DRAM system [20]. In our 5-tier design strategy, each of the 4 slave dies contain DRAM arrays, decoders, sense amps, and some parts of the control logic, while the master die contains I/O pads/circuits, buffers, and most of the

peripheral circuits. We also develop two design schemes to minimize TSV usage in our design. Our simulations show that the maximum mechanical stress induced in our DRAM design style is reduced by 49.1%. Also, this proposed design leads to a total power consumption reduction by 23.6% for write operations and 27.3% for read operations. There are also performance benefits, i.e. tRCD write **(**row address to column address delay) reduction by 1.9ns (15.6%).

This dissertation is organized as follows. Chapter 2 presents a background of the related work and prior research. In Chapter 3, the faults considered and their models in an SRAM cell are presented. Chapter 4 presents BIST and statistical analysis methodologies for diagnosis of wearout mechanisms. Chapter 5 shows the estimation of the remaining life of a processor based on separate wearout distributions from Chapter 4. In Chapter 6, we presents a comparative study of reliability, power, performance, and yield analysis of 3D SDRAM designs built with two practical die partitioning styles, namely, cell/logic-mixed and cell/logic-split. Chapter 7 concludes this dissertation.

CHAPTER 2

BACKGROUND

Reliability of VLSI systems, such as CPUs, GPUs, high computing processors, and application processors, is regarded as the one of barriers for process technology scaling. Aggressive process technology scaling accelerates wearout of devices and interconnects, especially with nanoscale technologies. The frontend wearout mechanisms consist of gate oxide time-dependent dielectric breakdown (GTDDB), bias temperature instability (BTI), and hot carrier injection (HCI) and backend wearout is induced by backend time-dependent dielectric breakdown (BTDDB), stress induced voiding (SIV), and electromigration (EM).

Failures due to the SIV mechanism have been researched in [27]-[29]. Directionally biased motion of atoms is induced by thermal mechanical stress between metals and dielectric materials. The biased motion of atoms can create voids inside of vias and can increase the via resistance. This failure mechanism is called stress induced voiding, and it leads to timing and functional failures in digital systems.

Electromigration (EM) can result in exactly the same electrical failure signature in a chip. The EM mechanism leads to the transfer of momentum from electrical current to ions in the metallic lattice. The metallic ions are transported into the neighboring material due to the transfer of momentum from EM, leading to a reduction of via dimensions and an increase in resistance [30]-[35].

Time-dependent dielectric breakdown consists of gate oxide time dependent breakdown (GTDDB) [36]-[38] and backend time dependent breakdown (BTDDB) [39]-[40]. These mechanisms lead to the same electrical faults, namely a resistive bridging

fault. GTDDB is the frontend mechanism which is induced by trap-assisted tunneling mechanisms or oxide breakdown in CMOS devices. BTDDB is one of the backend wearout mechanisms and is caused by dielectric breakdown between unconnected metal layers.

Bias temperature instability (BTI) and hot carrier injection (HCI) can cause the threshold voltage to shift [41]-[43]. The traps at the gate oxide interface and in the oxide lead to the BTI mechanism. BTI is induced when the CMOS devices are under constant stress. Negative bias temperature instability (NBTI) causes increases in the threshold voltage of PMOS devices and positive bias temperature instability (PBTI) causes the increase in the threshold voltages of NMOS devices. The HCI mechanism also shifts the threshold voltages of the CMOS devices when the devices are operated with high switching activity, since the HCI mechanism depends on the time under dynamic stress.

For aging analysis, first we model the time-dependent wearout mechanisms with the Weibull distribution as

$$P(t) = 1 - exp^{-(t/\eta)^{\beta}} \tag{2.1}$$

where η is the characteristic lifetime, β is the shape parameter which describes the dispersion of the failure rate population, t is time, and P is the probability of failure [44].

Equation (2.1) is reformatted to extract Weibull parameters from data as follows:

$$-ln\big(1 - P(t)\big) = (t/\eta)^{\beta} \tag{2.2}$$

$$ln\Big(-ln\big(1 - P(t)\big)\Big) = \beta ln(t) - \beta ln(\eta). \tag{2.3}$$

The characteristic lifetime, η, is the time when the probability, $P(t) = 63\% = 1 - \exp(-1)$ has failed.

The methodologies to detect the frontend wearout mechanisms in an SRAM array have been studied in [45]-[47]. In these papers, current test methodologies have been presented to detect the GTDDB and NBTI mechanisms in an SRAM cell. However, although GTDDB and BTI are expected to be the dominant failures in an SRAM, the backend wearout mechanisms, such as BTDDB, SIV, and EM, can also be induced in an SRAM cell, especially with advanced technology nodes. Moreover, all wearout mechanisms can be confounded in a single SRAM cell.

To improve a manufacturing process and guarantee system reliability, separate wearout distributions for each mechanism are required to check whether lifetime is correctly estimated. Hence, there is a need to develop new diagnosis methodologies to detect and distinguish all possible wearout mechanisms when they are confounded in a single SRAM array at the same time.

To identify the cause of a fault to reduce the cost of physical failure analysis, diagnosis techniques are presented in prior research [48]-[50]. These studies have mainly focused on diagnosis methodologies to identify the physical layer which contains the resistive short fault. They have proposed algorithms to identify the cause of failures using the inclusion of color bitmaps and/or current test techniques. Unlike these prior research techniques on test, our proposed research presents a diagnosis methodology for wearout mechanisms in an SRAM cell.

The prior research on wearout test [48]-[50] has focused on the cell-level test techniques to detect GTDDB or BTI mechanisms in a single cell. Critical manufacturing and test issues, especially test time and cost, are not considered in the previous studies.

Hence, a system-level test methodology and algorithms for the entire memory bank and cache clusters should be investigated to minimize test cost and to enhance test coverage.

The current monitoring in the prior studies is sensitive to the capacitance and resistance of the bitline pair. Hence, when the test technique is used for a larger memory array, additional test techniques should be proposed to avoid errors in the current tests. Also, if an SRAM is designed with highly scaled technologies, the off-state leakage current cannot be ignored [51]. The leakage current can lead the current test methodology to be less effective. When we move to more advanced technologies, the leakage current should be carefully controlled with system-level test and design techniques.

To minimize the test cost and error for the diagnosis of all possible wearout mechanisms in an SRAM array, our study has focused on system-level BIST system and algorithms. The prior research in [52]-[59] has proposed system-level BIST, built-in repair analysis (BIRA), and built-in self repair (BISR) to enable automated test and repair of SRAMs. The test and repair systems presented in [52]-[59] detect defects and repair the memory systems with redundant arrays during the manufacturing process. However, the test and repair methodologies are less effective for wearout mechanisms, since wearout mechanisms are mostly induced after shipping the chip from the manufacturer. A fundamental solution to avoid wearout mechanisms is to improve the manufacturing process and device models to avoid the use of repair with redundant arrays. To improve the manufacturing process, the diagnosis of wearout mechanisms should extract separate wearout model parameters. The proposed research in this thesis is not just to detect wearout mechanisms and reconfigure the array for repair with the redundant array, but

also to diagnose the cause of wearout through failure analysis with electrical signals and statistical analysis methodologies.

A processor contains different types of cache clusters. The caches are designed with SRAM systems and are classified in hierarchies between one and three levels [60]. The first-level cache is usually designed with SRAM arrays containing several tens of kilobytes of cells, and upper level caches (L2 and L3) consist of between several hundred kilobytes and a few megabytes of cells [60],[61]. The first level cache should be synchronized with the fast clock since it can be accessed with a latency of one to four clock cycles. The operating speed for second and third level caches is slower since a latency of several tens of clock cycles is allowed.

The customized BIST system and algorithm for wearout mechanisms should be reconfigurable for various memory architectures, with different operating speeds and array sizes. Design of different BIST systems for different memory specifications increases the design cost significantly. Hence, there is a need to develop the reconfigurable platform and in-house tool flow to generate the BIST system and joint test action group (JTAG) test bench for different memory systems. The prior studies in [62] present the usage of commercial tool flows for BIST design for their specific purposes. Based on the prior study, we have developed a reconfigurable platform to create the BIST system and JTAG test bench for the diagnosis of wearout mechanisms in the memory array.

The diagnosis results from SRAM BIST and statistical analysis can be used to estimate the remaining lifetime of the processor after shipping the chip from the manufacturer. Methodologies to estimate the remaining lifetime of a semiconductor

device have been proposed in [63]. Using embedded sensors, such as temperature sensors, current sensors, and voltage sensors, they have estimated the usage of the device based on operating parameters, which include the actual temperature, voltage, and operating frequency. Then, the remaining lifetime for the system is estimated based on the usage of the device calculated with the operating parameters. However, the design of additional sensors and controller blocks can lead to an area overhead and additional design cost. Also, the same sensors and systems may not be easily utilized for different applications because the operating parameters depend on the process technology. Hence, we aim to propose test methodologies to monitor the remaining lifetime of the entire processor based on our BIST techniques and statistical failure analysis that do not need major re-design for each application.

For the main memory system, the limitation to the device scaling has been considered as the one of the difficult challenges to move to the next DRAM generation. DRAM technology scaling can lead to many benefits due to its capacity, power, cost, and reliability [21]. Although many alternative memory solutions, such as STT-MRAM and ReRAM, have been proposed, TSV technology is regarded as one of the feasible solutions to lead to mass production of the emerging technology due to less challenges related to technology transfer, cost, and yield issues.

Although TSV stacking is the key enabling technology for 3D memories, the TSV can involve disruptive manufacturing issues compared with conventional 2D ICs [64]. TSVs cause significant thermo-mechanical stress that can induce performance, reliability, and yield degradation (see Fig. 2.1) [64]. Also, since it is not easy to reduce the TSV size due to manufacturing issues, the area and cost overhead issues can be another bottleneck

to enable bringing 3D DRAM technology to the market. Hence, there is a need to develop an optimized design solution to resolve the complex tradeoff between power, reliability, cost, and performance.

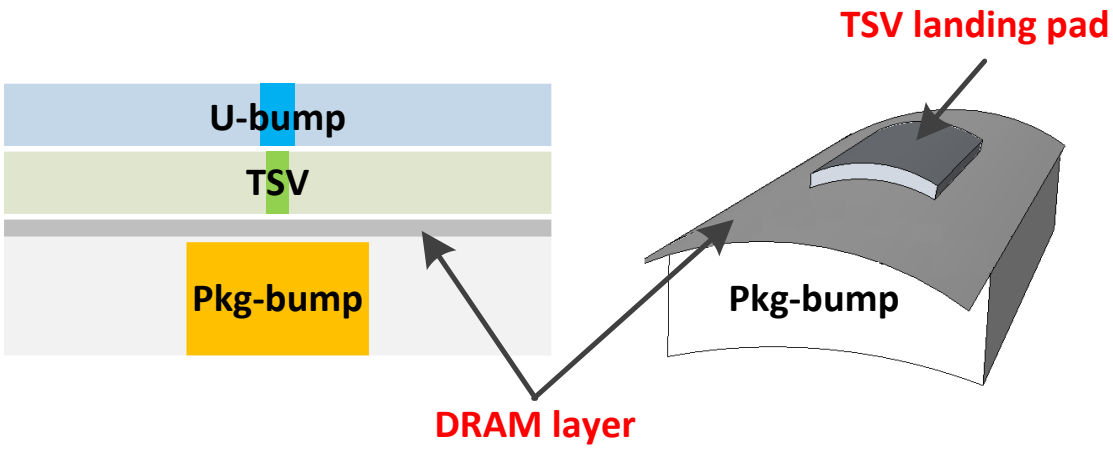

Figure. 2.1 Impact of bumps and underfill on the stress of device layer [64].

CHAPTER 3

WEAROUT MODELING IN AN SRAM CELL

There are many SRAM layout options which are designed to be appropriate for different purposes [65]. Among the SRAM layout options, we have used a physical layout which has many possible wearout sites. When the layout changes, the sites of frontend mechanisms, which are GTDDB and BTI, do not change, but sites for the backend wearout mechanisms can be changed. In this case, BIST patterns can be slightly revised to account for the different backend sites. When there are undetectable backend fault sites with the revised BIST patterns, the failure rates for the backend faults can be controlled by varying the design rule (DRC) margins for metal widths and lengths and spaces between adjacent metals in the physical layout. Also, since the frontend mechanisms are generally the dominant failure mechanisms in the SRAM array, the overall failure rate of the entire SRAM is not significantly impacted by several undetectable backend wearout mechanisms.

3.1 Modeling GTDDB and BTDDB Mechanisms

Gate oxide time-dependent dielectric breakdown (GTDDB) is modeled as a leakage path through the gate oxide of transistors in an SRAM cell [45]. Although the leakage path can be also induced between the gate and substrate, the gate-to-substrate leakage is neglected because it has little effect on the performance of the memory [45]. With this assumption, we model only the dominant paths which are the gate-to-source and gate-to-drain leakage paths.

Only GTDDB in the four transistors in the two inverters in a cell is considered in this work because stress for access transistors is almost negligible especially when the

system frequency is high. Fig. 3.1(a) presents that the lifetimes for the access transistors (M5, M6) due to GTDDB are much larger than those for other transistors in a cell. The leakage paths induced by GTDDB are modeled in an SRAM cell in the sites in Fig. 3.2 (G1-G8).

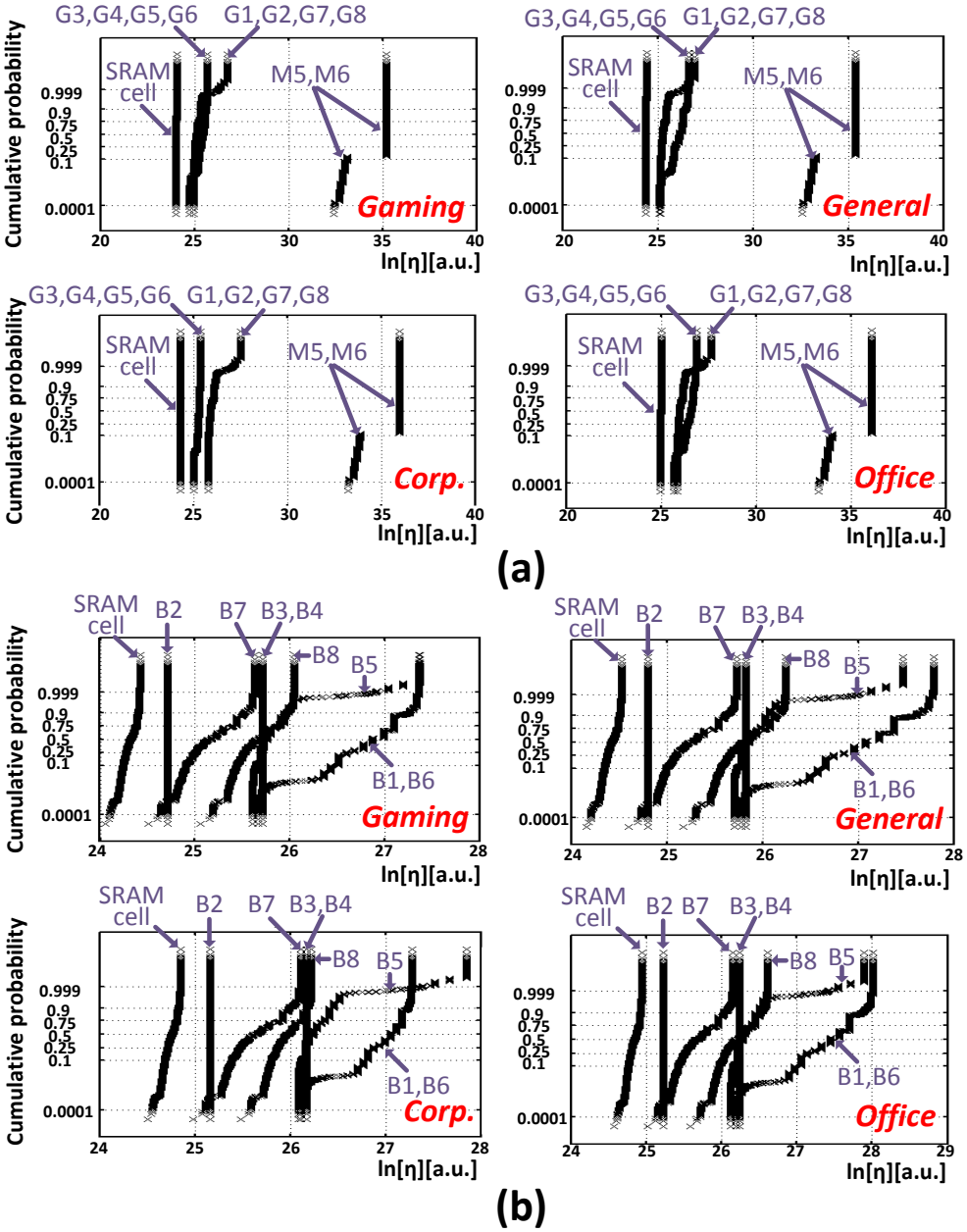

Figure 3.1 Cumulative probability distribution of characteristic lifetime for access and cell transistors for 32Kbit SRAM array with different use scenarios: (a) GTDDB, and (b) BTDDB. The overall result for all GTDDB and BTDDB faults for a cell is named as "SRAM cell" in (a) and (b), respectively.

The high electric fields with the advanced process technologies also lead to backend dielectric breakdown, which also induces leakage paths in an SRAM cell. Fig. 3.2 presents the sites of BTDDB in a physical layout of a cell. Six possible leakage paths due to dielectric breakdown are induced in a cell and two more BTDDB leakage paths exist between two adjacent SRAM cells. Fig. 3.2 presents the locations of the leakage paths induced by BTDDB in a schematic of an SRAM cell. Many leakage paths in a cell due to GTDDB and BTDDB are the same and electrical signatures from the two mechanisms cannot be distinguished using only electrical tests. Hence, we group the leakage paths due to GTDDB and BTDDB into four groups (SG1-SG4) presented in Table 3.1. The index k is used to denote the short group (SG1-4), and the index i is the index for the cell number. j is an index to indicate the short location for B1-B6 and G1-G8 within the short groups for each mechanism.

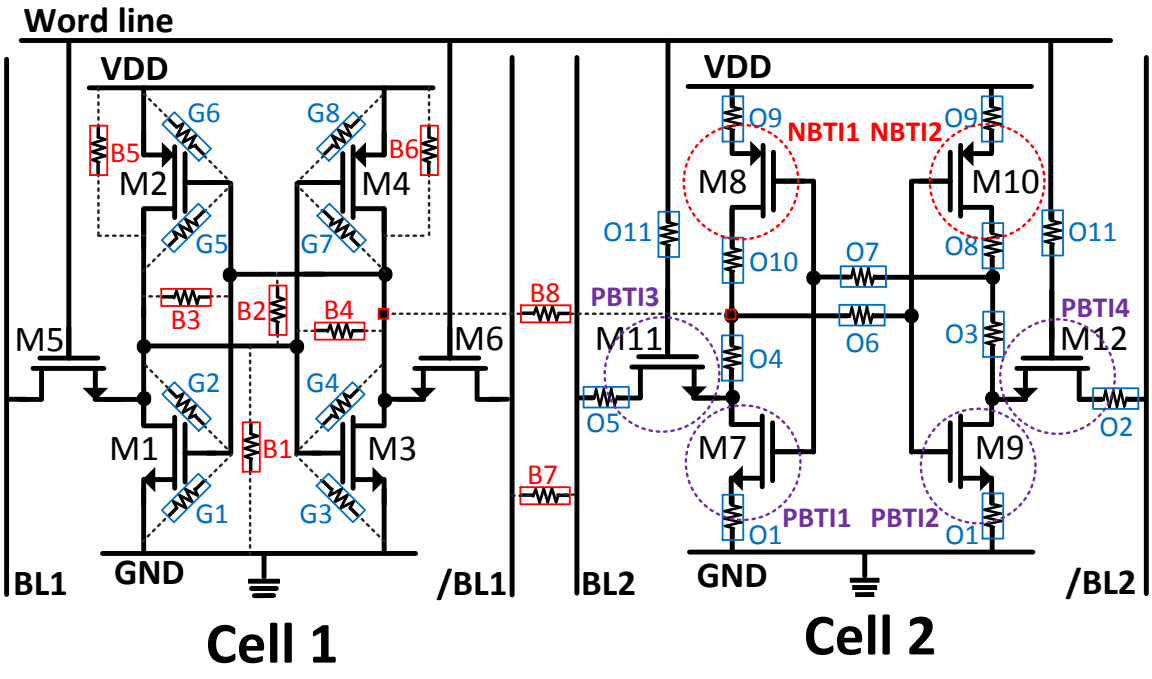

Figure 3.2 Modeling of wearouts for BTDDB (B1-B8), GTDDB (G1-G8), via/contact voiding (O1-O11), NBTI (NBTI1,NBTI2), and PBTI (PBTI1-PBTI4).

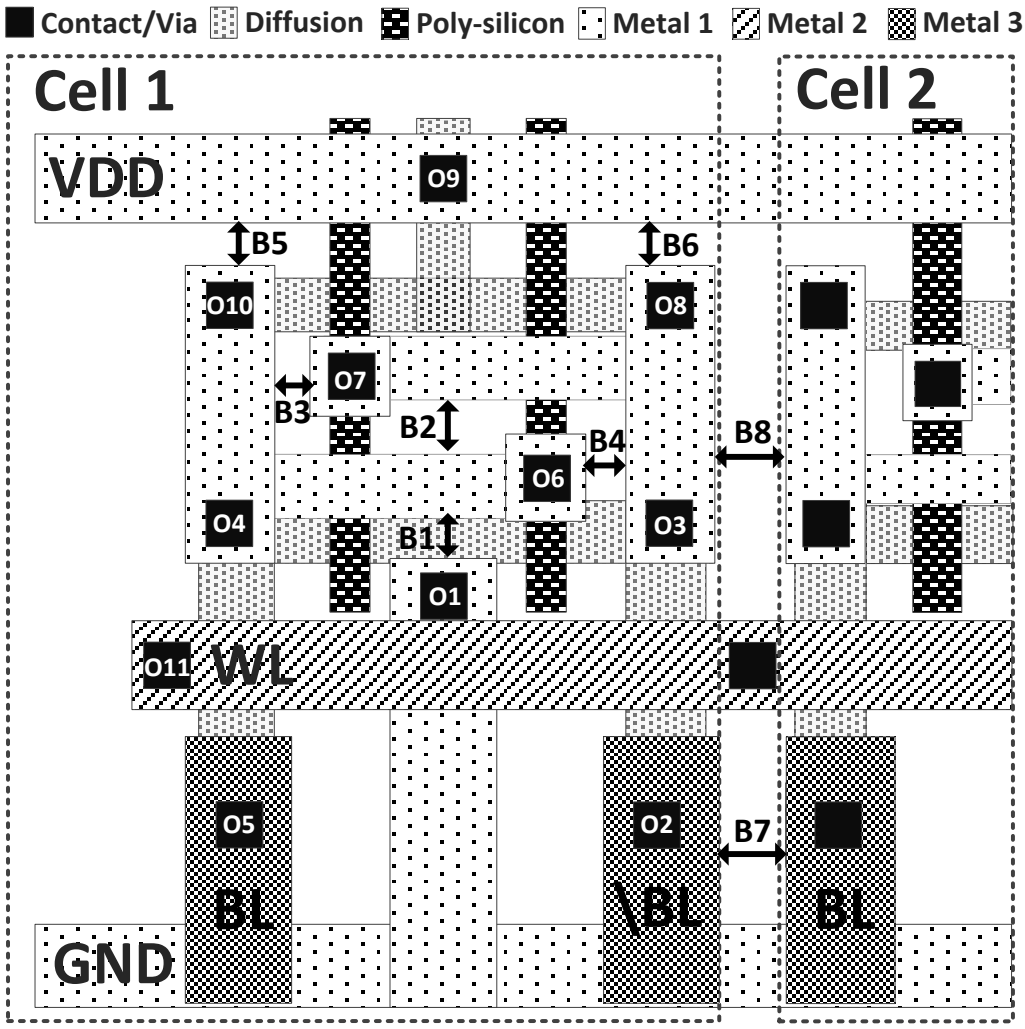

Figure 3.3 Backend wearout locations in a physical layout of an SRAM cell due to BTDDB (B1–B8) and via/contact voiding because of EM and SIV (O1–O11).

TABLE 3.1. GROUPS AND INDICES FOR RESISTIVE SHORT FAULTS

Group	GTDDB	BTDDB
SG 1 (k=1)	G6 (j=1)	B6 (j=1)
SG 2 (k=2)	G8 (j=1)	B5 (j=1)
SG 3 (k=3)	G3 (j=1)	B1 (j=1)
SG 4 (k=4)	G2 (j=1),G4 (j=2), G5 (j=3) ,G7 (j=4)	B2 (j=1), B3 (j=2), B4 (j=3)

We model GTDDB and BTDDB mechanisms with Weibull distributions with two parameters, a characteristic lifetime (η) and a shape parameter (β). The characteristic lifetime, η_{GTDDB}, for GTDDB is as follows [5],[7]:

$$\eta_{GTDDB} = A_{ox}\left(\frac{1}{WL}\right)^{\frac{1}{\beta_{OX}}} exp\left(\frac{-1}{\beta_{OX}}\right) V^{a+bT} exp\left(\frac{c}{T}+\frac{d}{T^2}\right)/s \quad (3.1)$$

where W and L are the width and length of device, respectively, β_{OX} is the Weibull shape parameter, s is the fraction of time that the gate is under stress, T is temperature, V is the gate voltage, and a, b, c, d, and A_{ox} are fitting parameters for the wearout model. The characteristic lifetime for GTDDB is a function of the location of the failure site because all failure sites do not experience the same stress which depends on workload.

The characteristic lifetime for the BTDDB mechanism is [4]-[7]:

$$\eta_{BTDDB} = A_{BTDDB} L_i^{\frac{-1}{\beta_{BTDDB}}} exp\left(-\gamma E^M - E_a/k_B T\right)/\alpha'. \quad (3.2)$$

The characteristic lifetime is a function of the vulnerable length, L_i, its associated line space, S, the corresponding electric field, E=V/S, where V is the supply voltage, the Weibull shape parameter, β_{BTDDB}, the field acceleration factor, γ, the activation energy, E_a , Boltzmann's constant, k_B, the probability that the adjacent nets to the dielectric segment are at opposite voltages, α',and fitting parameters, A_{BTDDB} and M [4]-[7].

Fig. 3.1 presents that the cumulative probability distributions of the characteristic lifetimes of the resistive short sites due to GTDDB are not the same as those due to BTDDB, even if these faults result in exactly the same electrical failure signature (same resistive short site). To apply our diagnosis methodology to various applications, the relative failure rates of specific sites are utilized to diagnose the failure rate for GTDDB and BTDDB for the SRAM array. When a different process technology is used, the characteristic lifetime values in Fig. 3.1 can change. However, our statistical analysis method is still valid because it involves the relative failure rate of specific sites for each mechanism.

3.2 Modeling Via and Contact Voiding by EM and SIV Mechanisms

Current transfers momentum to ions in the metallic lattice, leading some of the metallic ions to be transferred to the adjacent material. This causes the electromigration (EM) effect, leading to the reduction of via/contact dimensions and an increase in resistance [4],[6]-[7]. The characteristic lifetime of a via/contact due to EM, η_{EM}, is modeled as

$$\eta_{EM} = A_{EM}\, T / j_{EM} \tag{3.3}$$

where T is operating temperature, j_{EM} is the current density, and A_{EM} is a technology dependent constant [4],[6]-[7]. The rate of increase in via or contact resistance is a function of the average current density which flows through a via/contact [4],[6]-[7].

With highly scaled process technologies, vias/contacts connected to shorter metal wires do not suffer from voids since the gradual movement of conductor atoms can create a back-stress to reduce the effective material flow caused by EM [30]-[35]. The minimum wire length, called the Blech length, and a current density product that causes via voiding are defined to address the EM effect. In an SRAM cell, via/contacts connected to bitline pairs and the VDD path can experience a risk for via/contact voiding due to EM mechanism [31]. Other via/contacts in the cell do not meet the critical requirements for the Blech length or the high unidirectional current density to form via voids. Hence, we can assume that only O2, O5, and O9 in Fig. 3.3 have a risk of void formation due to the EM mechanism. Although the EM mechanism is more likely with a larger memory array (level 2 or level 3 caches) which provides a longer Blech length for vias/contacts connected to VDD and bitline pairs, we include EM models in our work to make the diagnosis methodology more general for various types of memory applications.

Thermal mechanical stress between the metal and the dielectric causes directionally biased motion of atoms at high temperatures. This induces stress-induced voiding (SIV), leading to an increases in via/contact resistance and eventually voiding inside of a via [27]-[29]. The resistance of a via/contact is the function of the difference between the operating temperature and the stress-free temperature of the material. The characteristic lifetime, η_{SIV}, due to the SIV mechanism can be modeled as

$$\eta_{SIV} = A_{SIV} W_{SIV}^{-M} W^{-M} (T_0 - T)^{-N} exp(E_a/kT) \quad (3.4)$$

which depends on the linewidth, W_{SIV}, the geometry stress component, M, the stress-free temperature, T_0, the thermal stress component, N, the activation energy, E_a, and a constant, A_{SIV} [6]. Unlike the resistive-open fault model presented in prior work [3], there are 11 possible worn-out via/contact locations (O1-O11) due to SIV in Fig. 3.2 and Fig. 3.3.

Note that the stress experienced by each via/contact depends on the average current density, temperature, and the geometry components (see equation (3.3)). When stress varies significantly for each via/contact, the lifetimes of each via/contact within a cell due to EM are different. Fig. 3.4(a) shows the cumulative characteristic lifetime distribution due to EM mechanism for the 32Kb cells for different use scenarios. It can be seen that the lifetimes are different for some via/contact locations even in the same cell. Also, the characteristic lifetimes of each via/contact due to SIV is function of on the linewidth of metal above the via/contact and the stress component (see equation (3.4)). Since they are not the same for all via/contacts in an SRAM cell, the lifetimes due to the SIV mechanism are also not the same (see Fig. 3.4(b)).

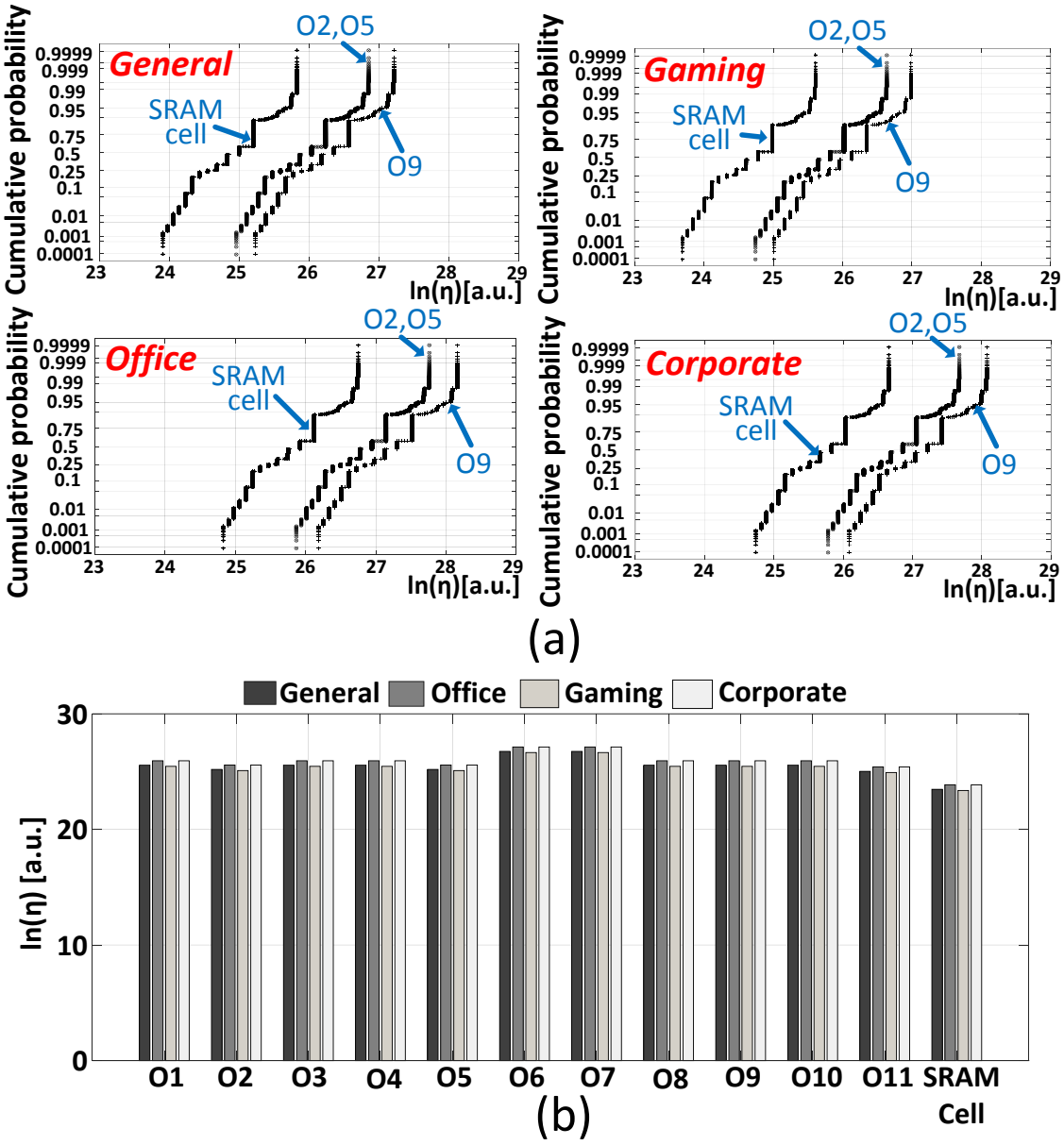

Figure 3.4 The characteristic lifetimes of vias/contacts due to EM and SIV for 32Kb cells for different use scenarios: (a) the cumulative probability distribution of lifetime for vias/contacts due to EM mechanism, and (b) average lifetime for vias/contacts in a cell due to SIV mechanism.

The resistive open defects for O2, O5, and O9 due to the EM and SIV mechanisms can lead to the same electrical failure signatures in an SRAM array. Hence, statistical failure analysis is also conducted to diagnose the probability distributions of the

causes of failure using the relative failure rates at each site for each mechanism in Fig. 3.4. The three possible open groups due to EM and SIV are summarized in Table 3.2.

TABLE 3.2 FAULT GROUPS AND INDICES FOR RESISTIVE OPEN FAULTS DUE TO EM AND SIV

Group	EM	SIV
OG 1 (m=1)	O2_EM	O2_SIV
OG 2 (m=2)	O5_EM	O5_SIV
OG 3 (m=3)	O9_EM	O9_SIV

3.3 Modeling NBTI, PBTI, and HCI

The presence of traps at the gate oxide interface and in the oxide induces the NBTI mechanism. NBTI can lead to an increase in the threshold voltage of PMOS devices when the devices are under stress [46]. When an SRAM cell holds a fixed state for a long time during standby, the cell performances become skewed, with one PMOS in the cell being largely unaffected, while the other degrades [46]. When PMOS device (M8) in Fig. 3.2 suffers from NBTI degradation (V_{tp} threshold voltage shift), we define this NBTI model as NBTI 1. When the other PMOS device (M10) in the same cell in Fig. 3.2 suffers from NBTI, we call the NBTI model NBTI 2.

The PBTI mechanism impacts V_{tn} of the four NMOS devices in an SRAM cell. Although the PBTI mechanism is unlikely with our 90nm technology, we have included PBTI models to make our methodology more general and useful for future technology generations. Fig. 3.2 shows definitions of PBTI 1, PBTI 2, PBTI 3, and PBTI 4 in a cell.

HCI also induces the threshold voltages of devices to shift. However, if the switching activity is relatively low, as is typical, SRAM cells are much more prone to BTI degradation, which is a function of constant stress, rather than HCI which depends on the time under dynamic stress [46]. If our methodology diagnoses BTI degradation in an SRAM array, then it can also diagnose the threshold voltage shifts due to HCI.

CHAPTER 4

BUILT IN SELF TEST METHODOLOGY WITH STATISTICAL ANALYSIS FOR ELECTRICAL DIAGNOSIS OF WEAROUT IN A STATIC RANDOM ACCESS MEMORY ARRAY

4.1 Built-In Self-Test System

4.1.1 BIST Controller

The BIST controller in Fig. 4.1 consists of a test pattern generator (TPG). The test patterns generated by the TPG contain write driver enable signals (T_WE), data inputs (T_Data), sense amplifier enable signals (T_SAE), precharge circuit enable signals (T_PRE), precharge voltages (T_V_pre), a pull-down control signal (P_down), and addresses (T_row_addr, T_col_addr). To generate test row/column addresses, up/down counters are implemented using register-type circuits.

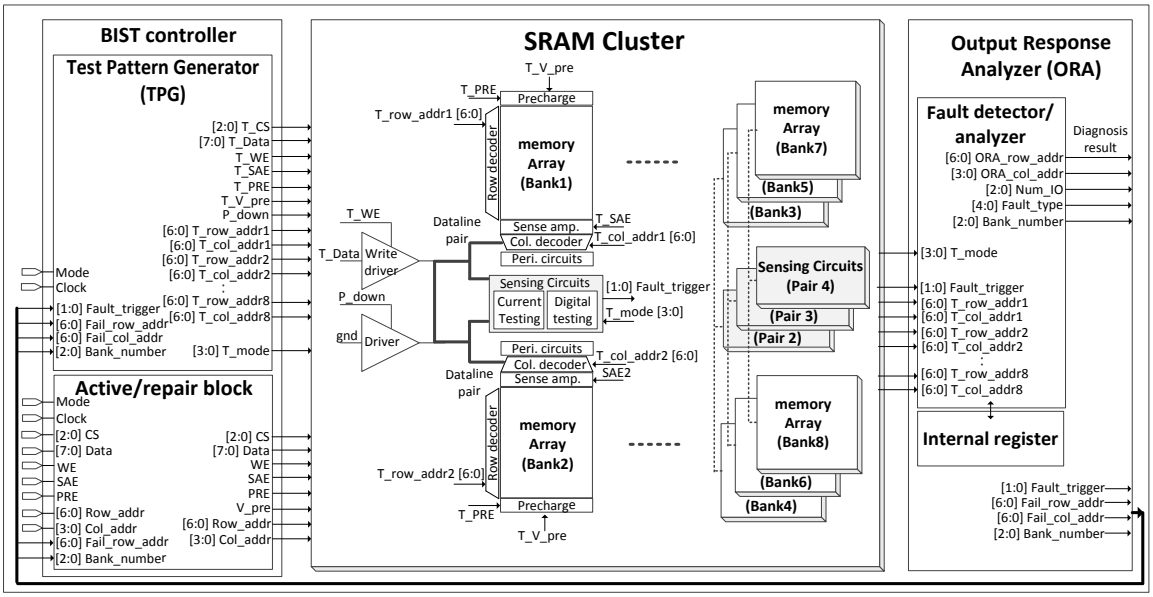

Figure 4.1 System architecture and floorplan of the BIST system.

In test mode, the BIST controller disconnects the test area from some of the control signals from the processor and connects them to the test patterns from the TPG. After the test steps are finished, the active and repair block performs the repair procedure. The SRAM bank contains redundant arrays in each bank and fail row addresses in the registers of the repair block are used to repair memory bit fails due to defects or wearout.

4.1.2 Output Response Analyzer (ORA)

The output response analyzer (ORA) in Fig.4.1 stores the diagnosis results and sends the failure addresses of the faulty cells and their bank number to the TPG and the active/repair block.

In addition, it determines the wearout type and location of the faults through logical analysis of the signals from the sensing circuit (SC). 22 bit registers in the ORA block store the diagnosis result. 17 bits are used for the location of the faulty cells (11 bits for the addresses, three bits for the I/O number, and three bits for the bank number). Another five bits are utilized for the fault type and the specific location of the fault site in the cell from among the 18 possible short/open locations (7 for short groups and 11 for open via/contacts) and six possible BTI locations (see Fig. 3.2).

4.1.3 Built-In Self-Test Area

Figs. 4.1 and 4.2 present the test area in the SRAM system. The SRAM system incorporates eight banks which provide 128Kb memory capacity. Each bank has 16Kb memory cells, with 128 word lines and 128 bitline pairs. The column decoder acts as the bridge between the 128 bitline pairs and eight global data line pairs to be connected to eight I/Os. Hence, 8 global data line pairs for a single bank are selected from 128 bitlines and their complementary bitline-bars. However, to implement the special BIST

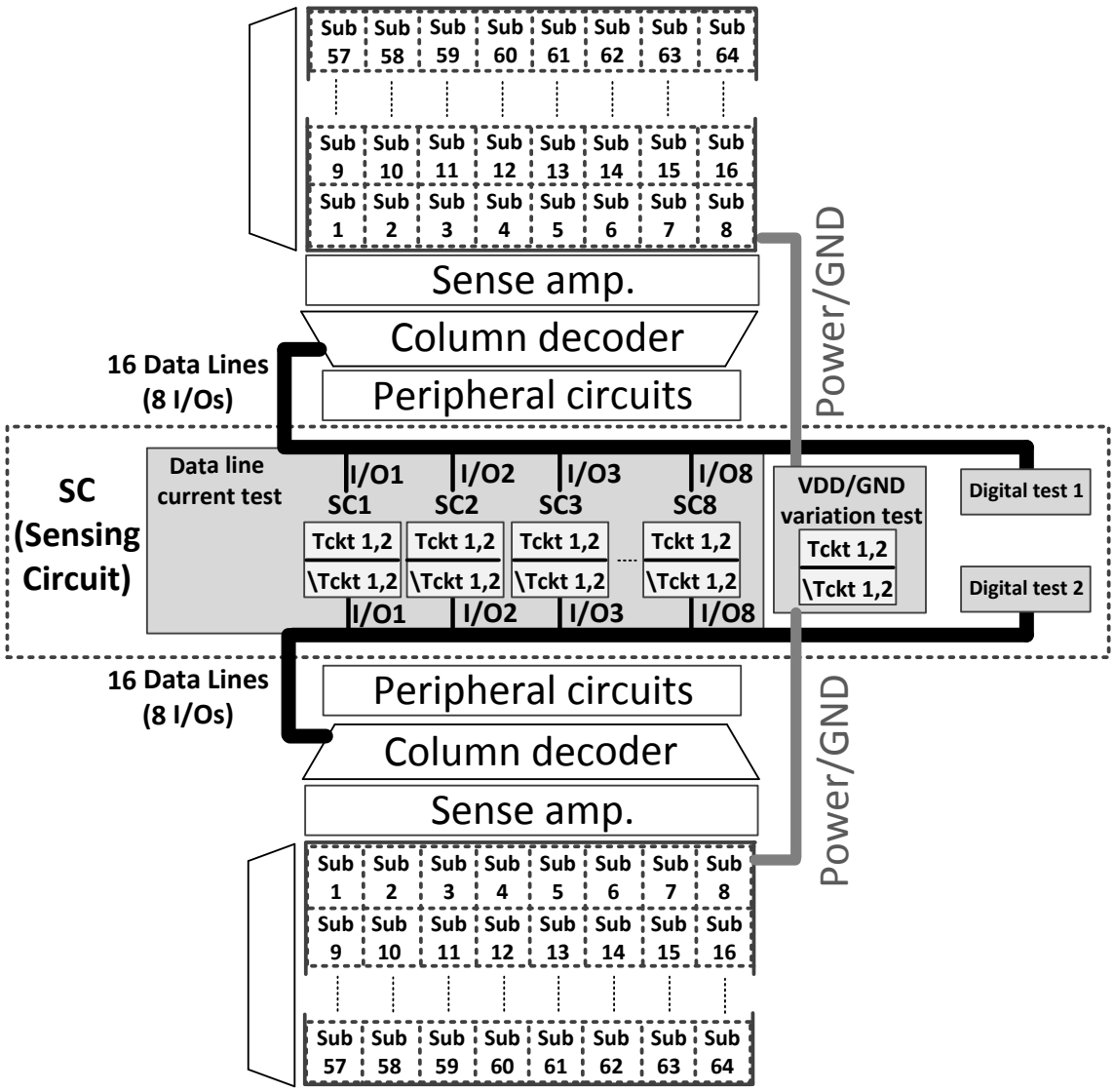

Figure 4.2 Test structures in the built-in self-test area.

algorithm for the diagnosis of wearout mechanisms, we activate and select individual cells for each test step. Hence, test column addresses (T_col_addr) are extended from four bits to seven bits so that they use the additional three bits addresses to select an individual I/O pair from among the eight I/O pairs. Eight SC components are shared by two banks and test 256 bitline pairs, 128 from the upper bank and 128 from the lower bank.

The current test circuit (Tckt) in Fig. 4.3 [66] tests the current variations in the data lines and power/ground networks. The current at input B is subtracted from the current at input A, which results in current I1 (see Fig. 4.3(a)). The current is then fed into the current amplifier and the amplified current, I2, is mirrored onto the current digitizer in Fig. 4.3(b). When the current digitizer detects that I2 is less than a current trigger level generated by the weighted reference current generator shown in Fig. 4.3(c), the output logic is '1'and this triggers the ORA block for diagnosis. We set the current trigger level by tuning widths of transistors (W1-W5) in Fig. 4.3(c). Our BIST system conducts several steps for test algorithms and each test algorithm requires a different current trigger level. To provide the corresponding current trigger level for each test algorithm, we have designed additional logic to control Wn in the current digitizer.

The current test method has been proposed to monitor the BTI, GTDDB, BTDDB, EM, and SIV wearout mechanisms in an SRAM array in [66]-[70]. In these works, we used current testing to locate and diagnose faulty cells suffering from wearout. Faulty cells due to wearout failures are located through a pairwise comparison of cells, one in each bank. By comparing pairs of cells, the cells that develop unusual leakage characteristics and current over time are identified.

To analyze current variations in data lines, each SC unit has two current sensing circuits (Tckt) for bitline testing and two others for bitline-bar testing (\Tckt). Each SC unit monitors a data line pair from the upper bank and another data line pair from the lower bank. Specifically, we connect a data line from the 16 bitlines to both input A of Tckt 1 and input B of Tckt 2 in the SC unit. Another operating current in the data line from the 16 bitlines in the lower bank flows into input B of Tck 1 and input A of Tckt 2

(see Figs. 4.3(a)). The bitline-bars are connected in a similar way to their corresponding SC units.

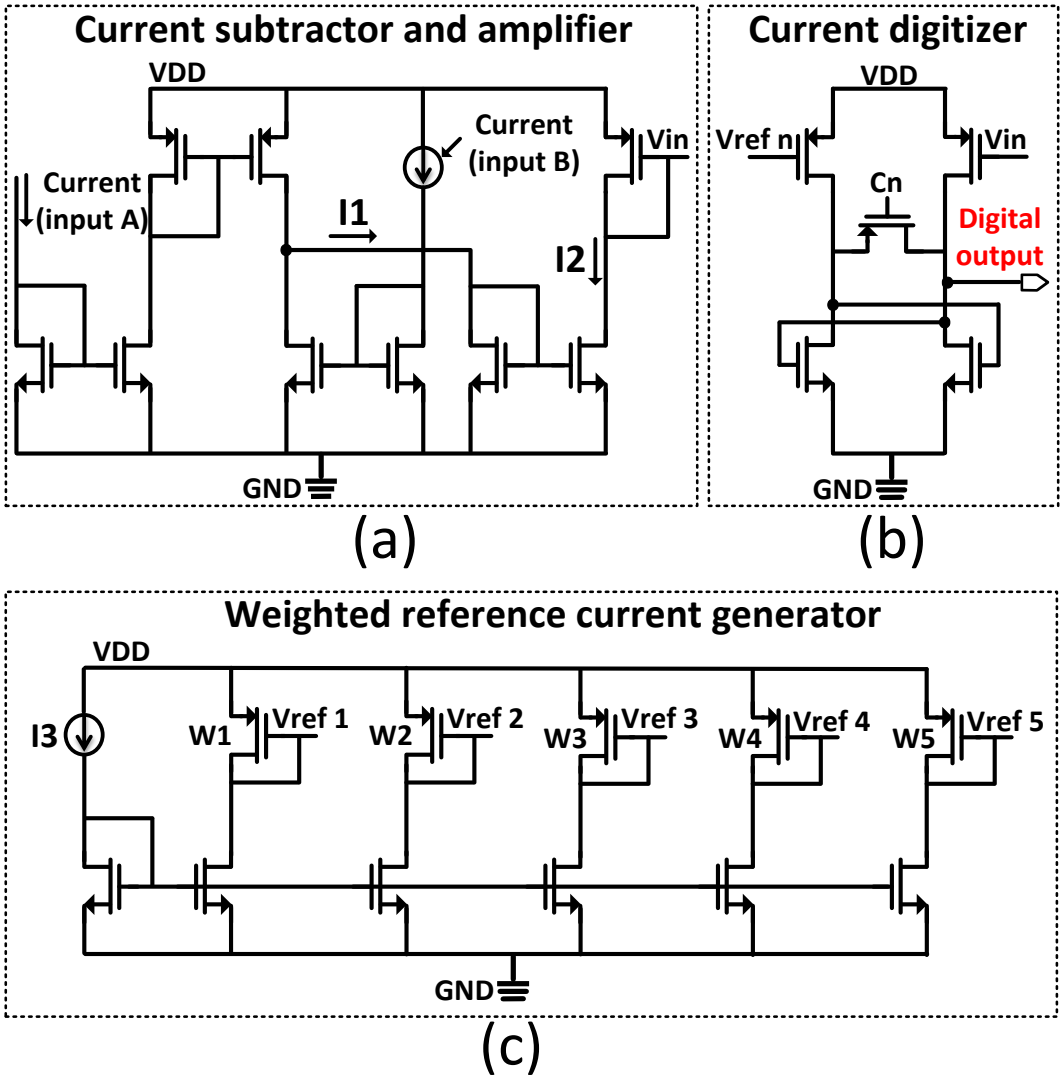

Figure 4.3 Sensing circuit for analysis of current variations due to wearouts in data lines and power/ground networks [66]: (a) current subtractor and amplifier block, (b) current digitizer, and (c) weighted reference current generator.

Four current test circuits detect current variations due to wearout mechanisms in the power/ground networks (see Fig. 4.2). The VDD paths for the upper bank are connected to input A of Tckt 1 and input B of Tckt 2. Another VDD line for the lower

bank is connected to input B of Tckt 1 and input A of Tckt 2. Ground paths for both banks are connected to \Tckt 1 and \Tckt 2 of the same structure.

Finally, two digital blocks check for functional errors in the data lines. The current analysis results are sensitive to the capacitance and resistance of a bitline pair and/or the VDD/GND paths. Thus, a significant mismatch in path length between a cell under test and the sensing circuit from the distance between a reference cell (a good cell) to the same test circuit leads to a false diagnosis result, even if both cells are good cells. We set the maximum allowed length mismatch between the data paths from the cells in the upper and lower banks to be 110um to reduce the chance of diagnosis errors due to mismatch in path length. To keep the length mismatch under the maximum limit, we divide each SRAM bank into 64 sub-blocks. When the cell under test is in the upper bank, we pick a reference cell in the same sub-block of the lower bank as the cell under test.

When the leakage currents from faulty cells are exactly the same, undetectable faults might exist. However, since the leakage currents depend on the degree of wearout, currents from two cells are generally different and shift with wearout. Thus, undetectable faults from matched leakage currents have little impact on the test coverage of the BIST methodology. Nevertheless, if there are undetectable wearout faults, a standard functional test algorithm, such as the March algorithm [3], can be conducted to check the distortion of output patterns. This helps to avoid the worst case scenario where the system fails due to functional faults in the SRAM in the field.

This research has considered only wearout failures in SRAM cells. This is because failures are much more likely in SRAM cells due to the smaller feature sizes.

The BIST system and peripheral circuits are designed with much looser design rules to reduce the vulnerability of these circuits to wearout problems. Moreover, the BIST block is powered down, except in test mode. Hence, the probability of failures due to wearout in the BIST circuity and peripheral circuits is much lower than the failure rate for the SRAM cells.

Keeping the strict policy of ensuring testability with conventional memory BIST, our BIST system is stitched to the data line in parallel, without impacting the timing performance and memory operation functions significantly. Nevertheless, the timing closure for the read and write drivers on the data lines should be carefully conducted to satisfy the timing specification and avoid timing violations, regardless of method to include the BIST system.

Our BIST system is a reconfigurable platform for various cache sizes. To increase test address ranges for a larger SRAM array, we simply add several registers for address counters in the BIST controller and additional registers to store the larger number of addresses of failed cells. Also, if we increase I/O widths for the larger memories, we need more test circuits, such as those shown in Fig. 4.3.

Note that current sensing is sensitive to the capacitance and resistance of the bitline pair. Hence, when we reconfigure the BIST for a larger memory array, we divide the SRAM array into more sub-blocks to keep the maximum allowed length mismatch to 110um, to avoid timing mismatch at the inputs of the test circuit in Fig. 4.3. For example, there need to be 131,072 sub-blocks and reference cells for a 32Mb SRAM array for a bank with 12 bit row addresses, 8 bits column addresses, and 32 I/Os.

The BIST system is designed to operate on each sub-block unit, and the test algorithm is repeated for different sub-blocks (see Fig. 4.2). Having more sub-blocks in a single bank does not impact test coverage. Also, there is no significant area overhead for the customized BIST system for a larger SRAM array since the algorithm for a sub-block is repeated for the larger array. The ratio of area for the customized BIST system presented in Fig. 4.1 to the SRAM system for 128Kb is just 0.67%. The ratio can be further reduced for a larger memory array. Generally, one conventional memory BIST module for the general functional test algorithm, such as the March algorithm, is shared for many memory blocks when implementing a larger SRAM array. Our customized BIST system in Fig. 4.1 is embedded in the conventional BIST circuit using a commercial BIST implementation flow [62]. The ratio of area of the conventional memory BIST system to the 32Mb SRAM system is just 0.043% and the ratio of the customized BIST component to the conventional memory BIST system is just 12.08%.

When the memory is designed with advanced process technologies, the off-state leakage current can be significant [51]. This may lead the current analysis methodologies to be less effective. However, our BIST system uses a current comparison between two cells in the paired sub-blocks. Since the off state leakage depends on the process technology, the initial level of the leakage from the paired cells is still likely to be similar, cancelling out any enhanced leakage. If the reference cell selection controls for the initial leakage currents, then it is likely that the BIST methodologies will work for more scaled technologies. Nevertheless, when we move to more scaled technologies, more reference cells and/or trigger limits may be required for the current tests to better account for variation in initial leakage currents.

The memory BIST platform is usually soft intellectual property (IP), which can be used for many applications without process dependence. However, our BIST system for wearout mechanisms also contains analog sensing circuits and digital test logic. To deliver the analog IP in our BIST system to different chips, there is a need to consider leakage and noise issues carefully in the target design chip. Also, timing closure with the digital test logic and process variations should be carefully checked, with regards to the timing libraries for the specific target process technology and applications.

4.2 BIST Algorithms for Failure Analysis

4.2.1. Overview of Test Algorithm

Fig. 4.4 and Table 4.1 present the test algorithm for wearout mechanisms. The BIST block first conducts screening tests to identify a proper reference cell for each sub-bank, as shown in Fig. 4.2. Test of bitline current using a paired comparison between each cell and the proper reference cell in the paired sub-block identifies the reference cells and all faulty cells, except those with NBTI, PBTI, O1, and O8-O11 faults (see Table 4.2).

Next, for each of the cells identified through the screening test presented in Table 4.2, the BIST controller conducts test steps from CF1 to TF3, shown in Fig. 4.4 to diagnose the cause of failure for each sub- block. More details for test algorithms are provided in Table 4.1. In this step, the reference cells which were found from the wearout screening test are utilized to provide the reference current to the sensing circuit in Fig. 4.3(a). After tests of the faulty cells determined through wearout screening are finished for all sub-blocks in an SRAM bank, the BIST controller starts the TF4 algorithm to identify the remaining faulty cells and their cause of failure.

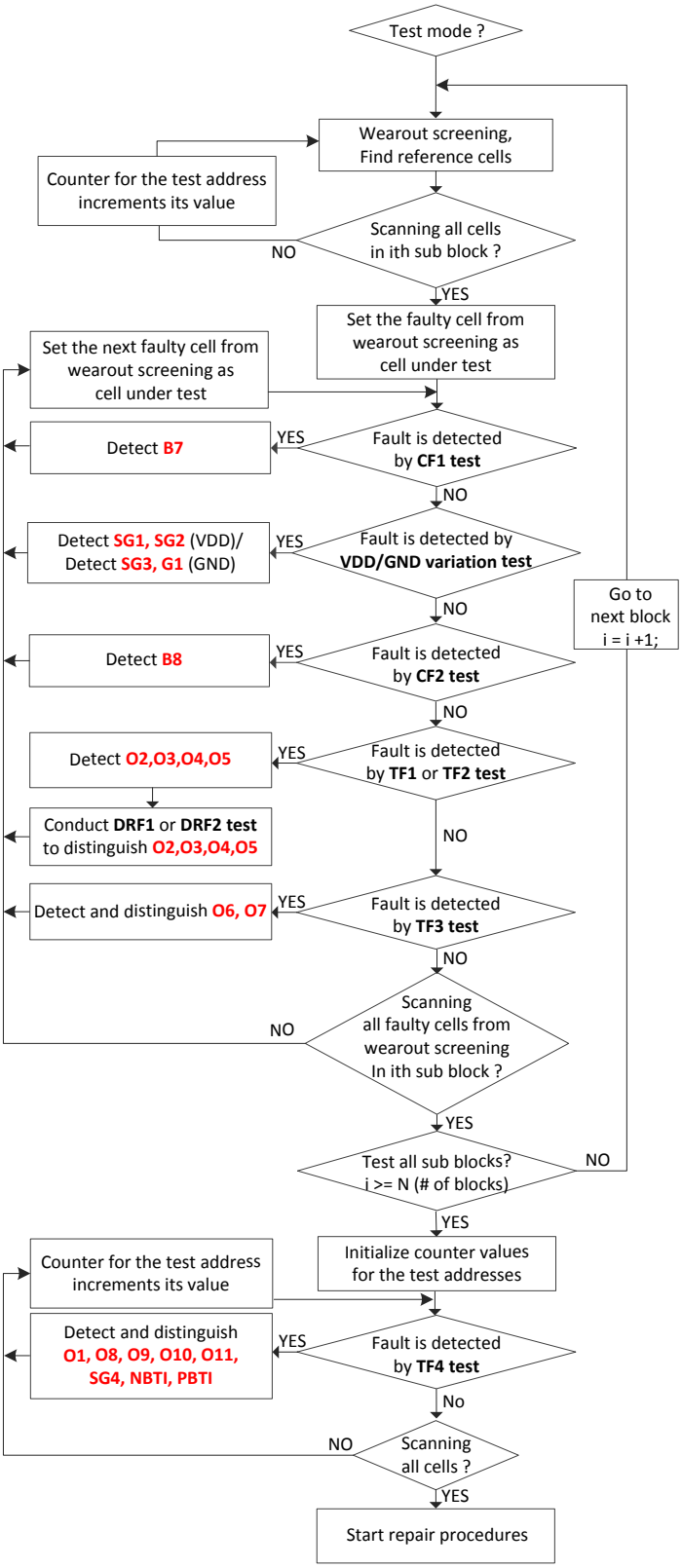

Figure 4.4 Test algorithm for wearout mechanism.

TABLE 4.1. TEST MODES AND PATTERNS FOR DIAGNOSIS OF WEAROUTS

Test mode	**Testing point**	**Test patterns**	**Detected faults**	**name**
Current	Data	(w1,r1,w0,r0) x 2	O2-O7,SG1-SG4,G1, B7, B8	Screen
Digital	Data	(w1,w0)	B7	CF 1
Current	VDD	(w1,r1)	SG1	TV1
Current	VDD	(w0,r0)	SG2	TV2
Current	GND	(w1,r1)	SG3	TG1
Current	GND	(w0,r0)	G1	TG2
Digital	Data	(w1,w0,r0)	B8	CF 2
Current	Data	(w1,w0,r0)	O4, O5	TF 1
Current	Data	(w1,w0,pre[1.2V],r0)	O4 VS O5	DRF 1
Current	Data	(w0,w1,r1)	O2, O3	TF 2
Current	Data	(w0,w1,pre[1.2V],r1)	O2 VS O3	DRF 2
Digital	Data	(w1,w0,r0)	O6, O7	TF 3
Digital	Data	(w1,w0,pre[1.2V],r0) (w1,w0,pre[0V],r0) (w0,w1,pre[0V],r1) (w1,w0,pull_dw[0V],r0) (w0,w1,pull_dw[0V],r1)	O1, O8-O11, SG4, NBTI 1-2 PBTI 1-4	TF4

We set the resistance to 10Ω for resistive bridging defects and to 10MΩ for resistive open defects for the fault models presented in Fig. 3.2. In our simulations for all TDDB, EM, and SIV cases in Fig. 3.2, functional and timing violations during read and write operations occur with 10Ω for resistive bridging models and 10MΩ for resistive open models.

Unlike the resistance models, ΔV_t due to BTI may not distort the read and write data functions significantly. However, BTI in the cell can reduce the read static noise margin (SNM) which guarantees reliable memory operations even with noisy signals [41]. We set ΔV_t to 30% for the tests of for NBTI and PBTI degradations. In the simulations, the read static noise margin is reduced by 7.35% for a 30% ΔV_{tp} shift due to the NBTI mechanism in a cell and by 10.52% for a 30% ΔV_{tn} shift due to PBTI in a cell.

TABLE 4.2. TEST MODES AND PATTERNS

Fault	Data line current variation (max) at input of SC	Wearout Screening
Proper	0 µA	No
NBTI 1,2	0.5 µA >	No
PBTI 1-4	0.5 µA >	No
O1	0.5 µA >	No
O2, O5	**29.34 µA**	**Yes**
O3, O4	**29.31 µA**	**Yes**
O6	**9.8 µA**	**Yes**
O7	**8.2 µA**	**Yes**
O8-O11	0.5 µA >	No
SG1, SG2	**27.4 µA**	**Yes**
SG3, G1	**31.9 µA**	**Yes**
SG4	**22.5 µA**	**Yes**
B7	**64.2 µA**	**Yes**
B8	**27.31 µA**	**Yes**

This level of degradation due to wearout mechanisms is achieved after aging the circuit over 10^{15}s with the four test scenarios in Fig. 1.2 [9]. Hence, we can assume that the significantly degraded cells due to the BTI mechanism can be modeled with the 30% V_t shift. Although ΔV_t of the access transistor due to PBTI does not worsen the read static noise margin significantly, the weak transistors can cause write and read timing faults [41]. Especially, a 30% ΔV_t variation for an access transistor increases the cell access time ($T_{ACCEESS}$) by 11.1%. This can lead to an access timing failure when delay exceeds the maximum tolerate limit (T_{MAX}) with a fast operating clock and tight timing margin [71].

4.2.2. Step 1: Wearout Screening and Finding Reference Cells

The wearout screening test consists of two sub procedures involving current testing of the data lines using the SC in Fig. 4.3. To distinguish the faulty cells from proper cells without fault, we use W1 in Fig. 4.3(c) to set the trigger level to 4.0µA for wearout screening. This is larger than the maximum variation in current (1.06µA) that can be observed between two good cells even with 10% corner process variations. The trigger level can be set, by adding a margin for noise.

The first step is to find the reference cells which do not have any fault. During this step, a cell in the upper bank is paired with a cell in the lower bank for the test. If the current is the same, then both cells can be reference cells and their addresses are captured in register-type circuits under the name *Reg_refer1*. These proper cells can be references for all other cells in the paired bank in the same sector.

When the current is different, both cells are included in the suspect set (as illustrated in Fig. 4.5(a)). When the cells are in the suspect set, the algorithm has to search for proper reference cells, since the cells in the suspect set cannot be proper reference cells. To do this, the counters increase the register value for both test column addresses, until the SC does not detect a leakage current difference (as illustrated in Fig. 4.5(b)). The result for the cell location is stored in *Reg_refer1*.

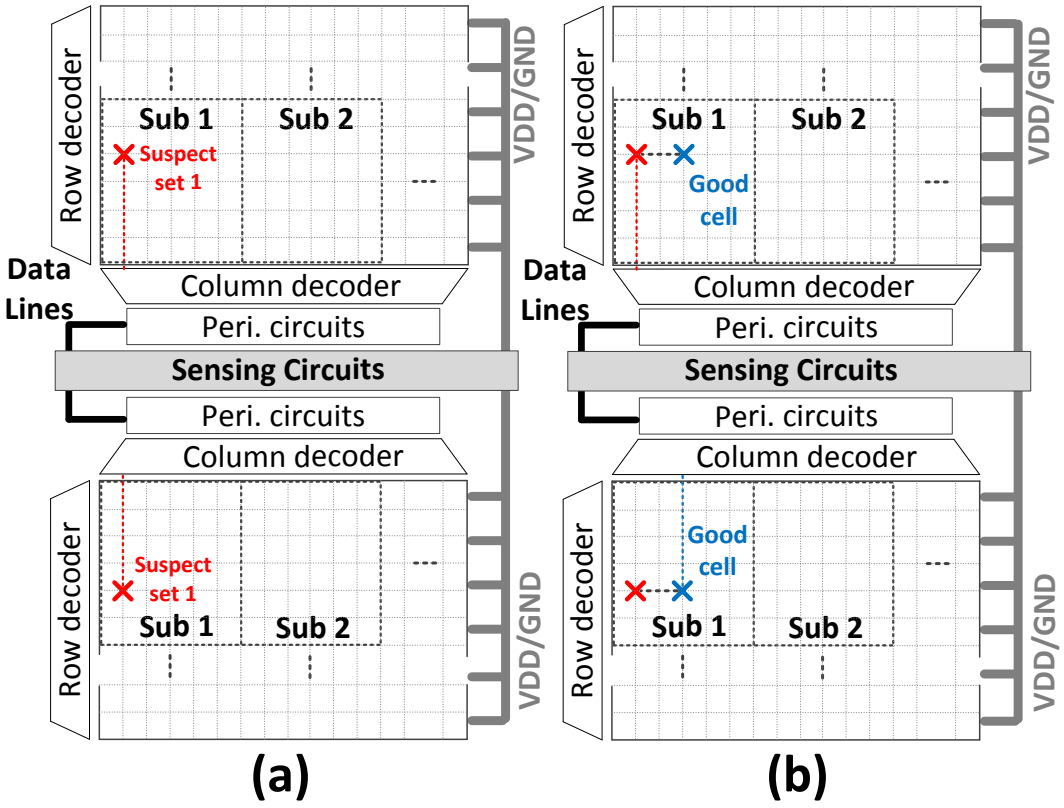

Figure 4.5 Test architecture and algorithm for wearout screening test: (a) Finding suspect sets, and (b) Finding proper reference cells.

After the wearout screening test for each sub-block in a single bank is completed, the proper reference cell in each sub-bank is used for the other current test steps in Table 4.1. All SRAM cells are tested during the step to identify reference cells, even though only one reference cell is stored for each sector since the scan through all cells also identifies a suspect set of potentially faulty cells. Hence, all cells are paired with their complementary cell in the paired bank and the ORA stores the cell addresses in *Reg_suspect1* if a current difference is detected.

It is necessary to determine which of the two complementary cells is faulty in the suspect set presented in Fig. 4.5 after the proper reference cells have been identified. Each cell in the suspect set is tested using the proper reference cell in the complementary bank to determine whether it is faulty.

4.2.3. Step 2: Coupling Fault (CF1) Diagnosis for B7 fault

The BIST system tests the identified faulty cells to determine their cause of wearout (see Table 4.1). The first fault model to be diagnosed is B7 (see Fig. 4.4). The B7 fault is induced by dielectric breakdown between bitline-bar connected to cell 1 and bitline connected to cell 2, which increases the bitline-bar and bitline loads significantly (see Fig. 3.2). A write driver cannot pull up the voltage of bitline-bar for cell 1 to 0.6V due to the increased load.

For the detection of B7, the TPG generates the (w1, w0) pattern and analyzes the voltage patterns on the bitline pair with digital logic. During the write '1' operation, the digital block stores both digitized values from the bitline pair in register-type circuits with the names Rg_1 and Rg_2. During the subsequent write '0' operation, the digital logic stores the digitized values in $Rg_3$and Rg_4. The counter counts clock edges to set the

capture time for the digitized values. The digital logic detects and diagnoses the cells (cell 1 in Fig. 3.3) which contain B7 on bitline-bar and generates the fault trigger signal *(F_{B7})* using the following Boolean equation:

$$F_{B7} = Rg_1 \cap !\,Rg_2 \cap !\,Rg_3 \cap !\,Rg_4\,. \tag{4.1}$$

Table 4.3 shows that the F_{B7} signal is generated only if a cell with the B7 fault on bitline-bar is tested.

TABLE 4.3 SIMULATION RESULTS FOR THE CF1 TEST WITH B7 FAULT

Fault	**Write '1' operation**		**Write '0' operation**	
	Bitline logic	Bitline-bar logic	Bitline logic	Bitline-bar logic
Proper	Logic 1	Logic 0	Logic 0	Logic 1
NBTI 1,2	Logic 1	Logic 0	Logic 0	Logic 1
PBTI 1-4	Logic 1	Logic 0	Logic 0	Logic 1
O1 – O11	Logic 1	Logic 0	Logic 0	Logic 1
SG1-SG4	Logic 1	Logic 0	Logic 0	Logic 1
G1, B8	Logic 1	Logic 0	Logic 0	Logic 1
B7 (cell 1)	**Logic 1**	**Logic 0**	**Logic 0**	**Logic 0 (0.54V)**
Reg.	Rg_1	Rg_2	Rg_3	Rg_4

4.2.4. Step 3: Current Variation Analysis of Power/Ground Distribution Networks for Diagnosis of SG1-SG4

The BIST system next starts a current analysis on the VDD lines to screen bridging faults between VDD and a signal node (B5, B6, G6, and G8 in Fig. 3.2). We connect the SCs in Fig. 4.3 to the VDD paths for both upper and lower banks. The BIST controller sends the test addresses of cells under test to detect the current variation. To make VDD/GND variation more visible so that the sensing circuit can detect it, an additional test structure between the global power/ground network and an SRAM bank is added, as shown in Fig. 4.6.

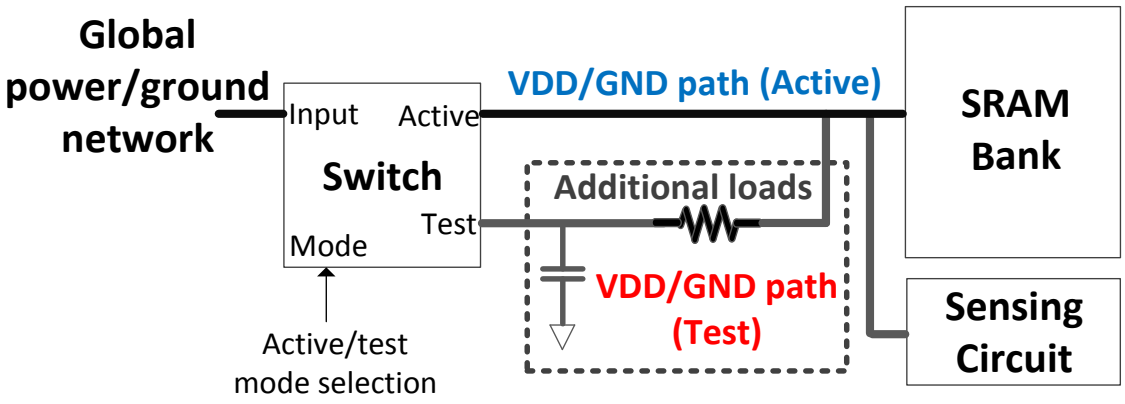

Figure 4.6 Additional structure for VDD/GND variation test in the memory system.

In test mode, a switch in the test area switches the global VDD/GND paths to another VDD/GND test path with the inserted larger resistance. Due to the larger noise, VDD/GND variations from bridging faults are easily detected. During write '1' and read '1' operations, SG1 (B6 and G6 in Table 3.1) becomes the bridge enabling the current in the VDD line to flow to the GND path. Also, the leakage path due to SG2 (B5,G8) leads current in VDD to flow to GND path during write '0' and read '0' operations (see Fig. 3.2).

The leakage current between a signal line and GND is induced by B1, G1, and G3 (see Fig. 3.2). If the signal line is shorted to GND through the leakage path due to SG3 (B1, G3), the GND level temporarily goes up during the write '1' and read '1' operations with the TG1 pattern in Table 4.1. It increases due to G1 during write '0' and read '0' operations with this (w0,r0) pattern.

Table 4.4 shows that SG1 and SG2 are detected with analysis of the VDD path. To distinguish them from other faults, the reference device width, W2, in Fig. 4.3(c) is set to a current trigger level of 7.4μA. SG3 and G1 are distinguished from other mechanisms

through GND path analysis. W3 sets the current trigger level to 4.1µA for the detection of SG3 and G1.

TABLE 4.4 VDD/GND VARIATION ANALYSIS RESULTS FOR SHORT GROUPS

Fault	**VDD current variation (max) at input of SC**	**GND current variation (max) at input of SC**
Proper	0 µA	0 µA
NBTI 1,2	Less than 0.1 µA	Less than 0.1 µA
PBTI 1-4	Less than 0.1 µA	Less than 0.1 µA
O1	0.3 µA	Less than 0.1 µA
O2-O5	2 µA	0.5 µA
O6-O7	3.0 µA	Less than 0.1 µA
O8-O10	Less than 0.1 µA	Less than 0.1 µA
O11	1.3 µA	Less than 0.1 µA
SG1, SG2	**13.2 µA**	0.2 µA
SG3, G1	3.1 µA	**7.3 µA**
SG4	2.8 µA	0.1 µA
B8	6.7 µA	Less than 0.1 µA

4.2.5. Step 4: Coupling Fault (CF2) Diagnosis for B8

The BIST controller generates the (w1, w0, r0) pattern for the diagnosis of B8 (see Fig. 4.7). For this test pattern, bitline sense amplifiers are turned off during the read operation. Bitline mismatch does not occur during the read operation when there is no wearout, when the bitline sense amplifiers are turned off.

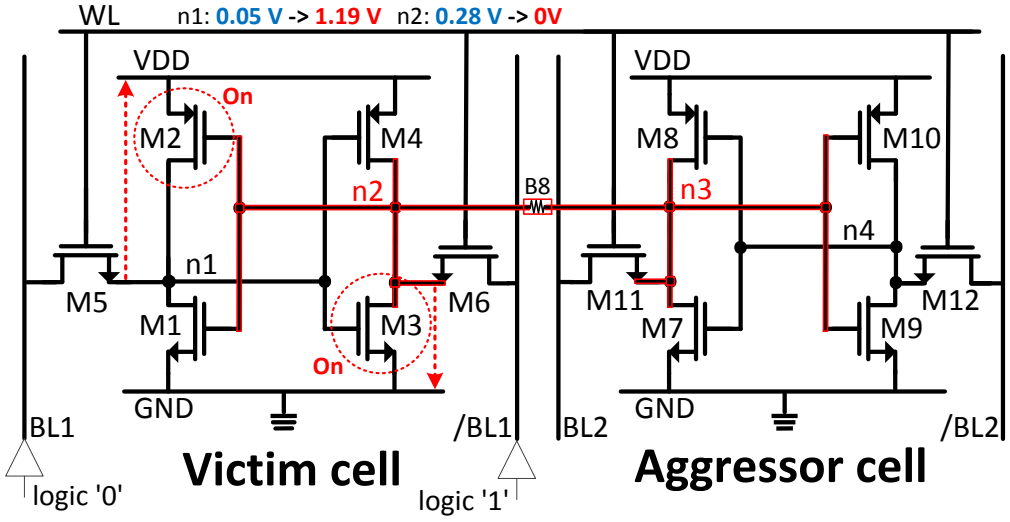

Figure 4.7 Write ‘0’ and read ‘0’ operations for victim and aggressor cells with the B8 coupling fault in an SRAM array.

Fig. 4.7 presents write ‘0’ and read ‘0’ in a victim cell. The load on the n2 node is increased because it is stuck to n3 in an aggressor cell through the B8 fault. The B8 fault can break the load balance between the n1 and n2 nodes in the victim cell. The write driver drives the bitline to logic ‘0’ and bitline-bar to logic ‘1’ during the write ‘0’ operation. The M5 transistor in the victim cell pulls down the n1 node to 0.05V. However, the M6 transistor cannot drive the n2 node to logic ‘1' due to the increased load at n2. The M6 transistor pulls up the voltage on n2 in the victim cell to 0.28V.

Although the n1 node (0.05V) is connected to the gate of PMOS M4 and the n2 node (0.28V) is fed into the gate of M2, M2 pulls up the n1 node instead of M4, since the load on n2 is much larger than the load on n1. M2 transistor pulls up the n1 node to 1.19V, and this turns M3 on, pulling down the n2 node to 0V (see Fig. 4.7).

Fig. 4.8(a) shows that the M2 transistor is turned on, and this leads to an increased bitline voltage. Also, the bitline-bar voltage starts to decrease since the M3 transistor is turned on. The current comparison in Fig.4.8(a) shows that the source current of the M2 transistor increases from 16.3µA to 20.5µA and that of the M4 transistor decreases from 27.3µA to 0 during the read ‘0’. This shows that the M2 transistor pulls up the signal node instead of the M4 transistor. The digitized values from the bitline and bitline-bar vary from logic ‘0’ to ‘1’ and from ‘1’ to ‘0’, respectively (see Fig. 4.8(b)).

The waveform for the digitized value patterns is used to identify the victim cells with B8 faults (see Table 4.5). The digital logic stores digitized values at four test points from the bitline pair in register-type circuits with names Rg_5, Rg_6, Rg_7, and Rg_8(see Fig. 4.8(b)). The digital circuit diagnoses the victim cell with the B8 fault using:

$$F_{B8} = (!\,Rg_5) \cap (Rg_6) \cap (Rg_7) \cap (!\,Rg_8). \quad (4.2)$$

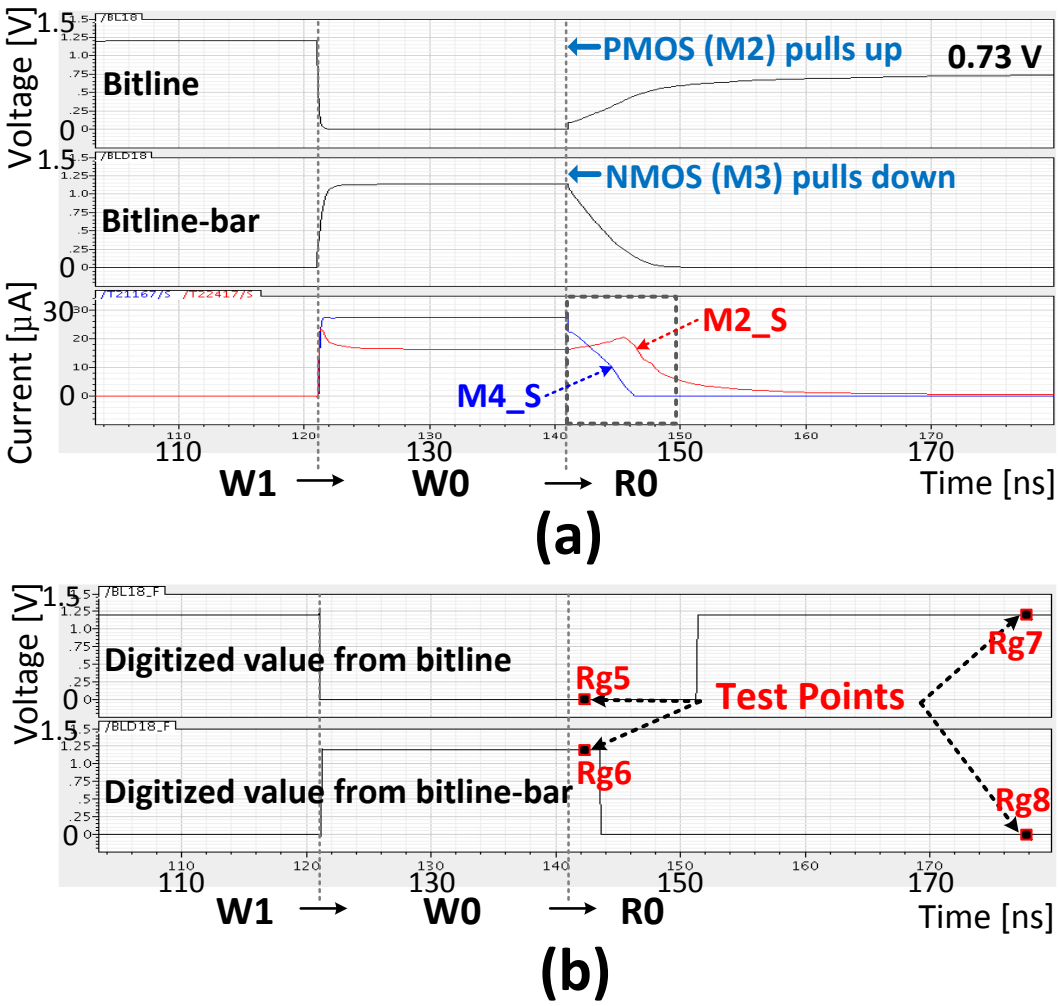

Figure 4.8 Simulation results for the victim cell with B8 fault with the pattern (w1, w0, r0): (a) bitline pair voltages and current at the sources of transistors M2 and M4, and (b) digitized values from the bitline pair.

TABLE 4.5 SIMULATION RESULTS FOR CF2 DURING THE READ '0' OPERATION

Fault	Bitline voltage [V]	Bitline-bar voltage [V]	Bitline logic	Bitline-bar logic
Proper	0	1.08	0	1
NBTI 1,2	0	1.08	0	1
PBTI 1-4	0	1.08	0	1
O1	0->0.22	1.08	0	1
O2, O3	0	1.08	0	1
O4, O5	0->0.21	1.08->0	0	1->0
O6	0	1.08->1.01	0	1
O7, O10	0	1.08	0	1
O11	0->0.23	1.08->0.87	0	1
SG4	0->0.55	1.08->0.55	0	1->0
B8 (victim)	**0->0.73**	**1.08->0**	**0->1**	**1->0**

4.2.6. Step 5: TF1, TF2, DRF1, and DRF2 Tests for O2–O5

The TF1 (transition fault) algorithm is used to detect O4 and O5, as shown in Table 4.1. The complementary TF2 pattern detects O2 and O3. The BIST controller follows with the DRF1 (data retention fault) pattern to distinguish O5 from O4 and the DRF2 test to distinguish O2 from O3 (see Table 4.1). Bitline sense amplifiers are turned off for these test patterns.

Fig. 4.9(a) presents the TF1 algorithm with the O4 fault. With the O4 or O5 fault, the write '1' operation is done properly. The write driver drives bitline-bar to 0V, and the M6 drives n2 node to ground. Node n2 is discharged to under 0.6V through the M6 transistor, and the n1 value changes to logic '1'. During the write '0' operation after the M3 transistor is turned on, n1 becomes stuck at logic '1', since M5 cannot pull down n1 to logic '0' due to the large load caused by O4. Then the M6 transistor cannot pull up the n2 node to 0.6V due to the path from n2 to ground through M3. During the read logic '0' operation after the write operations, the voltage on bitline-bar is discharged from 1.09 V to 0.01V, since it is connected to the n2 node which holds 0V.

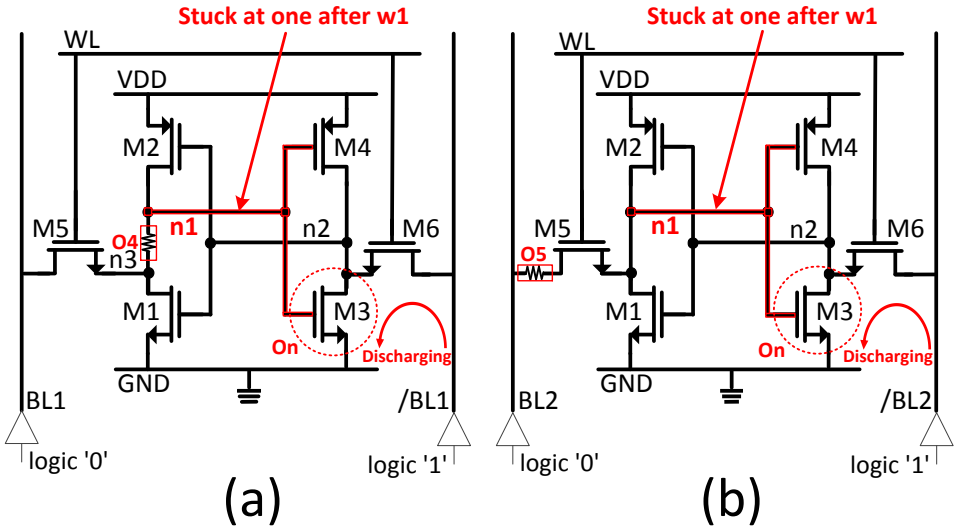

Figure 4.9 Write '0' operation with the TF1 algorithm presented in Table 4.1 for (a) an SRAM cell with O4 fault, and (b) an SRAM cell with O5 fault.

TABLE 4.6 SIMULATION RESULTS FOR THE TF1 AND TF2 ALGORITHMS

Fault	**Read '0' (TF1)**		**Read '1' (TF2)**	
	\BL voltage	Current variation from \BL (max)	BL voltage	Current variation from BL (max)
Proper	1.09 V	0 µA	1.09 V	0 µA
NBTI 1,2	1.09 V	0.1> µA	1.09 V	0.1> µA
PBTI 1-2	1.09 V	0.1> µA	1.09 V	0.1> µA
PBTI 3	1.09V	0.1> µA	1.09V	1.9 µA
PBTI 4	1.09 V	1.9 µA	1.09V	0.1> µA
O1	1.09 V	0.3 µA	1.09 V	0.3 µA
O2, O3	1.09 V	4.5 µA	**> 0.01 V**	**29.3 µA**
O4, O5	**> 0.01 V**	**29.3 µA**	1.09 V	4.5 µA
O6	0.96 V	1.0 µA	1.09 V	0.1 µA
O7	1.09 V	0.1 µA	0.96 V	1.0 µA
O8	1.09 V	4.5 µA	1.09 V	0.1> µA
O9	1.09 V	4.6 µA	1.09 V	4.6 µA
O10	1.09 V	0.1> µA	1.09 V	4.5 µA
O11	1.09 V	4.5 µA	1.09 V	4.5 µA
SG4	0.55 V	4.0 µA	0.55 V	4.0 µA

When the O5 fault is placed between a bitline and the M5 transistor, the problem with the TF1 pattern is the same (see Fig. 4.9(b)). Since the write driver cannot pull down the n1 node due to the inserted large load during the write '0' operation, n1 is stuck at logic '1', and this prevents the M6 transistor from pulling up the n2 node. Thus, the voltage on bitline-bar varies from 1.09V to 0.01V. Table 4.6 shows that the voltage on bitline-bar with O4 or O5 is different from other wearout faults during the read logic '0' operation, and this results in current variation in the bitline-bar data line. W4 in Fig. 4.3(c) can be used to set the reference current to 6.6µA for O4 and O5 detection.

Additional test steps are needed to distinguish O4 from O5. The DRF1 test step in Table 4.1 analyzes data retention properties during a very long read operation. The (w1, w0) test pattern causes the n1 node in Fig. 4.9 to be stuck at logic '1'. When the write '0' is completed, the BIST controller sends T_PRE (the precharge circuit enable signal) and T_V_pre (1.2V) to the bank (see Fig. 4.1). The bitline is pulled up above

logic ‘1’, and the read ‘0’ starts. During a very long read ‘0’ operation (20µs), n3 node in a cell with an O4 fault is charged to 986mV, and the M5 transistor prevents the bitline from being discharged, as illustrated in Fig. 4.10. On the other hand, since the M5 transistor in a cell with an O5 fault cannot hold the bitline charge due to the large inserted load, the bitline connected to the cell with an O5 fault is easily discharged. Fig. 4.10 shows that a voltage difference between the two types of faults is detected, and the faults are distinguished with 20µs test time.

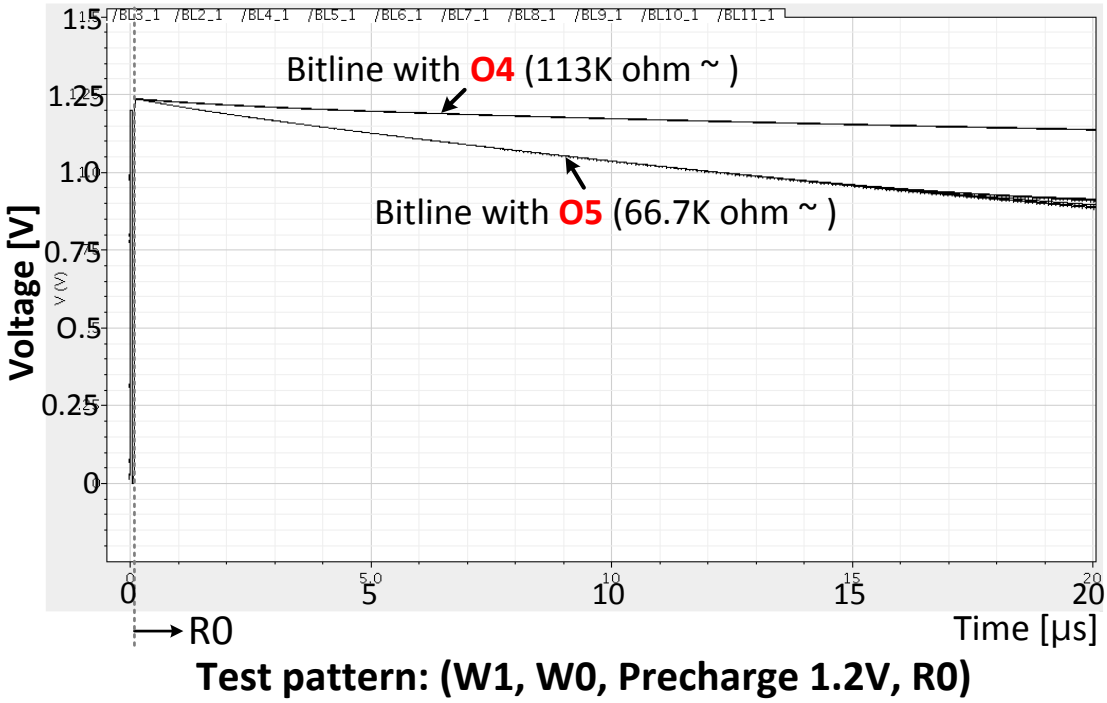

Figure 4.10 DRF1 algorithm to distinguish O4 from O5.

4.2.7. Step 6: TF3 Pattern for O6 and O7

For the TF3 algorithm, the BIST system writes logic ‘1’ with a period of 70ns in faulty cells to set the initial values of n1 and n3 to ‘1’ and n2 and n4 to ‘0’ (see Fig. 4.11). Then write ‘0’ and read ‘0’ operations are executed. Sense amplifiers are turned off during the read ‘0’ operation with this pattern.

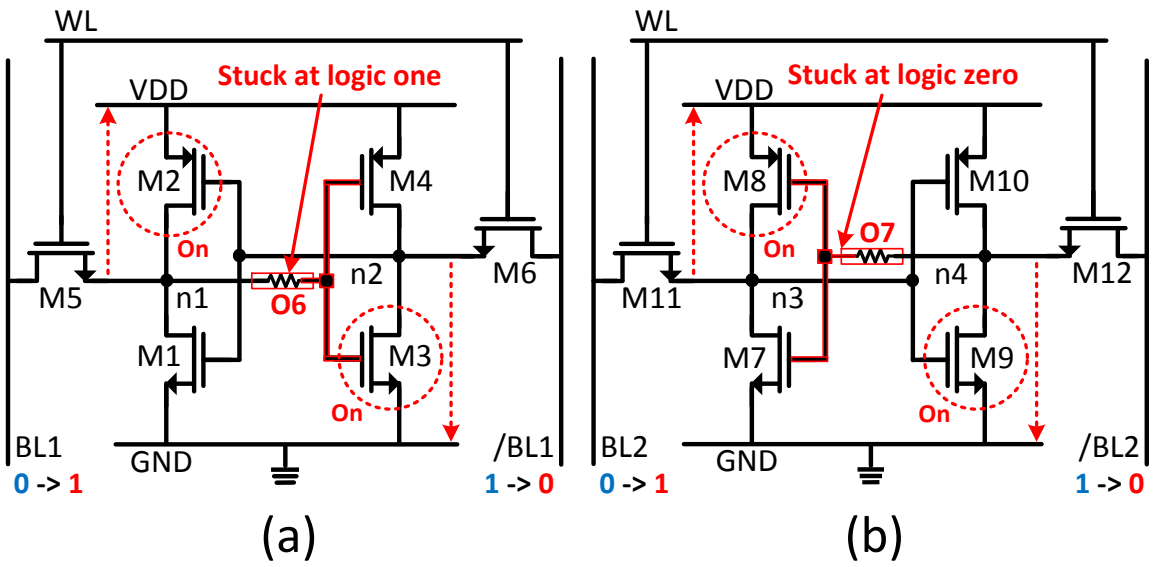

Figure 4.11 Write and read logic '0' after a write '1' operation in an SRAM cell with (a) O6 and (b) O7.

Fig. 4.11(a) presents the write '0' and read '0' operations for a faulty cell containing O6 at the n1 node. During the short write '0' (5ns), M5 and M1 cannot pull down the gate of M4 to 0V due to the large resistance. The gate becomes stuck at logic '1', and this turns M3 transistor on. Since the current from M6 is directly discharged through M3 during the write '0', the voltage on n2 (0V) does not change, and M2 stays on. When the read '0' starts, M3 transistor pulls down bitline-bar from 1.13 V to 0V, and the M2 transistor pulls up the bitline voltage from 0.01V to 0.74 V. Fig. 4.12(a) shows that both the M2 and M3 transistors turn on at 106ns.

Fig. 4.11(b) presents a cell with O7 at the n4 node with the same initial conditions. Similarly, the M10 and M12 transistors cannot pull up the gate of M8 to logic '1' during the short write '0' operation, pulling down n3 node to 0V. When the read '0' starts and the write driver is disconnected, the M8 transistor pulls up n3 from '0' to '1'. The new value of the n3 node turns M9 transistor on, pulling down the bitline-bar voltage from

1.21V to 0V. More test time is needed for M8 to pull n3 up than for M3 to pull n2 down. Thus, the M9 transistor is turned on at 111.9 ns, leading the bitline-bar voltage to decrease at 111.9 ns.

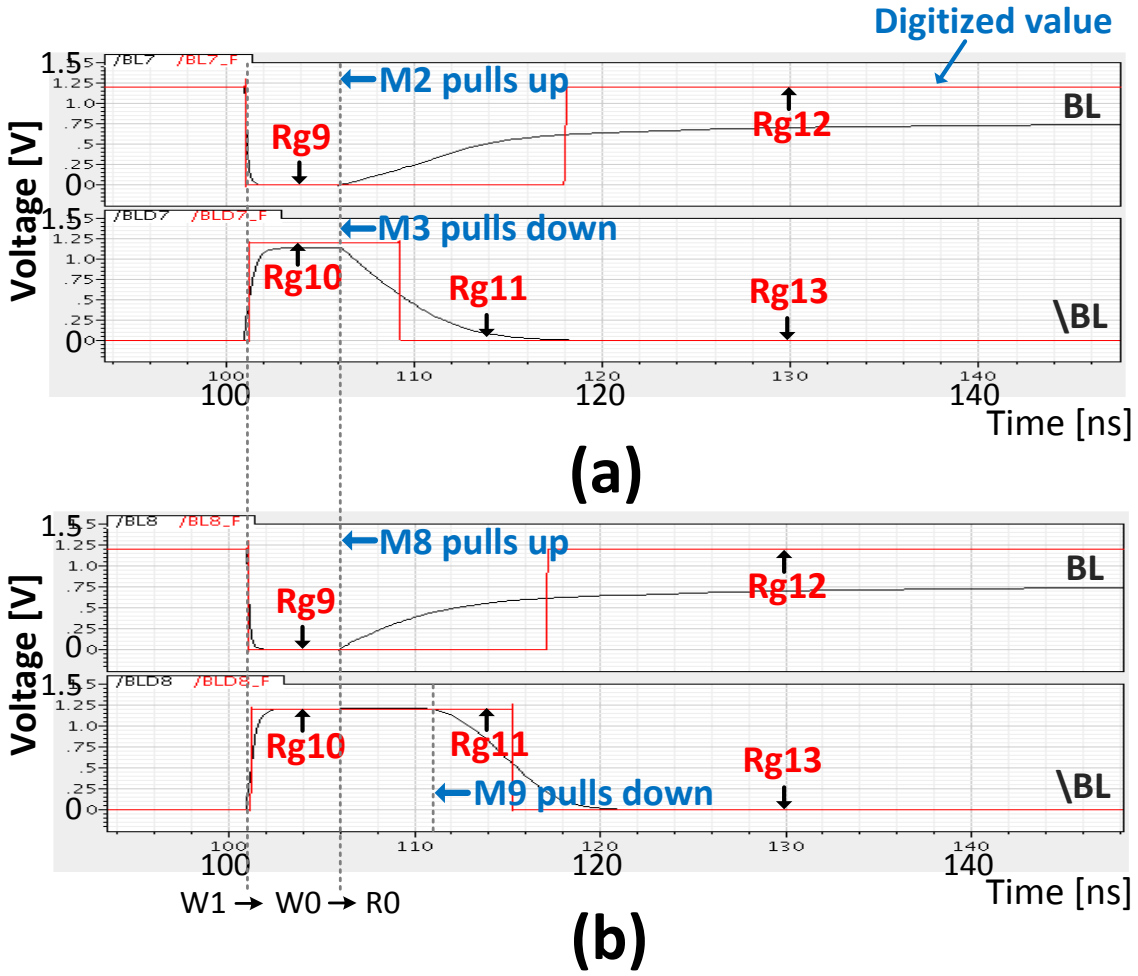

Figure 4.12 Bitline pair voltages and their digitized values for a cell with test pattern (w1, w0, r0) (a) for an O6 fault and (b) for an O7 fault.

The waveforms for bitline pairs, presented in Fig. 4.12, are used to distinguish O6 from O7 (see Table 4.7). The BIST system stores five time points (Rg_9~Rg_{13}) for each data line pair, as presented in Fig. 4.12. Since the falling edge of bitline-bar is different for the O6 and O7 faults, Rg_{11} can be used to distinguish these faults. With the stored

register values, the ORA block diagnoses the O6 and O7 faults using the following equations:

$$F_{O6} = (!\,Rg_9) \cap (Rg_{10}) \cap (!\,Rg_{11}) \cap (Rg_{12}) \cap (!\,Rg_{13}) \quad (4.3)$$

$$F_{O7} = (!\,Rg_9) \cap (Rg_{10}) \cap (Rg_{11}) \cap (Rg_{12}) \cap (!\,Rg_{13}). \quad (4.4)$$

TABLE 4.7 SIMULATION RESULTS FOR THE TF3 TEST FOR O6 AND O7 (READ '0')

Fault	Bitline voltage [V]	Bitline-bar voltage [V]	Logic from bitline	Logic from bitline-bar
Proper	0	1.2	0	1
NBTI 1,2	0	1.2	0	1
PBTI 1-4	0	1.2	0	1
O1	0	1.2	0	1
O6	**0.01 -> 0.74**	**1.13-> 0**	**0 -> 1**	**1 -> 0**
O7	**0.01 -> 0.74**	**1.21 -> 0**	**0 -> 1**	**1 -> 0**
O8–O11	0	1.2	0	1
SG4	0.02 -> 0.55	1.14-> 0.55	0	1 -> 0

4.2.8. Step 7: TF4 Algorithm for Remaining Faults

All faults detected with current screening have been identified in the previous sections. However, O1, O8-O11, SG4, NBTI, and PBTI cannot be detected through the current screening test. Hence, diagnosing these faults requires scanning the entire array with the TF4 pattern (see Fig. 4.4 and Table 4.1). The test pattern contains several sub-steps, with a non-standard precharge between write and read operations. For instance, the BIST controller sets the precharge voltage to 0V to pull down a bitline pair during the 20ns hold time before the read operation for sub-steps 2 and 3 of TF4 in Table 4.8. The BIST controller sends the pull-down control signal (P_down) to the driver between the write and read operations.

TABLE 4.8 SIMULATION RESULTS FOR THE TF4 TEST DURING READ OPERATIONS

Fault	**Sub step 1 w1 -> w0-> pre 1.2V (20ns) -> r0**		**Sub step 2 w1 -> w0-> pre 0V (20ns) -> r0**		**Sub step 3 w0 -> w1-> pre 0V (20ns) -> r1**	
	BL[V]	\BL[V]	BL[V]	\BL[V]	BL[V]	\BL[V]
Proper	0	1	0	1	1	0
NBTI 1	0	1	0	1	1	0
NBTI 2	0	1	0	1	1	0
PBTI 1	0	1	0	1	1	0
PBTI 2	0	1	0	1	1	0
PBTI 3	0	1	0	1	1	0
PBTI 4	0	1	0	1	1	0
O1	1	1	0	1	1	0
O8	0	1	1	0	1	0
O9	0	1	0	0	0	0
O10	0	1	0	1	0	1
O11	1	1	0	0	0	0
SG4	0	0	0	0	0	0
Reg.	Rg_{14}	Rg_{15}	Rg_{16}	Rg_{17}	Rg_{18}	Rg_{19}

Fault	**Sub step 4 w1 -> w0-> pull-down 0V -> r0**		**Sub step 5 w0 -> w1-> pull-down 0V -> r1**	
	BL[V]	\BL[V]	BL[V]	\BL[V]
Proper	0	1 / 1	1 / 1	0
NBTI 1	0	1 / 1	0 / 0	1
NBTI 2	1	0 / 0	1 / 1	0
PBTI 1	0	0 / 1	1 / 1	0
PBTI 2	0	1 / 1	0 / 1	0
PBTI 3	1	0 / 0	0 / 1	0
PBTI 4	0	0 / 1	0 / 0	1
O1	0	0 / 1	0 / 1	0
O8	1	0 / 0	1 / 1	0
O9	0	0 / 0	0 / 0	0
O10	0	1 / 1	0 / 0	1
O11	0	0 / 0	0 / 0	0
SG4	0	0 / 0	0 /0	0
Reg.	Rg_{20}	Rg_{21}/Rg_{22}	Rg_{23}/Rg_{24}	Rg_{25}

Defect SG4 is the resistive-short defect between the internal cell nodes (see Fig. 3.2 and Table 3.1). Since the signal nodes are connected via the BTDDB or GTDDB mechanism, both voltages on a bitline pair go up to 0.54 V during the read '0' operation. However, 0.54 V cannot flip the digitized value to logic '1' with 1.2V VDD. Table 4.8 shows that digitized values from a bitline pair with faults in SG4 for all sub-cases of TF4

are logic ‘0’. It also shows that cells with these faults are distinguished from the fault-free cell.

Defect O11 disconnects the access transistors from the cell under test. The logic on a bitline pair cannot be changed after precharging since the access transistors are not functional (see Fig. 3.2).

Defect O1 is the worn out contact between sources of NMOS cell transistors and ground (see Fig. 3.2). Either the M7 or M9 transistor cannot pull down a bitline or a bitline-bar during the read operation. We use the test pattern (w1, w0, precharge (1.2 V), r0) to test the ability of the cell to pull down. For a proper cell without a fault, M7 in Fig. 3.2 discharges the bitline to 0V during the read ‘0’ operation. However, the M7 transistor in a faulty cell with O1 cannot discharge the bitline due to the large resistance between the M7 transistor and ground. Table 4.8 shows that the test result for a cell with the O1 fault is different from that of a proper cell for sub-step 1 of TF4.

Defect O9 is the worn out contact between sources of a PMOS device and VDD (see Fig. 3.2). Similar to detection of the O1 fault, the large resistance keeps M8 or M10 from pulling up a bitline or bitline-bar during a read operation. Similarly, sub-step 2 and sub-step 4 determine whether the M10 transistor can pull up the bitline-bar properly. When the read ‘0’ starts, the M10 transistor in the faulty cell with O9 cannot pull up a bitline- bar due to the large inserted resistance (O9). We can see that both the bitline and bitline-bar are logic ‘0’ in the sub-step 2-5 columns of Table 4.8.

Defects O8 and O10 are the worn out contacts between a drain of a PMOS transistor and a signal path in an SRAM cell (see Fig. 3.2). To detect the O8 fault in the cell, the test pattern in sub-step 2 of TF4 is utilized. When the read ‘0’ starts, the M10

transistor in Fig. 3.2 has to hold the signal node connected to its drain at logic '1', and the M7 transistor holds its drain node at logic '0' for proper operation. However since the M10 transistor cannot hold the node at logic '1' due to the large resistance of the O8 fault, the node is discharged, leading to a change of the logic value to logic '0'. There is also a change of the logic value at the drain node of the M7 transistor to logic '1'. This changes the logic on the bitline to 'high' (0.75V) and the logic on bitline-bar to 'low' (0V), as presented in the sub-step 2 column of Table 4.8**.**

To detect the O10 fault, the pattern is the opposite (w0, w1, precharge (0V), r1). The logic values on the bitline pair are swapped for the O10 fault with sub-step 3 for the same reason as for O8 fault with sub-step 2 in Table 4.8.

BTI Degradation NBTI degradation in an SRAM cell causes the V_{tp} of PMOS M2 (NBTI 1) and PMOS M4 (NBTI 2) to shift (see Fig. 3.2). V_{tp} for our process technology (90nm technology) is –175mV, and we set ΔV_{tp} for our simulations to -52.5mV (30%). The effect of NBTI degradation is similar to the O8 and O10 faults presented in Fig. 3.2. Hence, the test algorithm must distinguish NBTI 1 from the O10 fault and NBTI 2 from the O8 fault. With the NBTI 1 degradation effect, the M2 transistor has a weaker drive strength when pulling up the internal node connected to its drain. The PMOS M8 transistor with O10 loses its driving ability when pulling up the same node. The driving ability of the M8 transistor with O10 is much weaker than the M2 transistor with NBTI 1.

Although NBTI degradation leads an SRAM cell to be skewed and weakens the driving ability of the PMOS devices, the PMOS can hold the charge on the internal node connected to its drain, unlike with the O8 or O10 fault. Hence, for sub-step 2 and 3 in

Table 4.8, the skewed property from ΔV_{tp} due to NBTI degradation cannot swap the logic states of the internal node if the absolute value of ΔV_{tp} is less than 84 mV, even in the presence of process variations. Therefore NBTI is distinguished from O8 and O10.

To detect NBTI degradation, there is a need to conduct additional steps, sub-steps 4 and 5 in Table 4.8. The sub-steps are similar to sub-steps 2 and 3, except during the pull-down process the access transistors are turned on, so that there is no voltage difference between the internal nodes in the SRAM cell.

When the voltages of the internal nodes are almost the same, the PMOS without NBTI pulls up the internal node, and the PMOS with NBTI is turned off. For the NBTI 1 model, the voltage on bitline-bar always goes high with the test pattern of sub-steps 4 and 5 (see Table 4.8). On the other hand, the voltage on the bitline is always pulled up with NBTI 2 during the same test pattern. Hence, sub-step 4 detects NBTI 2 degradation, and sub-step 5 detects NBTI 1 degradation.

The PBTI mechanism shifts the $V_{tn}$of NMOS transistors in a cell. See Fig. 3.2 for definitions of PBTI 1, PBTI 2, PBTI 3, and PBTI 4. For PBTI 1, the M7 transistor in Fig. 3.2 drives M10, which pulls up bitline-bar during the read '0' operation with sub-step 4. However, the weaker driving ability of the M7 transistor with PBTI 1 causes a delay for the bitline-bar to be pulled up to logic '1'. Hence, the logic on bitline-bar is logic '0' at the first data capturing point (Rg_{21}) and logic '1' at the second data capturing point, Rg_{22} (see Table 4.8). Similarly, there is a delay for the bitline voltage to be pulled up to high from the SRAM cell with PBTI 2 during read '1' of sub-step 5 (see the Rg_{23} and Rg_{24} values in Table 4.8).

For PBTI 4, the M12 transistor has a weaker driving ability. After pulling down the bitline pair with sub-step 5, the M11 transistor without PBTI degradation turns on M10 earlier, turning off the M8 transistor even if the stored value on the node connected to the drain of the M10 transistor is logic '0'. For the read '0' operation with sub-step 4, the M12 transistor, which is driven by M10 transistor, pulls up bitline-bar to logic '1'. However, since the M12 transistor with PBTI degradation is weaker, there is a delay for bitline-bar to be pulled up, and the delay is detected using Rg_{21} and Rg_{22} in Fig. 4.13. Table 4.8 indicates the swapped logic values of the bitline pair with sub-step 5 for an SRAM cell with PBTI 4. The PBTI 3 model is similarly detected with sub-step 4 and by the delay to pull up the bitline with sub-step 5.

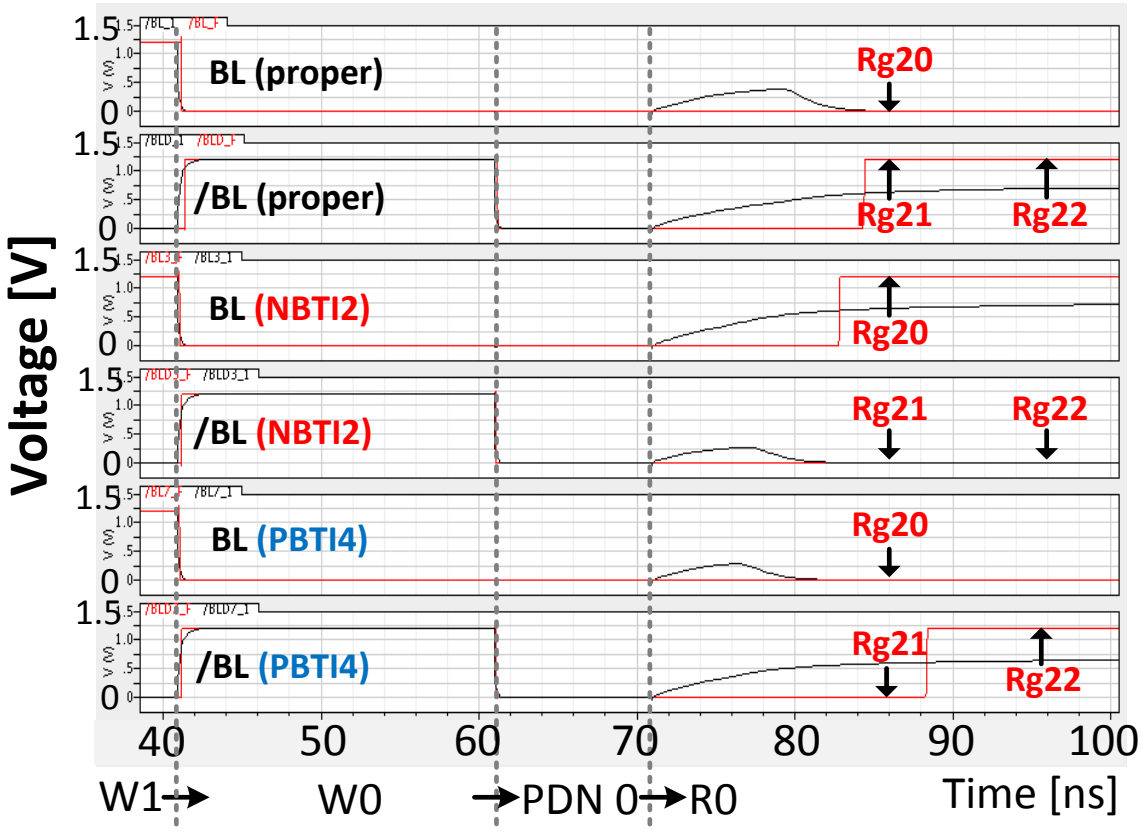

Figure 4.13 Simulation of the voltages on bitline pairs from a proper cell, a cell with NBTI2, and a cell with PBTI4 for sub-step 4 of TF4 pattern.

Boolean Equations for Diagnosis: The digitized values from the bitline with the TF4 pattern are stored in Rg_{14}, Rg_{16},Rg_{18}, Rg_{20}, Rg_{23}, and Rg_{24}. Also, the values from bitline-bar are stored in Rg_{15}, Rg_{17}, Rg_{19}, Rg_{21}, Rg_{22}, and Rg_{25} with the same pattern. Using digital test logic, we diagnose all of the wearout mechanisms.

4.2.9. Detectable Range for Wearout Mechanisms With BIST

Table 4.9 shows a summary of detectable ranges of inserted resistances for all possible mechanisms with the maximum allowed bitline length mismatch. We apply 10% process variation corners for Range 2 in Table 4.9 when determining the detectable range. Range 1, presented in Table 4.9, is the detectable range without process variations. Process variations degrade detection of wearout. Circuits with more extreme variations in process parameters will suffer from delayed detection of resistive shorts and opens since resistance degrades with time.

TABLE 4.9 DETECTABLE RANGE OF INSERTED RESISTANCES FOR EACH FAULT

Fault	**Range 1 [Ω]**	**Range 2 (with PV)[Ω]**	**The worst range [Ω]**
O1	> 179K	> 184.7K	> 184.7K
O2,O5	> 63.9K	> 66.7K	> 66.7K
O3,O4	> 109K	> 113K	> 113K
O6	> 2.38M	> 2.62M	> 2.62M
O7	> 4.21M	> 4.37M	> 4.37M
O8,O10	> 170K	> 230K	> 230K
O9	> 374K	> 400K	> 400K
O11	> 5.69M	> 5.94M	> 5.94M
SG1-2	< 15.4K	< 15.4K	< 15.4K
SG3,G1	< 2.41K	< 1.51K	< 1.51K
SG4	< 29.6K	< 26.8K	< 26.8K
B7	< 0.625K	< 0.583K	< 0.583K
B8	< 18.6K	< 18.2K	< 18.2K
NBTI	9.8-57.8%	16.4-48%	16.4-48%
PBTI 1,2	12.4-100%	20.9-100%	20.9-100%
PBTI 3,4	20.8 -75%	29.5-72%	29.5-72%

The threshold voltage variation due to process variations limits the effectiveness of the BIST technique. The critical limit on threshold voltage variation is 34.51%, which

makes the B7 fault in Table 4.9 undiagnosable. The faults which have resistance value equal to the limited range in Table 4.9 are less detectable in the presence of process variations. If the process is controlled well, keeping the process variations under the critical limit, our BIST system detects and distinguishes all possible wearout mechanisms in an SRAM array.

4.3 Statistical Failure Analysis to Separate Wearout Distributions for GTDDB vs. BTDDB and EM vs. SIV

For short groups (SG1-4) due to the GTDDB and BTDDB mechanisms in Table 3.1 and open groups (OG1-3) due to the EM and SIV mechanisms in Table 3.2, the cause of a fault cannot be identified using only electrical tests since both mechanisms cause the same shorts or opens.

Hence, there is a need to find an additional analysis methodology to determine the cause of wearout. We propose to diagnose the fraction of failures for each confounded mechanism with statistical analysis combined with field test results from BIST and the reliability simulator. The fraction of failures from GTDDB vs. BTDDB and EM vs. SIV are estimated by matching the failure rate of each fault site from BIST to simulation data from a reliability simulator [4]-[10].

For short groups due to GTDDB and BTDDB, the characteristic lifetime is $\eta_{i,j,k}$ and the shape parameter is $\beta_{i,j,k}$. k is an index for the short group (SG1-4) in ith cell and j indicates the short location within the short groups (see Table 3.1). For example, $\eta_{30000,2,4}$ for GTDDB is the characteristic lifetime for G4 (j=2) of SG4 (k=4) in the 30,000th cell (i=30000) (see Fig. 3.2 and Table 3.1). For open groups due to EM and SIV, the characteristic lifetime and the shape parameter are $\eta_{l,m}$ and $\beta_{l,m}$, respectively. m is

the index for the open group (OG1-3) in the lth cell. The simulator estimates $\eta_{i,j,k}$ and $\eta_{l,m}$ and the corresponding values of the shape parameter for all possible wearout sites in the SRAM system using benchmarks and use scenarios to determine the stress profiles.

$\beta_{i,j,k}$ (for shorts) and $\beta_{l,m}$ (for opens) are assumed to have a constant value for each mechanism. Then, the Weibull characteristic lifetimes for each fault group for shorts, η_k , and for each group for opens, η_m can be computed with

$$\eta_k = \left(\sum_j \sum_{i=1}^{32768} \frac{1}{\eta_{i,j,k}^{\beta}} \right)^{-1/\beta} \tag{4.5}$$

and

$$\eta_m = \left(\sum_{l=1}^{32768} \frac{1}{\eta_{l,m}^{\beta}} \right)^{-1/\beta}. \tag{4.6}$$

The overall lifetime of the SRAM system, η_{chip}, for each mechanism is the solution of

$$1 = \sum_k \left(\eta_{chip} / \eta_k \right)^{\beta} \text{ and } 1 = \sum_m \left(\eta_{chip} / \eta_m \right)^{\beta}. \tag{4.7}$$

Given, η_k, k=1,…,4, for each short group in Table 3.1 and η_m, m=1,…,3, for each open group in Table 3.2, the probability that the failure is located in the k^{th} group of locations and the m^{th} group of locations is

$$P_k = \left(\eta_{chip} / \eta_k \right)^{\beta} \text{ and } P_m = \left(\eta_{chip} / \eta_m \right)^{\beta}. \tag{4.8}$$

The relative frequency of different short groups depends on the relative frequency of each wearout mechanism, which is estimated by the relative frequency of GTDDB (γ) and BTDDB ($1 - \gamma$). The observed overall relative frequency of the short groups, $P_{k,chip}$, is a function of the probabilities of GTDDB ($P_{k,GTDDB}$) and BTDDB ($P_{k,BTDDB}$), i.e.,

$$P_{k,chip} = \gamma P_{k,GTDDB} + (1 - \gamma) P_{k,BTDDB}. \tag{4.9}$$

$P_{k,GTDDB}$ and $P_{k,BTDDB}$ are the probabilities of failure of each short group when GTDDB and BTDDB were the only failure mechanism, respectively. The relative frequency of each open group is estimated based on the relative frequency of SIV (λ) and EM $(1-\lambda)$. Overall, the relative frequency of the fault sites in the SRAM chip, $P_{m,chip}$, is

$$P_{m,chip} = \lambda P_{m,SIV} + (1-\lambda)P_{m,EM}. \tag{4.10}$$

where the probabilities of SIV and EM for each open group are $P_{m,SIV}$ and $P_{m,EM}$, respectively.

Fig. 4.14(a) presents $P_{k,chip}$ for GTDDB and BTDDB with different use scenarios by changing the relative fraction of GTDDB and BTDDB failures, γ. Similarly, Fig. 4.14(b) shows the failure rate, $P_{m,chip}$, due to EM and SIV, by varying the relative fraction of SIV and EM failures, λ. This is the expected failure rate computed by simulation.

$P_{k,chip}$ and $P_{m,chip}$ are obtained from the observed fraction of failures for each short and open group respectively, using electrical test with our BIST methodology. When we collect the relative failure rates for each group from the chip with the BIST system, we can estimate $P_{k,chip}$ and $P_{m,chip}$. $P_{k,GTDDB}$, $P_{k,BTDDB}$, $P_{m,SIV}$,and $P_{m,EM}$ are computed with the reliability simulator [4]-[10].

Specifically, the reliability simulator computes the lifetime of each cell due to each mechanism in equations (4.5),(4.6), and then the probability of failures at each site is estimated with equations (4.7),(4.8). The parameters, γ and λ are computed by regression:

$$\gamma = \frac{\sum_{k=1}^{4}(P_{k,GTDDB}-P_{k,BTDDB})(P_{k,chip}-P_{k,BTDDB})}{\sum_{k=1}^{4}(P_{k,GTDDB}-P_{k,BTDDB})^2} \tag{4.11}$$

and

$$\lambda = \frac{\sum_{m=1}^{3}(P_{m,SIV} - P_{m,EM})(P_{m,chip} - P_{m,EM})}{\sum_{m=1}^{3}(P_{m,SIV} - P_{m,EM})^2} \quad . \tag{4.12}$$

By matching relative probabilities of each group from $P_{k,chip}$ or $P_{m,chip}$ to the probability of failures of each mechanism from the reliability simulator, we can estimate the values of γ and λ.

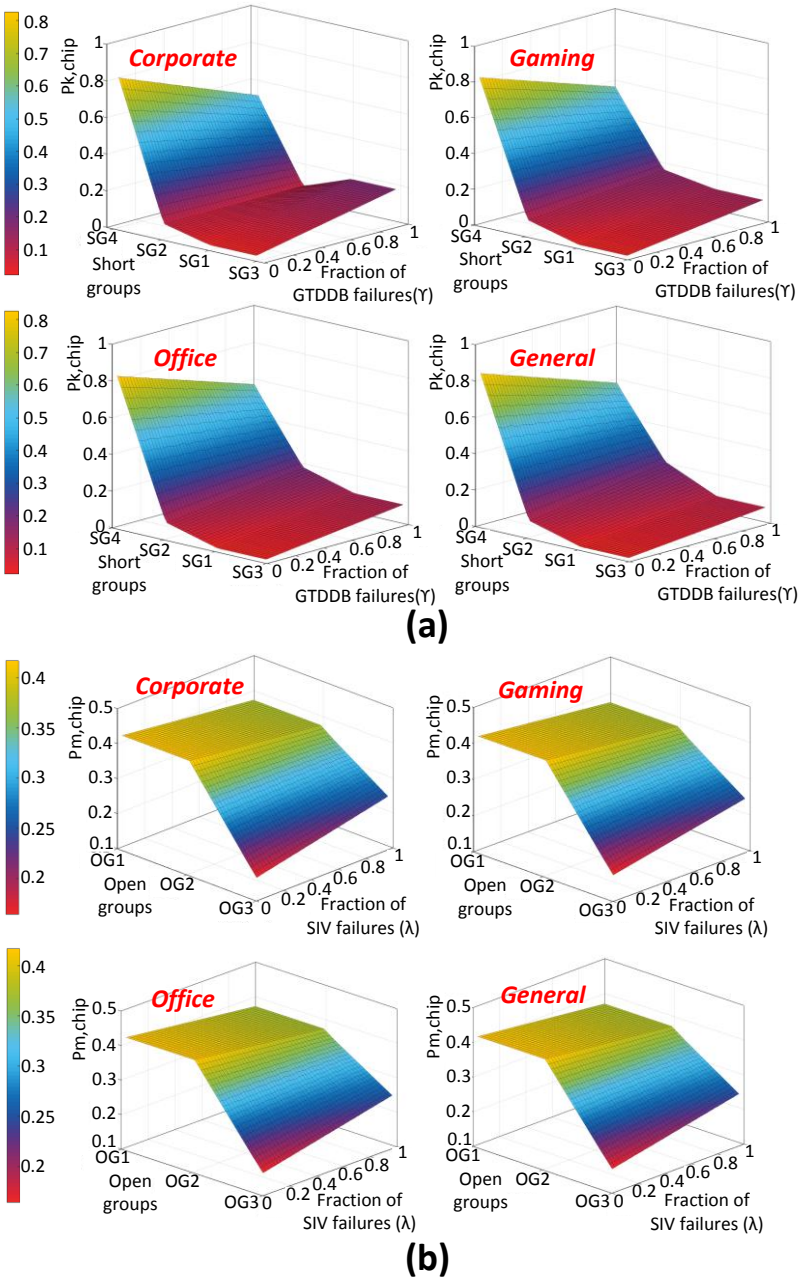

Figure 4.14 Failure rate distribution using a reliability simulator which determines the stress distribution of SRAM cells inside a microprocessor with different use scenarios (a) for GTDDB and BTDDB, and (b) for EM and SIV.

For the error analysis, we assume that $P_{k,GTDDB}$, $P_{k,BTDDB}$,$P_{m,EM}$, and $P_{m,SIV}$ from the simulation result are affected by errors with normal distributions with standard deviation, σ. This is because there is a gap between the simulation data and the real lifetime values. We assume the measured values of $P_{k,chip}$ and $P_{m,chip}$ are known for given values of γ and λ, and we estimate γ and λ with equations (4.11) and (4.12), respectively. The computed values, γ and λ, do not match the true values of γ and λ. Then, equation (4.11) is solved for γ' and equation (4.12) is solved for λ' by varying σ for the normal distribution for the error added to $P_{k,GTDDB}$, $P_{k,BTDDB}$,$P_{m,EM}$, and $P_{m,SIV}$. Fig. 4.15 shows the errors for both cases.

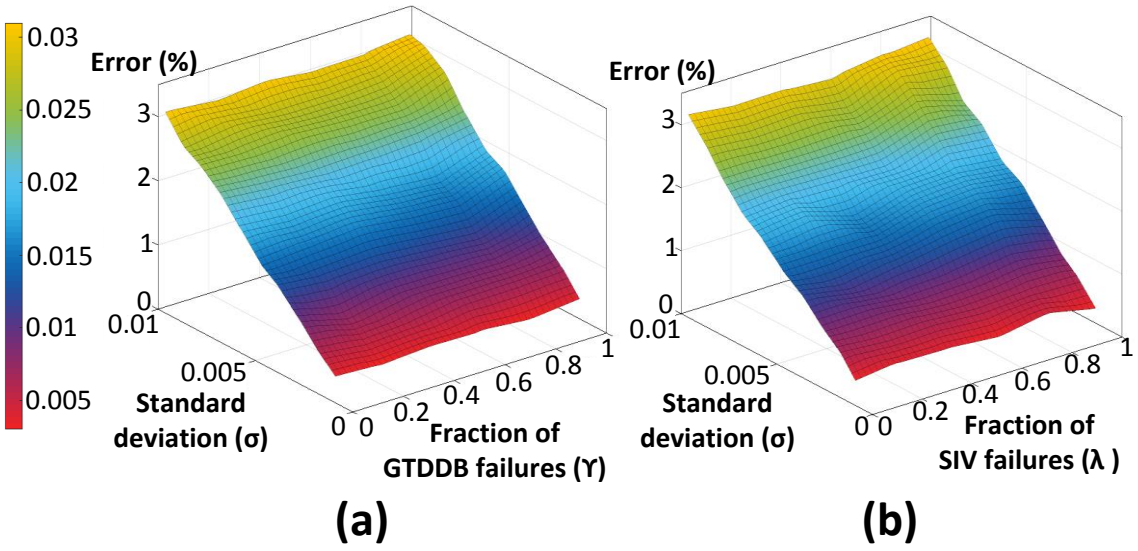

Figure 4.15 The error analysis for (a) $\gamma - \gamma'$ with $P_{k,GTDDB}$ and $P_{k,BTDDB}$, (b) $\lambda - \lambda'$ with $P_{m,SIV}$ and $P_{m,EM}$ for general use scenario.

If there is uncertainty in the actual use scenario, there can be errors in estimation of the probabilities of failure. If the simulator uses the gaming use scenario or the office work scenario instead of the corporate use scenario, the errors in estimation of

probabilities of failure are shown in Fig. 4.16. The use scenario affects the lifetimes from the simulator and the probabilities that the failure is observed at each site (equations (4.5)-(4.8)).

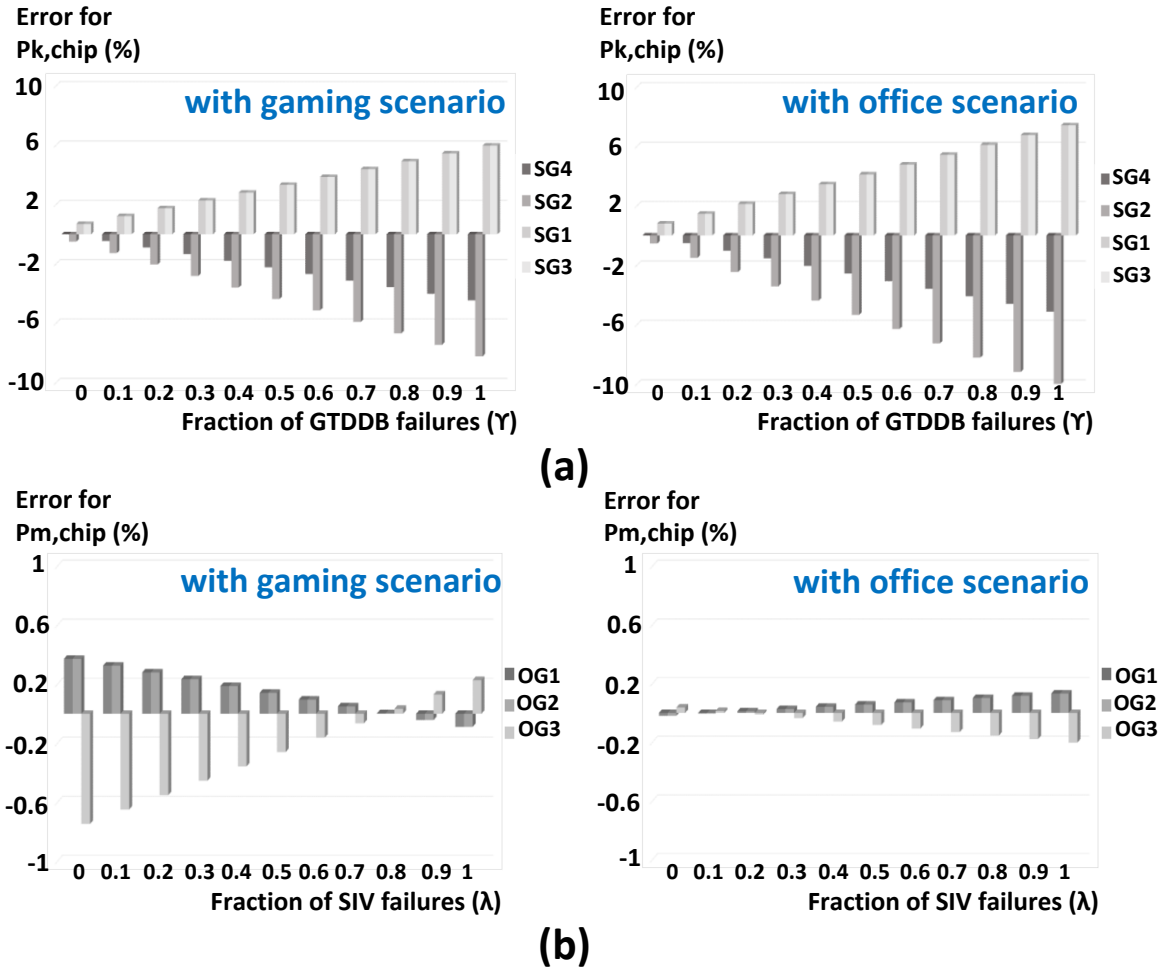

Figure 4.16 Error for P_{chip} when simulation data from the wrong use scenario (gaming senario and office scenario) are used for failure analysis for the "true" corporate scenario for (a) GTDDB and BTDDB and (b) EM and SIV.

4.4 Optimization of Stress Acceleration Tests for Statistical Analysis

The wearout mechanisms are a function of temperature and voltage (see equations (3.1)-(3.4)). For the correct fitting of the parameters in the equations, stress acceleration tests should be conducted to collect enough data for various voltage and temperature sets. The electrical failure signatures for the short sites due to GTDDB and BTDDB and the open sites due to EM and SIV are the same. For the parameter fitting for those mechanisms, the statistical analysis presented in section 4.3 should be combined with the stress acceleration tests with various test conditions.

Process variations within or between dies can create variations in each probability of failure value in Fig. 4.14. When we use the test conditions for short and open groups which are vulnerable to process variations, this can cause the errors in the fraction of failures for each mechanism in equations (4.11) and (4.12), leading to errors in the parameter fittings. Also, although more stress acceleration conditions for larger test sets can help to increase the correctness of the statistical methodology and the parameter fittings, this also increases the test time and cost significantly. Hence, there is a need to find an optimization methodology that finds a small set of the test conditions which are tolerant to process variations among the larger collection of stress acceleration sets.

For the optimization to select the proper test conditions for the statistical analysis, first we make various test sets for each mechanism by varying temperature and supply voltages and build the failure rate distributions as illustrated in Fig. 4.14. Then, we build one more failure rate distribution with the same temperature and voltage, but with process variations. Finally, based on the two sets of failure distributions for the short group

(GTDDB and BTDDB) and the open group (SIV and EM), we run our numerical optimization algorithm based on Lagrange multipliers with power iterations [72].

SIV is more sensitive to temperature variations than EM (see Equations (3.3),(3.4)) and GTDDB is more variable as a result of voltage variations than BTDDB (see Equation (3.1),(3.2)). Hence, in this work, we create more test sets with different temperature acceleration conditions for SIV and EM and with different voltage acceleration conditions for GTDDB and BTDDB. Using the different acceleration conditions, we can cause the failure distribution to vary significantly with the different relative fractions of SIV and EM failures (λ) and the different relative fractions of GTDDB and BTDDB failures (γ).

We set 14 voltage acceleration test sets for each short group (SG1, SG2, SG3, and SG4) and 20 temperature acceleration test sets for each open group (OG1, OG2, and OG3). The temperature acceleration test sets for each short group are specified in Table 4.10, and the temperature acceleration test sets for each open group are presented in Table 4.11. Then, combining the acceleration test sets with short groups in Table 3.1 and open groups in Table 3.2, we create 56 test sets (14 voltage conditions x 4 short groups) and 60 test sets (20 temperature sets x 3 open groups) for failure analysis. We also combine both different voltage and temperature sets in the short and open groups at the same time for the experiments.

TABLE 4.10 VOLTAGE ACCELERATION CONDITIONS

Voltage							
Index	v=1	v=2	v=3	v=4	v=5	v=6	v=7
[V]	1.2	1.225	1.25	1.275	1.3	1.325	1.35
Index	v=8	v=9	v=10	v=11	v=12	v=13	v=14
[V]	1.375	1.4	1.425	1.45	1.475	1.5	1.525

TABLE 4.11 TEMPERATURE ACCELERATION CONDITIONS

Temperature										
Index	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10
[K]	270	275	280	285	290	295	300	305	310	315
Index	t=11	t=12	t=13	t=14	t=15	t=16	t=17	t=18	t=19	t=20
[K]	320	325	330	335	340	345	350	355	360	365

Then, the characteristic lifetimes for each short group with various voltage acceleration test sets, $\eta_{v,k}$, can be computed with

$$\eta_{v,k} = \left(\sum_j \sum_{i=1}^{231168} \frac{1}{\eta_{i,j,k,v}^{\beta}} \right)^{-1/\beta} \tag{4.13}$$

where k is an index for the short group (SG1-4) and v is the index for the voltage in ith cell and j indicates the short location within the short groups (see Table 3.1 and Table 4.10).

The characteristic lifetimes for each open group with temperature sets, $\eta_{t,m}$, can be computed with

$$\eta_{t,m} = \left(\sum_{l=1}^{231168} \frac{1}{\eta_{l,m,t}^{\beta}} \right)^{-1/\beta}. \tag{4.14}$$

where m is the index for the open group (OG1-3) and t is the index for the temperature in the lth cell (see Table 3.2 and Table 4.11).

The overall lifetime of the SRAM, η_{chip}, for 225.75 Kb SRAM cells for each mechanism is the solution of

$$1 = \sum_v \sum_k \left(\eta_{chip} / \eta_{v,k} \right)^{\beta} \text{ and } \sum_t \sum_m \left(\eta_{chip} / \eta_{t,m} \right)^{\beta}. \tag{4.15}$$

Given, $\eta_{v,k}$, k=1,…,4, v=1,…,14, for each short group in Table 3.1 and Table 4.10 and η_m, m=1,…,3, t=1,…,20 for each open group in Table 3.2 and Table 4.11, the probability

that the failure is located in the v, k^{th} group of locations and the t, m^{th} group of locations is

$$P_{v,k} = \left(\eta_{chip}/\eta_{v,k}\right)^{\beta} \text{ and } P_{t,m} = \left(\eta_{chip}/\eta_{t,m}\right)^{\beta}. \quad (4.16)$$

Overall, the observed relative frequency of the short groups, $P_{v,k,chip}$, is a function of the probabilities of GTDDB ($P_{v,k,GTDDB}$) and BTDDB ($P_{v,k,BTDDB}$), i.e.,

$$P_{v,k,chip} = \gamma P_{v,k,GTDDB} + (1-\gamma)P_{v,k,BTDDB}. \quad (4.17)$$

The relative frequency of the open fault sites in the SRAM chip, $P_{t,m,chip}$, is

$$P_{t,m,chip} = \lambda P_{t,m,SIV} + (1-\lambda)P_{t,m,EM}. \quad (4.18)$$

The parameter, γ and λ are computed by regression:

$$\gamma = \frac{\sum_v \sum_k \left(P_{v,k,GTDDB} - P_{v,k,BTDDB}\right)\left(P_{v,k,chip} - P_{v,k,BTDDB}\right)}{\sum_v \sum_k \left(P_{v,k,GTDDB} - P_{v,k,BTDDB}\right)^2} \quad (4.19)$$

and

$$\lambda = \frac{\sum_t \sum_m \left(P_{t,m,SIV} - P_{t,m,EM}\right)\left(P_{t,m,chip} - P_{t,m,EM}\right)}{\sum_t \sum_m \left(P_{t,m,SIV} - P_{t,m,EM}\right)^2}. \quad (4.20)$$

$P_{v,k,chip}$ and $P_{t,m,chip}$ are measured from the observed fraction of failures for each short and open group, using our BIST methodology with various acceleration conditions. $P_{v,k,GTDDB}, P_{v,k,BTDDB}, P_{t,m,SIV}$, and $P_{t,m,EM}$ are collected with the aging reliability simulator [4]-[10].

Based on the failure rate distributions with the various sets of acceleration conditions in Table 4.10 and 4.11, we need to remove the test sets which are vulnerable to process variations. To optimize the statistical analysis, we build the numerical optimization algorithm to reduce the test sets for the optimization.

First, we convert the equations (4.17) and (4.18) to matrix form for the numerical optimization as follows:

$$M_{GTDDB}\boldsymbol{x^T} + M_{BTDDB}(\boldsymbol{a^T}\text{-}\boldsymbol{x^T}) = P_{Chip\_short} \quad (4.21)$$

and

$$M_{SIV}\boldsymbol{y^T} + M_{EM}(\boldsymbol{a^T}\text{-}\boldsymbol{y^T}) = P_{Chip\_open}. \quad (4.22)$$

M_{GTDDB} and M_{BTDDB} are the 1 by (v x k) matrices that describe $P_{v,k,GTDDB}$ and $P_{v,k,BTDDB}$ in equation (4.17) as follow:

$$M_{GTDDB} = (P_{1,1,GTDDB} \quad P_{1,2,GTDDB} \ \ldots. P_{v,k-1,GTDDB} \ P_{v,k,GTDDB}) \quad (4.23)$$

and

$$M_{BTDDB} = (P_{1,1,GTDDB} \quad P_{1,2,GTDDB} \ \ldots. P_{v,k-1,GTDDB} \ P_{v,k,GTDDB}). \quad (4.24)$$

Similarly, M_{SIV} and M_{EM} are used to denote $P_{t,m,SIV}$ and $P_{t,m,EM}$ in equation (4.18) as:

$$M_{SIV} = (P_{1,1,SIV} \quad P_{1,2,SIV} \ \ldots. P_{t,m-1,SIV} \ P_{t,m,SIV}) \quad (4.25)$$

and

$$M_{EM} = (P_{1,1,EM} \quad P_{1,2,EM} \ \ldots. P_{t,m-1,EM} \ P_{t,m,EM}). \quad (4.26)$$

The $\boldsymbol{x}$ and $\boldsymbol{y}$ vectors are the solution set for the γ and λ and the $\boldsymbol{a}$ vector is the vector whose elements are '1'. P_{Chip_short} and P_{Chip_open} are the relative failure rate of each short and open groups from the BIST methodology. Based on the given M_{GTDDB}, M_{BTDDB}, M_{SIV}, and M_{EM} matrices from the reliability simulator and P_{Chip_short} and P_{Chip_open}, γ and λ can be computed.

The problem is when process variations are applied to the test data, P_{Chip_short} and P_{Chip_open}. When we match deviated P_{Chip_short} and P_{Chip_open}values with the failure distribution, the standard failure distribution map built with M_{GTDDB}, M_{BTDDB}, M_{SIV}, and M_{EM} can create errors in the $\boldsymbol{x}$ and $\boldsymbol{y}$ vectors. The deviation in γ for GTDDB vs. BTDDB and λ for SIV vs. EM can also lead to a false diagnosis or false parameter fittings. Hence, to solve the problem, it is necessary to exclude the stress acceleration sets that cause a significant error in $\boldsymbol{x}$ and $\boldsymbol{y}$ with process variations. To optimize the problem, we developed a numerical optimization algorithm based on Lagrange multipliers.

We apply +-10% random variations of threshold voltage and device/interconnect lengths in our simulator [4]-[10] and compute sets of $P_{Chip\_short\_PV}$ and $P_{Chip\_open\_PV}$ with equation (4.13)-(4.26). When we use the reference set of M_{GTDDB}, M_{BTDDB}, M_{SIV}, and M_{EM} for all other chips with process variations, equations (4.21) and (4.22) can be slightly changed with $\boldsymbol{x}'$ and $\boldsymbol{y}'$ with error terms induced by $P_{Chip\_short\_PV}$ and $P_{Chip\_open\_PV}$ as follows:

$$M_{GTDDB}\boldsymbol{x'^{T}} + M_{BTDDB}(\boldsymbol{a^{T}}\text{-}\boldsymbol{x'^{T}}) = P_{Chip\_short\_PV} \tag{4.27}$$

and

$$M_{SIV}\boldsymbol{y'^{T}} + M_{EM}(\boldsymbol{a^{T}}\text{-}\boldsymbol{y'^{T}}) = P_{Chip\_open\_PV}. \tag{4.28}$$

Then, we define transformation matrices to choose the acceleration sets to minimize errors of $|\boldsymbol{x^{T}}\text{-}\boldsymbol{x'^{T}}|$ and $|\boldsymbol{y^{T}}\text{-}\boldsymbol{y'^{T}}|$ for γ and λ. The transformation matrix, T_{short} and T_{open}, are used to choose several columns (test sets) in the P_{Chip_short} and P_{Chip_open} matrices.

For short groups for GTDDB and BTDDB, when we reduce the test sets using the T_{short} matrix, the equations (4.21) and (4.27) can be changed as

$$T_{short}M_{GTDDB}\boldsymbol{x^T} + T_{short}M_{BTDDB}(\boldsymbol{a^T}\text{-}\boldsymbol{x^T}) = T_{short}P_{Chip\_short} \quad (4.29)$$

and

$$T_{short}M_{GTDDB}\boldsymbol{x'^T} + T_{short}M_{BTDDB}(\boldsymbol{a^T}\text{-}\boldsymbol{x'^T}) = T_{short}P_{Chip\_short\_PV}. \quad (4.30)$$

By subtracting equation (4.29) from equation (4.30), we can derive an equation to express the error term, $\boldsymbol{e_{short}} = \boldsymbol{x^T}\text{-}\boldsymbol{x'^T}$, as

$$T_{short}(M_{GTDDB} - M_{BTDDB})\boldsymbol{e_{short}} = T_{short}(P_{Chip\_short} - P_{Chip\_short\_PV}). \quad (4.31)$$

Since we know $M_{GTDDB}, M_{BTDDB,}$ P_{Chip_short}, and $P_{Chip\_short\_PV}$ from the simulator and the BIST methodology, we just need to find the T_{short} matrix to minimize the $\boldsymbol{e_{short}}$ term. When we define $M_{short} = M_{GTDDB} - M_{BTDDB}$ and $P_{short} = P_{Chip\_short\_PV} - P_{Chip\_short}$, the $\boldsymbol{e_{short}}$ term in $T_{short}M_{short}\boldsymbol{e_{short}} = -T_{short}P_{short}$ can be minimized with the Lagrange multiplier equation with

$$\min < |\boldsymbol{e_{short}}|_2 + \mu\,(|T_{short}M_{short}\boldsymbol{e_{short}} + T_{short}P_{short}|_2)^2 > \quad (4.32)$$

To minimize equation (4.32), we vary the T_{short} matrix for several μ values until the error values converge with the power iteration method.

Similarly, the T_{open} matrix can be found for the optimization for the open faults due to SIV and EM with the equation,

$$\min < \left|\boldsymbol{e_{open}}\right|_2 + \mu\left(\left|T_{open}M_{open}\boldsymbol{e_{open}} + T_{open}P_{open}\right|_2\right)^2 >, \quad (4.33)$$

where $M_{open} = M_{SIV} - M_{EM}$, $P_{open} = P_{Chip\_open\_PV} - P_{Chip\_open}$, and $\boldsymbol{e_{open}} = \boldsymbol{y^T}\text{-}\boldsymbol{y'^T}$.

Fig. 4.17 presents the failure rate, $P_{v,k,chip}$, due to GTDDB and BTDDB, by varying the relative fraction of GTDDB and BTDDB failures, γ. In Fig. 4.17(a), the failure rate distribution contains 56 short test groups (14 voltage sets X 4 short groups in each voltage set) before the reduction of the test sets. Each short group for each voltage set contains four sub-groups (k=1..4) in Table 3.1. We can see that there are some significant differences of the failure rate for some voltage sets with process variations (see Fig. 4.17(a)). Hence, we run the optimization algorithm to choose 10 sets among the 56 sets. Our algorithm finds the T_{short} matrix with 1000 iterations to minimize the error value in equation (4.32). Fig. 4.17(b) presents both cases of the failure rate distribution with and without process variations after the reduction of test sets with the T_{short} matrix. Since we exclude the test sets which make a significant difference between the two graphs, the failure rate distributions for both cases in Fig. 4.17(b) are mostly the same. Our simulation results indicate that $|\boldsymbol{x^T}\text{-}\boldsymbol{x'^T}|_2$ for γ error without optimization is 0.8531 and $|\boldsymbol{x^T}\text{-}\boldsymbol{x'^T}|_2$ after the optimization is reduced to 0.0661. In addition to the benefit, the reduction of the stress acceleration experiments using optimization can lead to a significant reduction of test cost and effort.

Fig. 4.18 presents the failure rate, $P_{t,m,chip}$, due to SIV and EM with the relative fraction of SIV and EM failures, λ. Before the optimization in Fig. 4.18(a), the failure rate distribution contains 60 open test groups (20 temperature sets X 3 open group in each voltage set). Then, Fig. 4.18(b) presents the failure rate distribution after the optimization with the T_{open} matrix. $|\boldsymbol{y^T}\text{-}\boldsymbol{y'^T}|_2$ for λ error is reduced to 0.0941 from 0.1260 even with the significant reduction in the number of experimental sets.

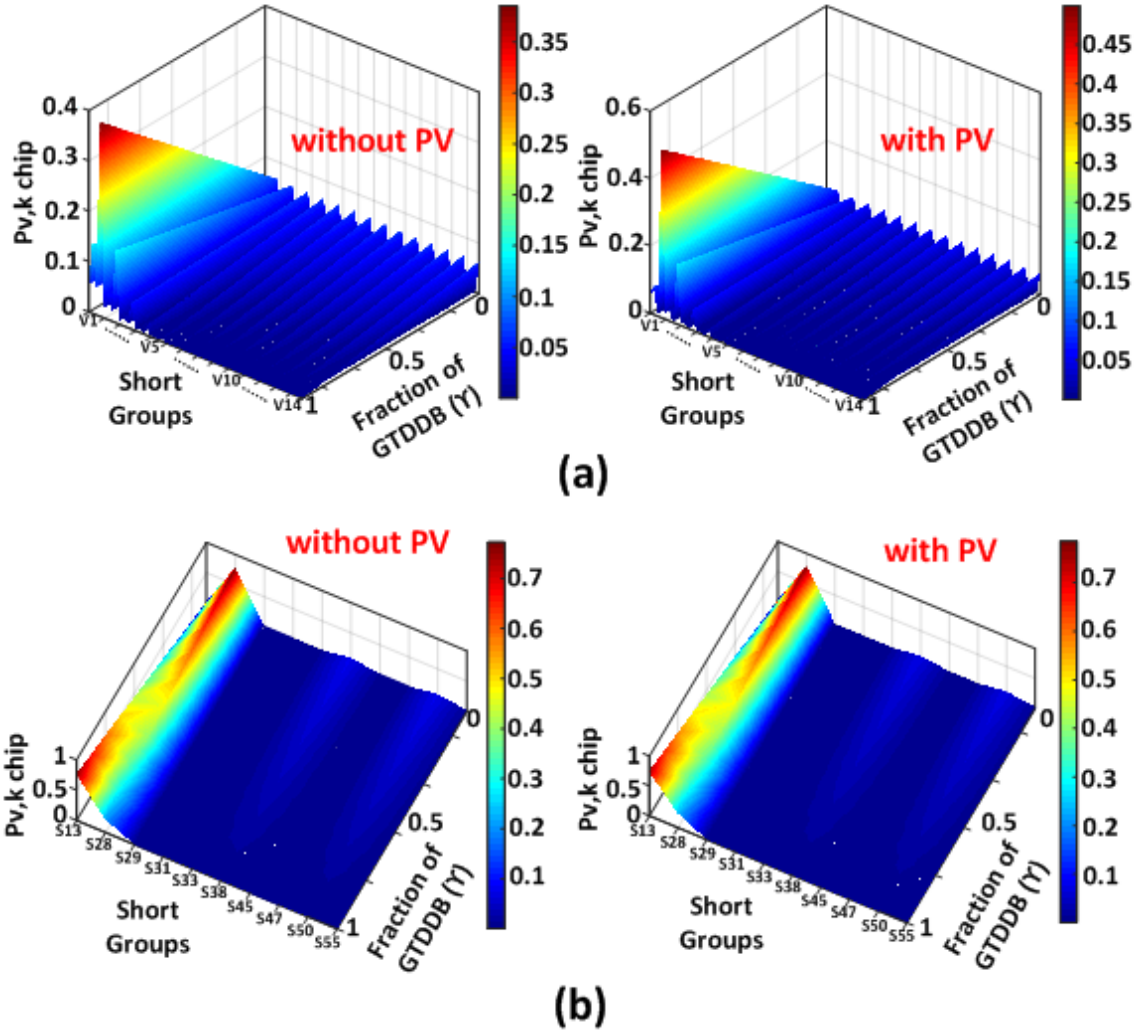

Figure 4.17 Failure rate distribution using a reliability simulator which determines the stress distribution of SRAM cells inside a microprocessor with general use scenario for GTDDB and BTDDB without process variation and with process variation (+- 10% threshold voltage and length variations) (a) before optimization, and (b) after optimization.

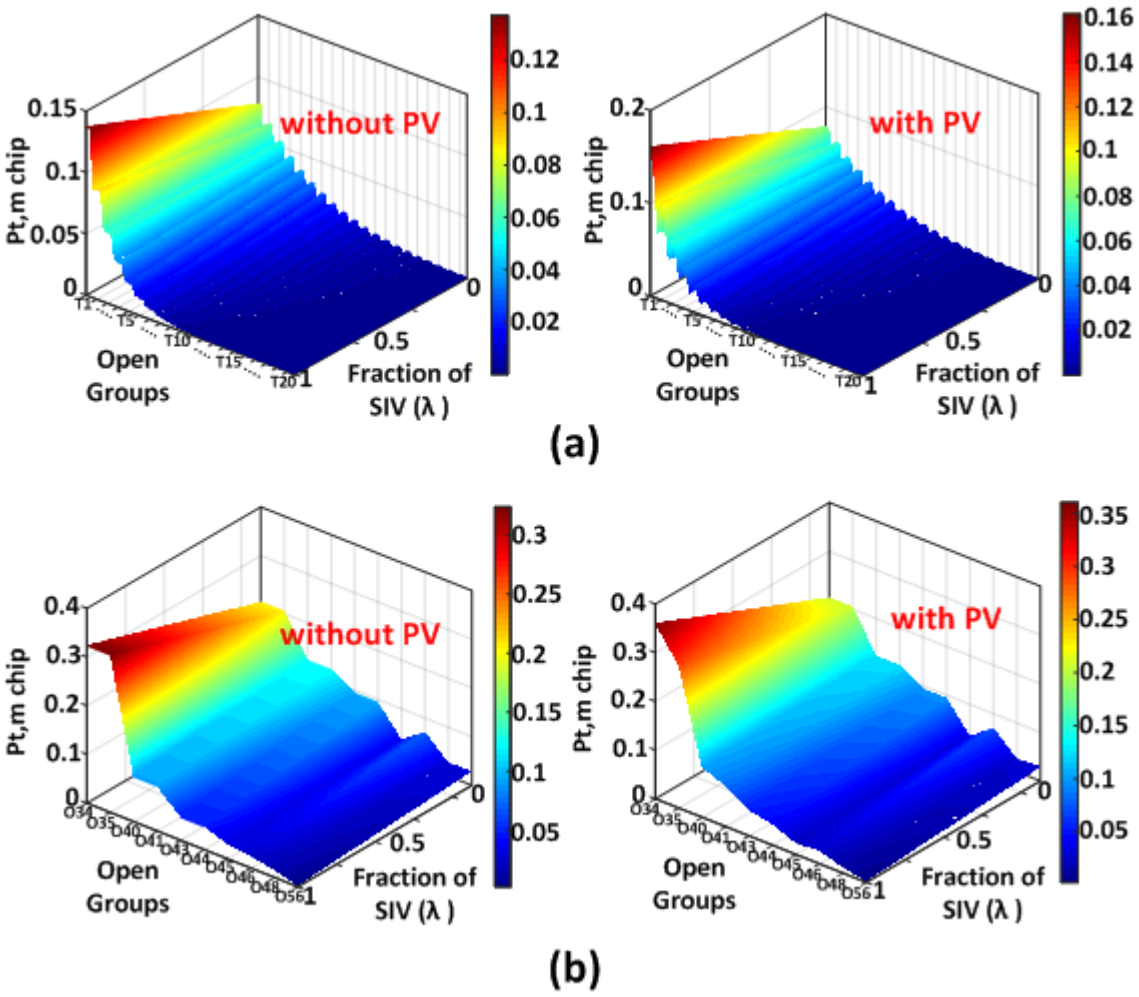

Figure 4.18 Failure rate distribution using a reliability simulator which determines the stress distribution of SRAM cells inside a microprocessor with gaming use scenario for SIV and EM without process variation and with process variation (+- 10% threshold voltage and length variations) (a) before optimization, and (b) after optimization.

Fig. 4.19 and Fig. 4.20 show how many iterations are needed to find the optimized T_{short} and T_{open} so that $|\boldsymbol{e_{short}}|_2$ and $|\boldsymbol{e_{open}}|_2$ converge within a fixed value of error, respectively. Our simulation data shows that the optimization algorithm can find the T_{short} and T_{open} matrices with just several hundred iterations.

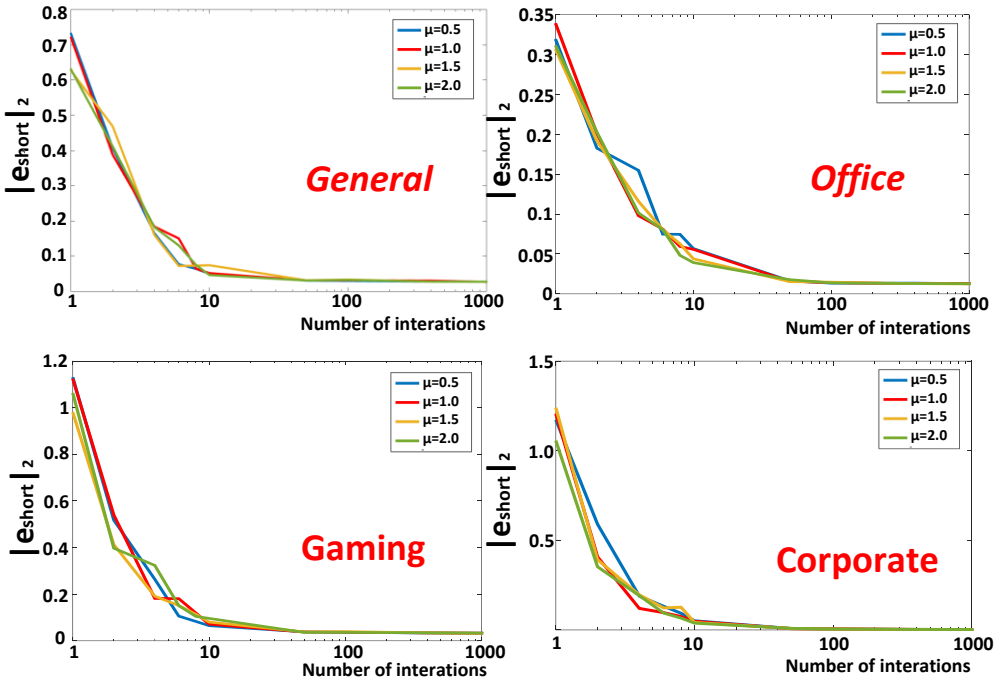

Figure 4.19 Number of iterations for the optimization of T_{short} vs. $|\boldsymbol{e_{short}}|_2 = |\boldsymbol{x^T}\text{-}\boldsymbol{x'^T}|_2$ values for GTDDB and BTDDB with different μ values for four usage scenarios.

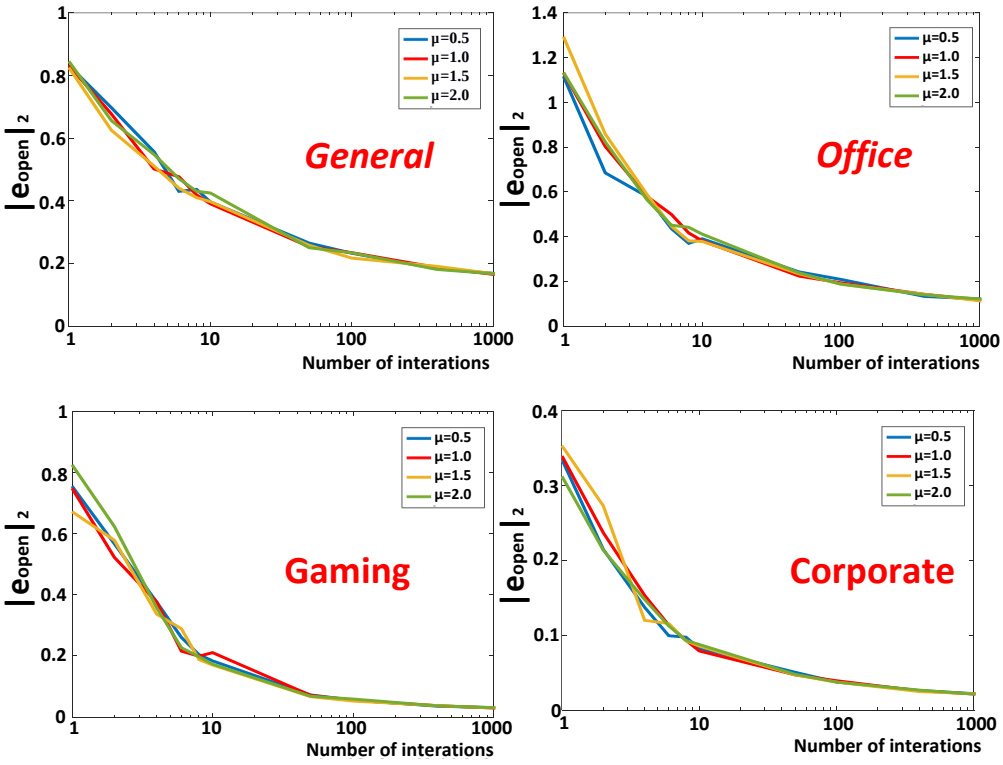

Figure 4.20 Number of iterations for the optimization of T_{open} vs. $|\boldsymbol{e_{open}}|_2 = |\boldsymbol{y^T}\text{-}\boldsymbol{y'^T}|_2$ values for SIV and EM with different μ values for four usage scenarios.

CHAPTER 5

DYNAMICALLY MONITORING SYSTEM HEALTH USING ON-CHIP CACHES AS A WEAROUT SENSOR

5.1 Estimation of Remaining Lifetime Using An SRAM System

5.1.1 Overview of Platform for Monitoring System Lifetime

Fig. 5.1 presents the platform to estimate the remaining lifetime of the processor using the SRAM array. The platform is based on the aging analysis framework presented in [4]-[10]. The implementation flow consists of four steps (see Fig. 5.1).

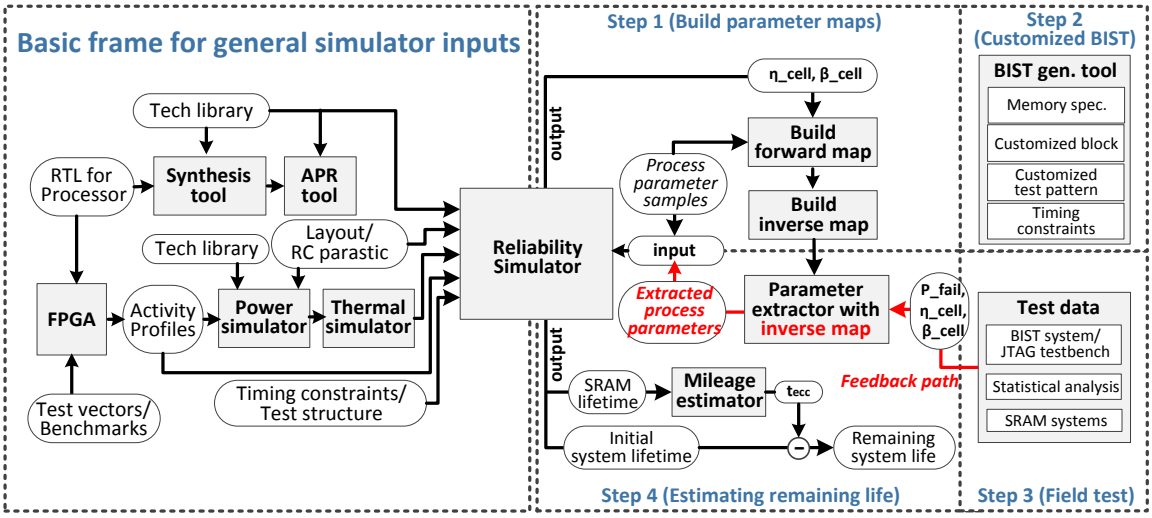

Figure 5.1 Overall platform for monitoring system lifetime [73],[74].

The first step in Fig. 5.1 starts to build Weibull parameter maps between process-level Weibull parameters and SRAM cell Weibull parameters. This is done with the reliability simulator in [4]-[10]. Based on an FPGA emulator, it creates the activity profile for the microprocessor. The extracted activity profiles combined with the layout are used to compute the power profile, which determines the temperature profile. Then, the vulnerable features extracted from the layout are combined with the electrical and

temperature profiles, from which feature lifetime is computed. The lifetime data are combined to estimate a lifetime distribution for each component. The Weibull parameters of the resulting distribution can be extracted. The parameter maps are then inverted so that wearout distribution parameters for each wearout mechanism are a function of SRAM cell wearout parameters for each wearout mechanism.

Step 2 in Fig. 5.1 generates the customized BIST netlist and joint test action group (JTAG) test benches for Table 4.1 using our reconfigurable platform based on a commercial BIST tool [62]. Next, the BIST methodology collects field test data.

In step 4, SRAM cell Weibull parameters are determined from the field test data in step 3. Then, process-level Weibull parameters can be extracted with the Weibull parameter maps and SRAM cell Weibull parameters. These maps are determined by the reliability simulator. The process-level Weibull parameters and the use scenarios are input into the microprocessor reliability simulator in step 4 to generate the remaining lifetime of the entire system at time zero. The usage of the circuit or a so-called mileage are estimated using the mileage estimator and the estimated lifetime from simulating by comparing the original and current remaining lifetime estimates. Finally, the remaining lifetime for the microprocessor is estimated by subtracting the mileage estimate from the time zero lifetime.

5.1.2 Step 1: Building the Weibull Parameter Maps

The observable parameters from the BIST system are Weibull parameters for the memory cells, not the Weibull parameters for the manufacturing process. Hence, we build the Weibull parameter maps between SRAM cell Weibull parameters and process-level Weibull parameters. The process-level Weibull parameters can be extracted from

measured SRAM cell Weibull parameters using the Weibull parameter maps. Step 1 builds the Weibull parameter maps for the extraction of the process-level parameters. Step 1 for the parameter maps consists of two sub-steps.

The sub-step 1 builds a forward map from process-level Weibull parameters to memory cell Weibull parameters. We sample process-level Weibull parameters and use the microprocessor reliability simulator presented in Fig. 5.1 to determine SRAM cell lifetime distributions. Specifically, we collect the corresponding SRAM cell lifetime (η_cell) and SRAM cell beta values (β_cell) for each wearout mechanism by varying values of the process-level Weibull parameters. Using the collected data, the forward map is built.

Sub-step 2 builds the corresponding inverse map, which indicates the estimated process-level parameters, given memory cell Weibull parameters. The inverse map is utilized in step 4 to extract the process-level Weibull parameters for the forward lifetime distribution prediction process.

Fig. 5.2 is the forward mapping from process parameters to memory cell Weibull parameters for two use scenarios for GTDDB. We varied A_{GTDDB} and β in equation (3.1) as the process-level Weibull parameters. Fig. 5.3 is the corresponding inverse map for the GTDDB mechanism for the same two test scenarios generated from the forward map.

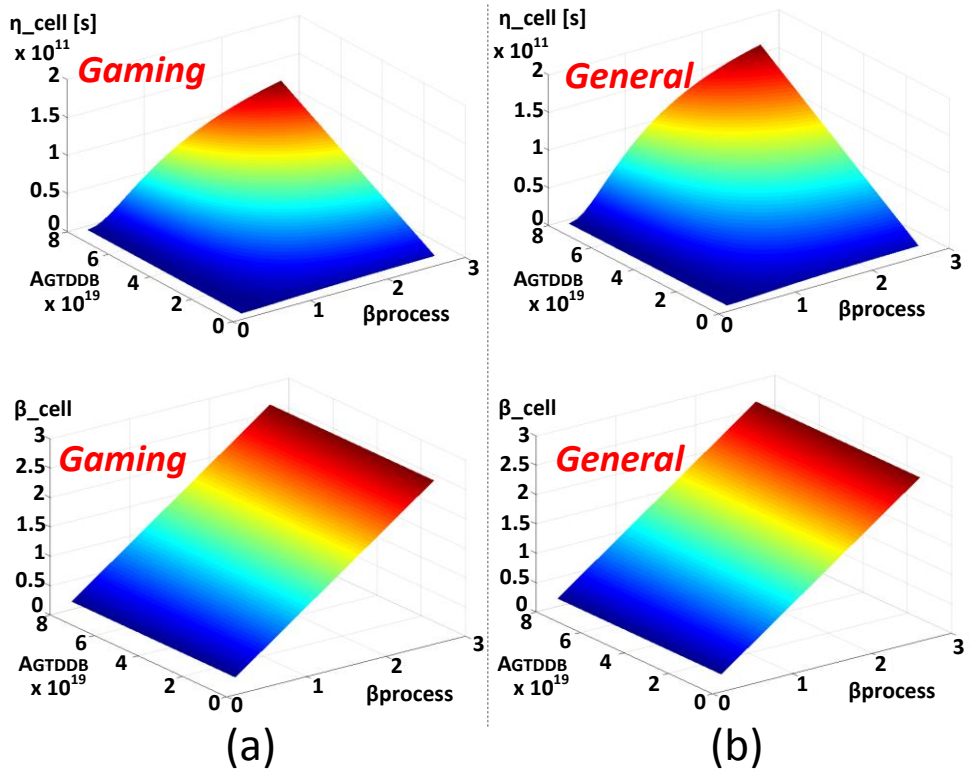

Figure 5.2 Forward mapping between process-level Weibull parameters and SRAM cell Weibull parameters for GTDDB, considering (a) gaming usage and (b) general usage.

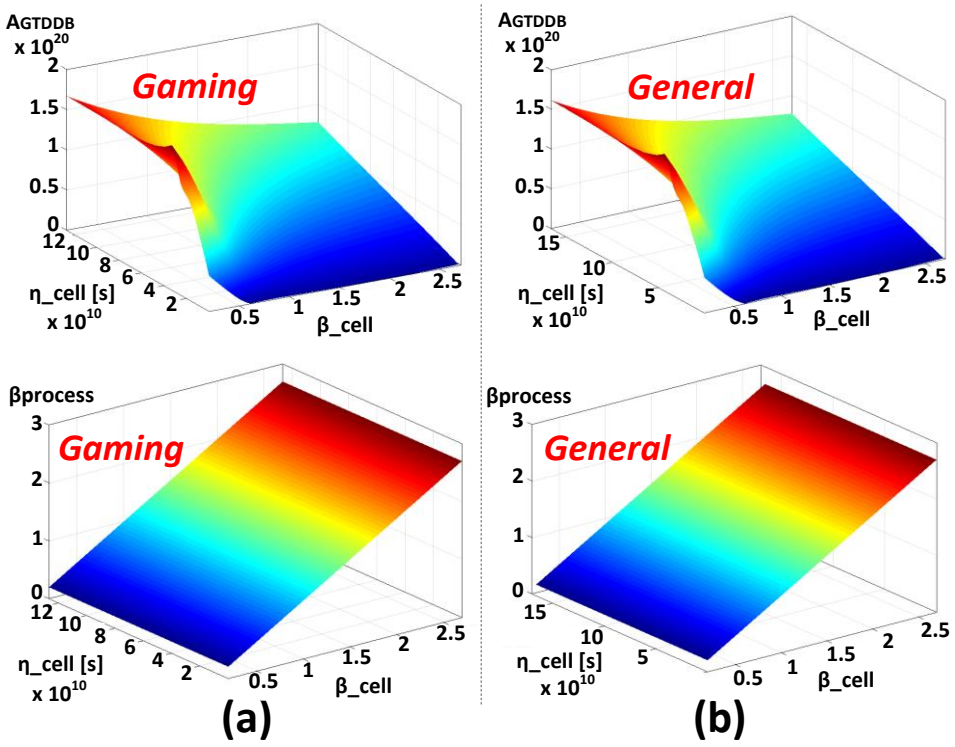

Figure 5.3 Inverse mapping between SRAM cell Weibull parameters for GTDDB and process-level Weibull parameters, considering (a) gaming usage and (b) general usage.

Fig. 5.4 presents a fitting methodology with the inverse map illustrated in Fig. 5.3. We assume that the BIST and statistical analysis proposed in Section 4 can extract the separate wearout distribution for each mechanism and workloads for the simulation are the similar with workloads used in the field. The measureable SRAM cell parameters, η_{cell} and β_{cell}, are estimated by combining the simulation data with the process-level parameters and test data from the BIST system. Then, the process-level parameters are updated and fitted with the inverse map and extracted SRAM cell parameters. Using updated process-level parameters, the reliability simulator for the SRAM system and the rest of logic parts in the processor can estimate the lifetime of processor.

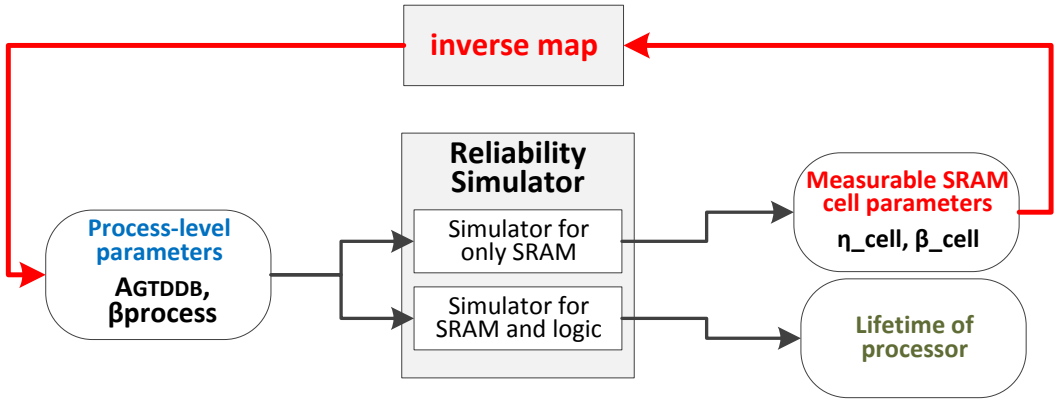

Figure 5.4 Fitting methodology with the inverse map.

5.1.3 Step 2: Reconfigurable Platform to Generate BIST Block and Test Bench

Caches are implemented in hierarchies of between 1 and 3 levels with various array sizes [60]. Step 2 in Fig. 5.1 generates the customized BIST system and test bench to implement the special BIST algorithm for wearout mechanisms in Table 4.1. We apply the BIST circuitry and algorithm to extract Weibull parameters, which can be used to estimate the lifetime of the full processor (see Fig. 5.1). To minimize the error for the estimation of the remaining lifetime, the ratio of area of the SRAM test array to the entire processor should be large enough. Hence, the customized BIST can test all designed

caches in the processor. Also, since cache sizes and operating frequencies are usually different, our BIST implementation platform should be flexible. Hence, a reconfigurable BIST implementation platform and flow are required to generate the customized BIST and test bench to test various types and sizes of memory systems in various processors (see Fig.5.5).

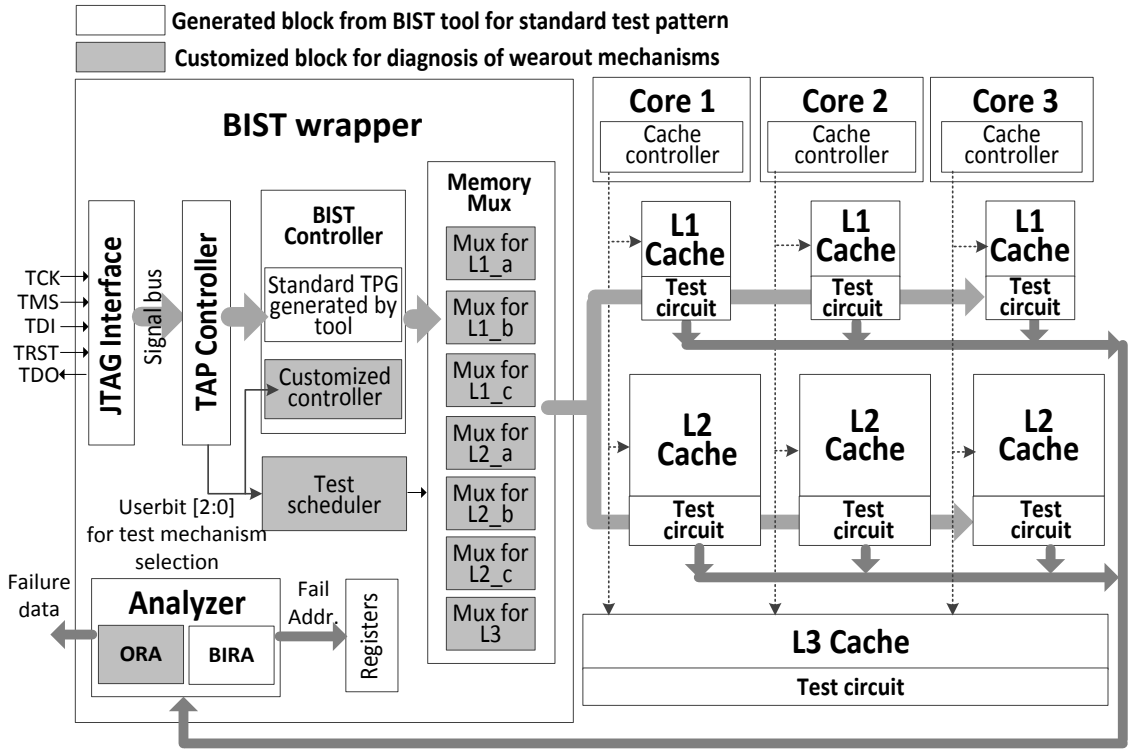

Figure 5.5 Reconfigurable platform to generate the customized BIST for wearout mechanisms for the various sizes of caches using a commercial tool [62].

Fig. 5.5 presents the BIST system architecture for each mechanism based on the BIST algorithm for the single SRAM system presented in Table 4.1. The system is a hybrid platform, combining the BIST part from the commercial BIST generation tool [62] and the customized part for wearout mechanisms. The hybrid platform based on the implementation flow from the commercial BIST tool makes the BIST system and JTAG

test bench highly reconfigurable for different process technologies, cache sizes and memory architectures.

In the BIST wrapper in Fig. 5.5, the standard test pattern generator (TPG) in the BIST controller, the built-in repair analysis (BIRA), test access port (TAP) controller, and the JTAG interface are generated from the commercial tool [62]. Based on the basic components, we have designed a customized controller in the BIST controller, a test scheduler, a customized output response analyzer (ORA), and mux systems in the BIST system wrapper to implement the special algorithms for each wearout mechanism in Table 4.1.

The BIST controller contains the standard test pattern generator (TPG) generated by the commercial BIST tool and the customized controller for wearout. The standard TPG is used to create the test pattern for addresses and read/write data for the standard test algorithms, such as the March algorithm before shipping the chip from the manufacturer [75]. The customized controller contains the register-type circuits to generate our special test patterns in Table 4.1. The customized output response analyzer (ORA) is embedded into the Analyzer with the BIRA module generated by the BIST generation tool. Using the results from the test circuit, the customized logic in the ORA determines the wearout failures with the algorithm in Table 4.1 (see Fig. 5.5). The standard BIRA module from the commercial tool is used when there is a need to execute standard test algorithms.

Also, since address sizes and input and output (I/O) widths are not the same for all different types of caches, there is a need to design mux systems in the BIST system wrapper between the BIST controller and each test memory to match the sizes of address

and I/O widths (see Fig. 5.5). The test scheduler in Fig. 5.5 uses the userbit registers in the TAP controller to set the test schedule for each test step presented in Table 4.1. The userbit is set in the BIST generation tool when the BIST netlist and testbench are generated (see Fig. 5.5).

Fig. 5.6 is the revised BIST implementation flow based on the flow from the commercial BIST tool to make the customized BIST system reconfigurable. As the tool inputs, we include the behavioral models of the top modules in the BIST system wrapper, memory definitions, and userbit definition for test algorithm selection. The behavioral models for our customized logic for the customized controller, test scheduler, and mux systems are included in the BIST tool input set. With BIST tool inputs, the commercial BIST implementation tool flows start. For step 1 and step 2 in Fig. 5.6, the tool assembles the BIST modules and generates the behavioral models for each top module for the JTAG interface, the TAP controller, the standard TPG, and BIRA.

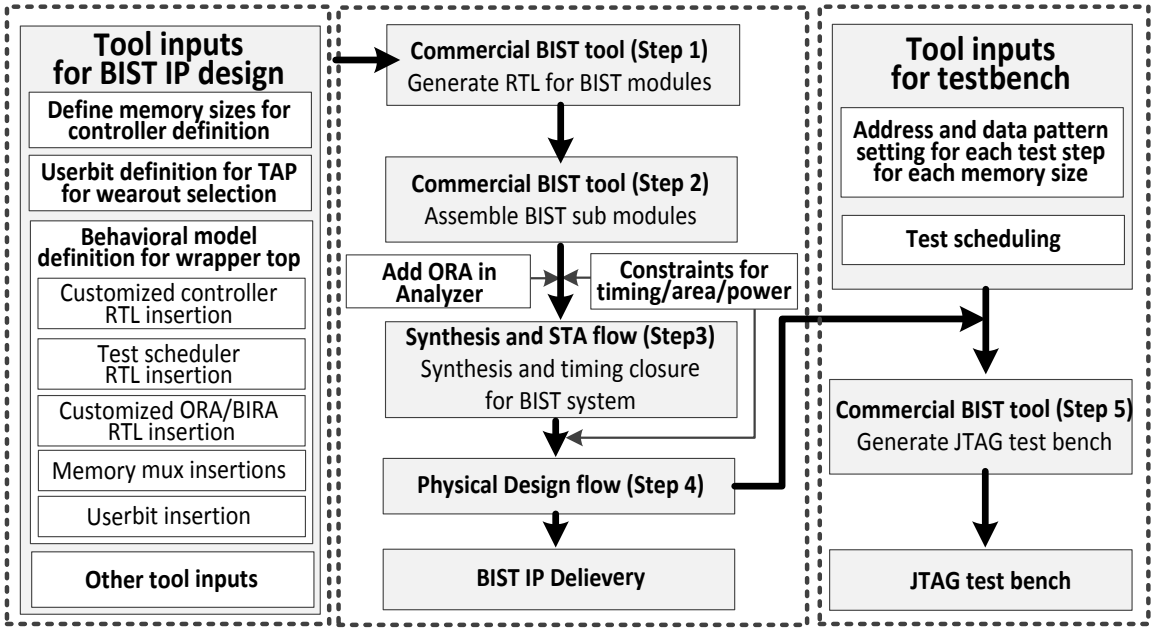

Figure 5.6 BIST implementation flow for wearout mechanisms based on the commercial tool from Mentor Graphics.

When the behavioral models for submodules are generated, we insert the behavioral model of ORA in the Analyzer block. Since the ORA module is connected to the submodules of BIRA generated by step 2, it can be added between step 2 and step 3. Then, step 3 and step 4 do synthesis and physical design with the behavioral models for the top and sub-modules with the design constraints for each application and process technology.

To generate the test bench as a JTAG standard for the special algorithm in Table 4.1, the BIST tool flow can be used (see Fig. 5.6). As the tool inputs for the generation of the test bench, we set the test pattern for addresses and data for each test step for each memory size in Table 4.1. With the specific inputs and the generated BIST intellectual property (IP), the test pattern and algorithm in Table 4.1 is converted to a JTAG standard through step 5 in Fig. 5.6.

5.1.4 Step 3 and Step 4: Process-Level Weibull Parameter Extraction and Estimation of Remaining Life

The diagnosis methodology outlined in Section 4 is utilized to track the failure of SRAM cells for each mechanism. Each of these wearout failures is diagnosed with on-chip BIST system and the JTAG test bench from step 2 in Fig. 5.1 to determine the location of the fault. If there is sufficient data, then the number of faults due to each mechanism is also determined, i.e. distinguishing BTDDB vs. GTDDB and EM vs. SIV using the failure distribution in Fig. 4.14. The next step is to estimate the wearout model parameters for each mechanism.

Specifically, when we track the failure of SRAM cells for each wearout mechanism, let's suppose that the time to failure of each cell in a memory system is modeled with a Weibull distribution, with two parameters, the characteristic lifetime, η,

and the shape parameter, β. If there are N memory cells in an SRAM array, then the first failure is associated with probability $1/2N$, the second failure is associated with probability $3/2N$, etc. When we record the time to failure, t_1 for the first failure, t_2 for the second fail bit, then with several failures, we can solve for the Weibull distribution parameters for the time-to-failure of the SRAM cells. Namely, if we plot the ordered pair $(\ln(t_1), \ln(-\ln\left(1-\frac{1}{2N}\right))$, $(\ln(t_2), \ln(-\ln\left(1-\frac{3}{2N}\right))$, etc., the x-intercept is $\ln(\eta)$ and the slope is β, as shown in Fig. 5.7. Hence, we can estimate the Weibull parameters of the time-to-failure of all SRAM cells from just determining the time-to-failure of several sample cells in the SRAM. Note that these are the Weibull parameters for the memory cells. Hence, the Weibull parameters for the SRAM cells should be converted to Weibull parameters for the manufacturing process wearout distributions through the parameter mapping with the inverse map presented in Fig. 5.3.

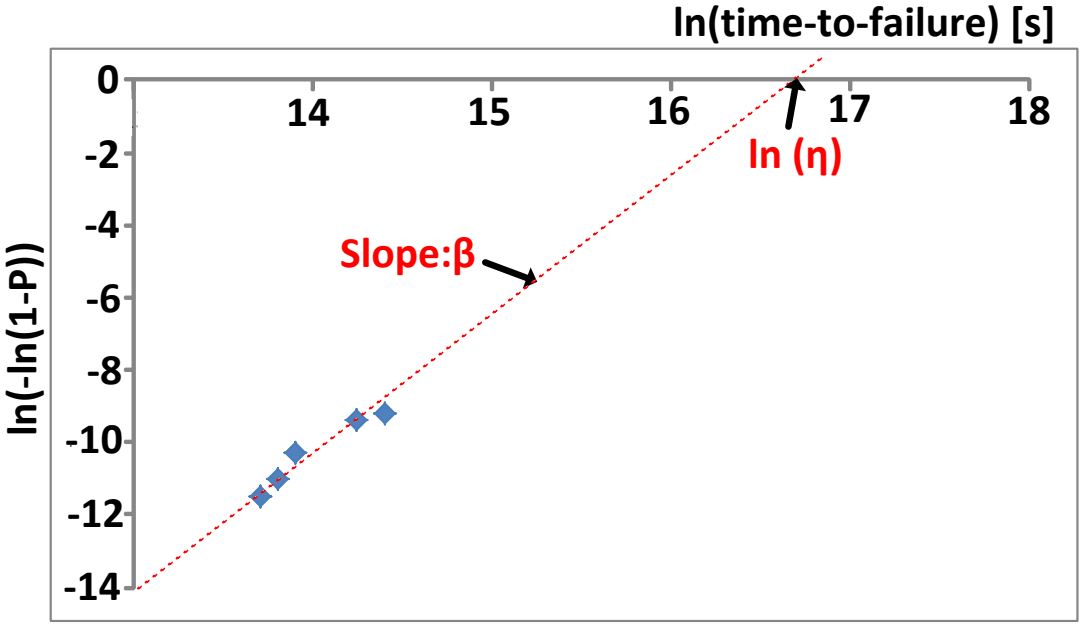

Figure 5.7 Extraction of Weibull parameters for the failure rate of memory cells by counting the number of failed memory cells.

The remaining lifetime of the entire system, η_{remain_system}, is estimated with the equation (5.1),

$$\eta_{remain\_system} = \eta_{initial\_system} - t_{ecc}, \quad (5.1)$$

$\eta_{initial\_system}$ is the initial lifetime of the system and t_{ecc} indicates the usage of the device which is the time for the memory bit failures when *n* ECC failures have been observed.

The initial lifetime of the system is estimated using the reliability simulator with inputs that include the extracted Weibull parameters from the inverse mapping. The lifetime of the entire processor takes into account both the logic and the memory blocks, with single bit error correction in the memory blocks to improve memory lifetime. The usage of the device, t_{ecc}, is estimated using the mileage estimator with the memory lifetime from the reliability simulator .

The mileage estimator in Fig. 5.1 estimates t_{ecc} which is used as the time-monitoring parameter with the following equation,

$$t_{ecc} = \eta_{cell} e^{\frac{\ln(-\ln(1-P_{ecc}))}{\beta_{cell}}}, \quad (5.2)$$

which is the solution of the equations [4]:

$$\ln(-\ln(1-P_{ecc})) = \beta_{cell}(\ln(t_{ecc}) - \ln(\eta_{cell})), \quad (5.3)$$

$$P_{ecc} = (1+2n)/2N, \quad (5.4)$$

where *n* is an observed number of ECC failures, η_{cell} is the cell lifetime, β_{cell} is memory cell shape parameter, and P_{ecc} is the probability of memory bit failure. *N* is the total number of SRAM cells, which are used for the test vehicle [4]. η_{cell} and β_{cell} data for SRAM systems are calibrated with the field data and data provided by the reliability simulator in Fig. 5.1.

Fig. 5.8 shows the ratio between the failure time (when 50% of samples have failed) for the entire system and when at least five ECC failures, t_{ecc}, have been observed for different mechanisms. Since the ratios are not constant, this graph presents that it is necessary to identify the cause of failure in order to correctly estimate the remaining lifetime from ECC failures. The diagnosis methodology in Section 4 provides the required data.

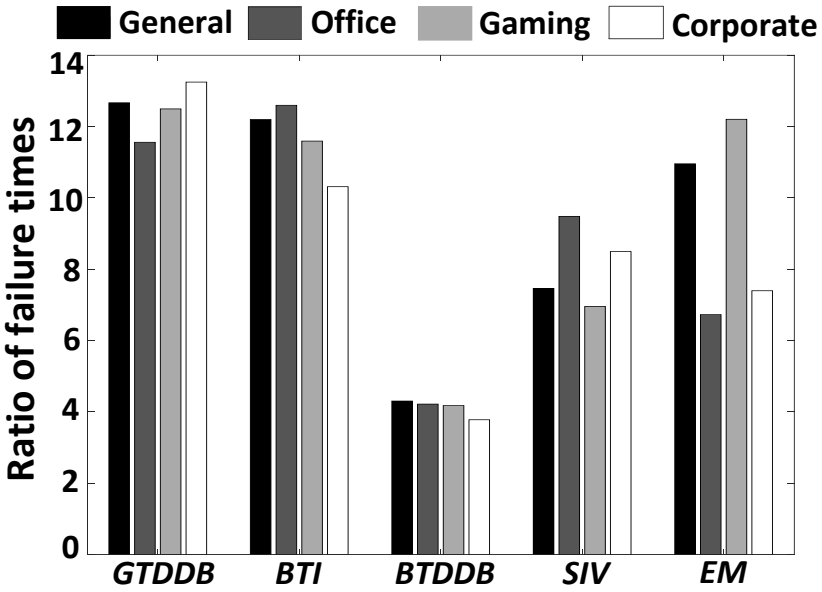

Figure 5.8 Simulation results on the ratio (γ) between the time for system failure for the LEON3 processor and the first five ECC failures for the embedded memory.

Fig. 5.9 presents the expected number of ECC failures, *n*, prior to the failure of a memory block for memories of different array sizes. To compute Fig. 5.9, the total SRAM array size is the product of the number of words, N_{word}, the number of columns, N_{col}, and the number of rows, N_{row}. Let assume that F_{bit} is the probability of failure of a bit. Then the probability of failure of a word is estimated with the binomial distribution:

$$F_{word} = 1 - (1 - F_{bit})^{N_{word}} - N_{word} F_{bit} (1 - F_{bit})^{N_{word}-1}. \quad (5.5)$$

The yield of the memory system is

$$Y = (1 - F_{word})^{N_{row}N_{col}}. \quad (5.6)$$

Using these equations, F_{bit} is estimated such that $Y = 0.5$, i.e. 50% of the SRAMs have failed. Again, using the binomial distribution, the total number of failed memory bits is

$$X = N_{row}N_{col}N_{word}F_{bit}(1 - F_{bit})^{N_{word}-1}. \quad (5.7)$$

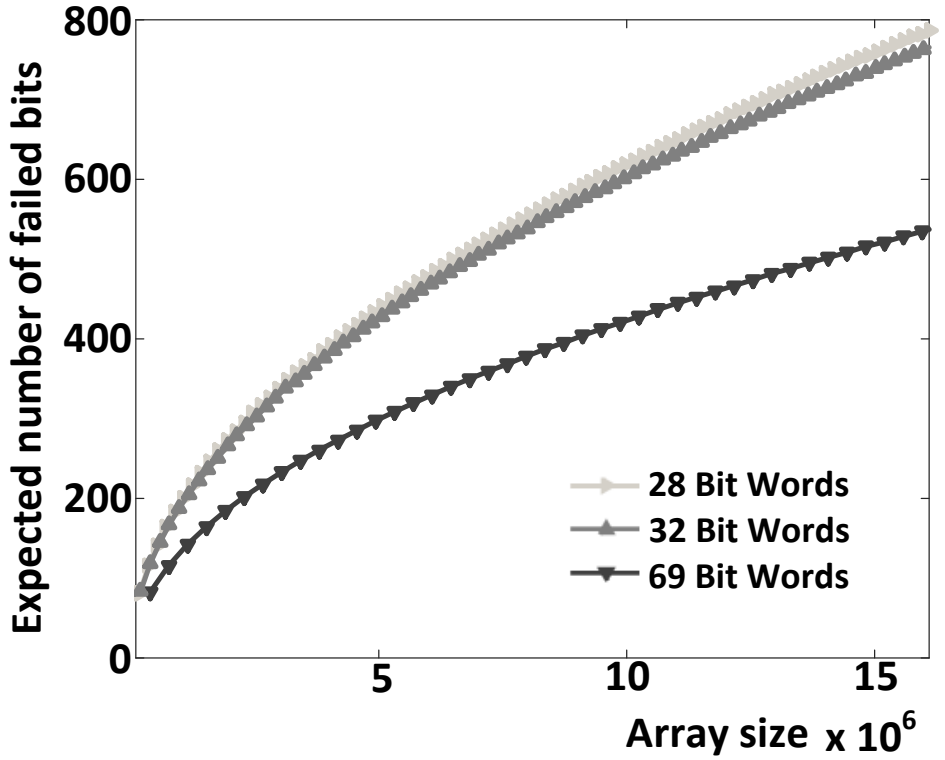

Figure 5.9 Simulation results for the expected number of ECC failures prior to the failure of an SRAM system.

Several ECC failures are available to estimate the lifetime of the processor. The LEON3 processor consists of 226K bits of memory when all of the embedded memories are combined, all of which contain ECCs. This is a small processor, and even for this processor, there are more than 88 failed bits that can provide an estimate of the system lifetime prior to the failure of the processor.

Fig. 5.10 is a simulation result to present the correlation between the number of bit failures (n) and the estimated remaining life of the entire system (η_{remain_system}) for

BTDDB, SIV, EM, GTDDB, and BTI for four usage scenarios. The number of failed bits correlates closely with the remaining life of the processor. Then, by tracking the ECC failure log, the remaining lifetime of the system can be estimated. The initial lifetimes for the summation of all the mechanisms are 12.53 years for general usage, 24.10 years for office usage, 10.64 years for gaming usage, and 14.75 years for corporate usage.

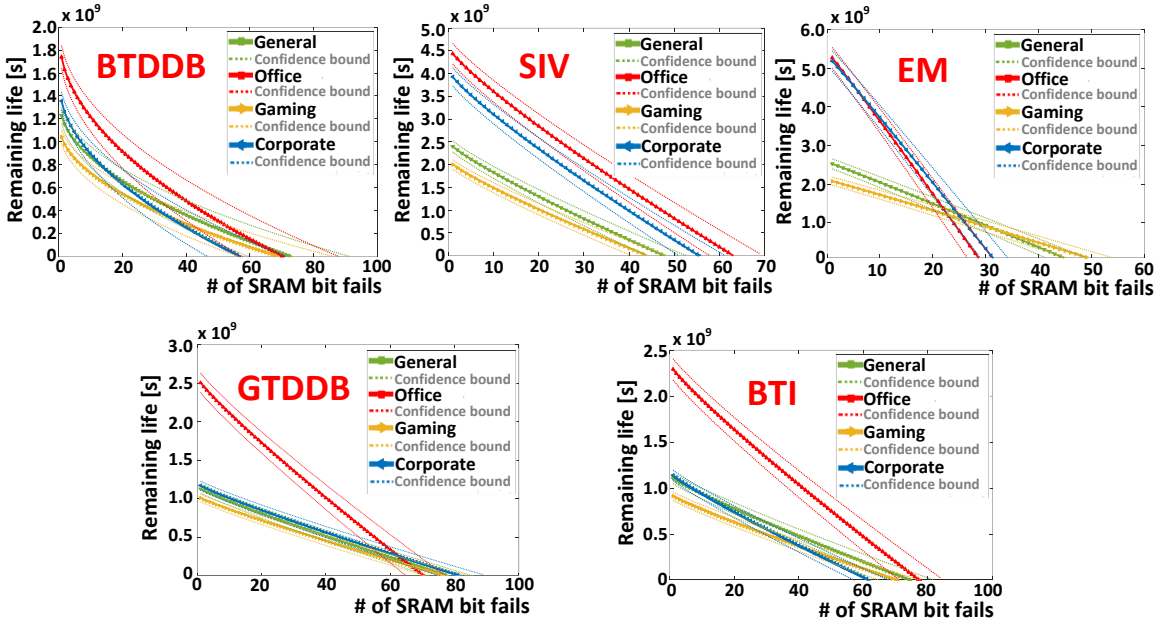

Figure 5.10 Simulation results for remaining lifetime vs. the number of failed bits for the LEON3 processor for various use conditions for BTDDB, SIV, EM, GTDDB, and BTI mechanisms.

During recording and plotting of the memory time-to-failures as presented in Fig. 5.7, there can be measurement deviations for different sets of chips. This is because there can be diagnosis errors using the BIST system or there might be process variations between different chips. Since the error can have an impact on the remaining life estimation results, we have to set the appropriate confidence bounds on the remaining lifetime result for each mechanism. First of all, note that Fig. 5.10 estimates η_{remain_system} in equation (5.1). The 90% confidence bounds on the true lifetime range

from 5% to three times the characteristic lifetime for the system. In addition, the regression used to compute η and β in Fig. 5.10, can also be used to estimate the standard error variance of $\ln(t_{ecc})$ in equation (5.3). This error is due to errors and variation in the data on time-to-failure of memory bits. This determines errors in η_{remain_system} in Fig. 5.10. The confidence bounds increase as a function of time since the error term is multiplied by an exponential, which is increasing with the increasing number of failing bits. The confidence bounds are presented in Fig. 5.10.

5.2 Statistical Failure Analysis For SRAM Failures due to GTDDB vs. BTDDB and EM vs. SIV.

5.2.1 Statistical Analysis for the Wearout Parameter Extractions

For short groups in Table 3.1 and open groups in Table 3.2, the cause of a fault cannot be determined using only electrical test because the failure signatures are the same exactly. Note that both the EM and SIV can induce resistive opens and both GTDDB and BTDDB cause resistive shorts in the same locations in an SRAM array. When the memory test is conduced to extract wearout parameters as shown in Fig. 5.7, the statistical analysis presented in section IV can distinguish GTDDB vs. BTDDB and EM vs. SIV. For wearout parameter extraction, sufficient failed bit samples should be collected before the extraction of η_{cell} and β_{cell} presented in Fig. 5.7. Then, the statistical failure analysis is conducted to estimate the fraction of each mechanism. Fig. 4.14 presents the failure rate distribution to distinguish the short and open groups.

5.2.2 Statistical Analysis for Failed Bits from ECCs

If we use the SRAM fail bits tracked by ECC as the time indicator to estimate the remaining lifetime, there is also a need to diagnose the cause of failures for the wearout

mechanisms. When we build the remaining lifetime graph in Fig. 5.10, the memory failure times for each mechanism are determined in Step 4 in Fig. 5.1. Fig. 5.11(a) shows the simulation results for the ratio of the number of GTDDB failures to the number of detected short faults for four different usage scenarios. The fraction of GTDDB, γ, is not the same in all the time intervals. The main cause of the variation in slope is that β for GTDDB and BTDDB is not the same. Fig. 5.11(b) for open faults also presents the fraction of SIV failures, λ, which is different at different time points. Hence, there is a need to utilize different γ and λ values for different time intervals instead of just one value for each group.

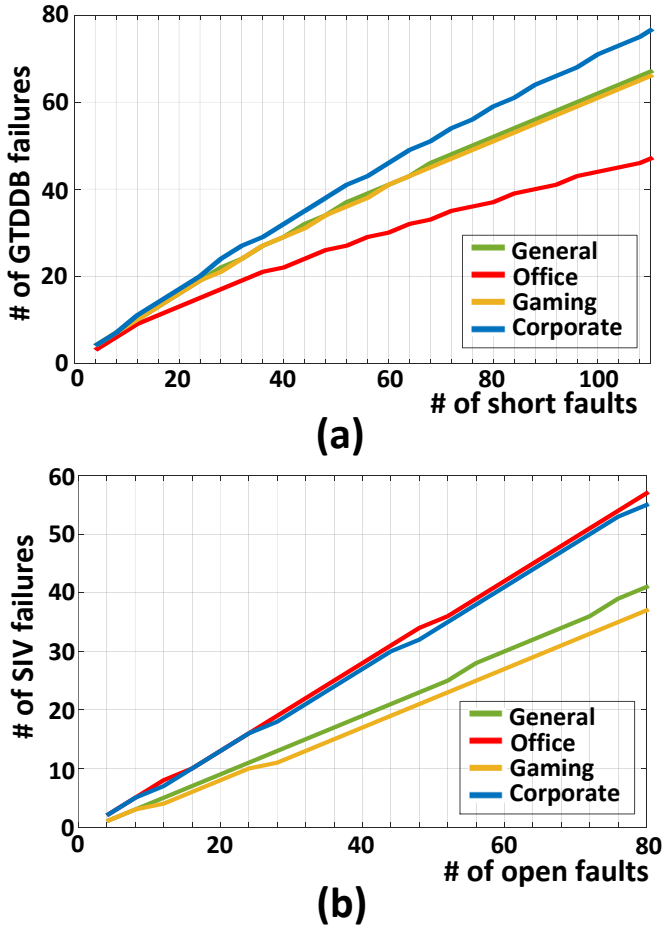

Figure 5.11 Simulation results (a) for the ratio of a number of GTDDB failures to a number of detected short faults in an SRAM array and (b) for the ratio of a number of SIV failures to a number of detected open faults in an SRAM array.

The γ and λ values for each time interval should be initially estimated from calibrated simulation using a reference chip. The reference chip is tested to build the remaining lifetime graph, which can then be used for different chip sets. When a different chip is monitored with ECC tracking and the remaining lifetime map, the γ and λ values for the reference chips are used to estimate the assignment of the cause of ECC fail bits from other chips.

Fig. 5.12 shows the experimental results for the statistical failure analysis methodology. We collected γ values for each of the four short faults and λ values for each of the four open faults from the calibrated simulation data. Then, we use the collected γ and λ for another chip with a different use scenario in Fig. 1.2. Then ECC and BIST system track the four short faults or the four open faults. Based on the data, we can diagnose the cause of failures using the corresponding γ and λ for each time interval and assign the time stamps of the failures accordingly. Then, we plot the remaining lifetime graph with the sampled memory time-to failures for the chip using γ and λ from calibrated simulation data. Fig. 5.12(a) shows a comparison between the remaining lifetime graph with the statistical analysis and the real simulation results for the BTTDB mechanism for the test chip. The gap between graphs shows the error due to the statistical diagnosis methodology to distinguish BTDDB vs. GTDDB. Fig. 5.12(b) presents the error between the remaining lifetime graph with the statistical analysis and the real simulation results for the EM mechanism.

Fig. 5.13 shows the average error for the remaining lifetime estimation due to statistical failure analysis for the GTDDB, BTDDB, EM, and SIV mechanisms for different sizes of the sampling group for the collection of γ and λ. The error from the

statistical methodology is less for a smaller sampling group size. Also, the average errors from the statistical methodology are under 7% for all cases using the smallest sample size, 4.

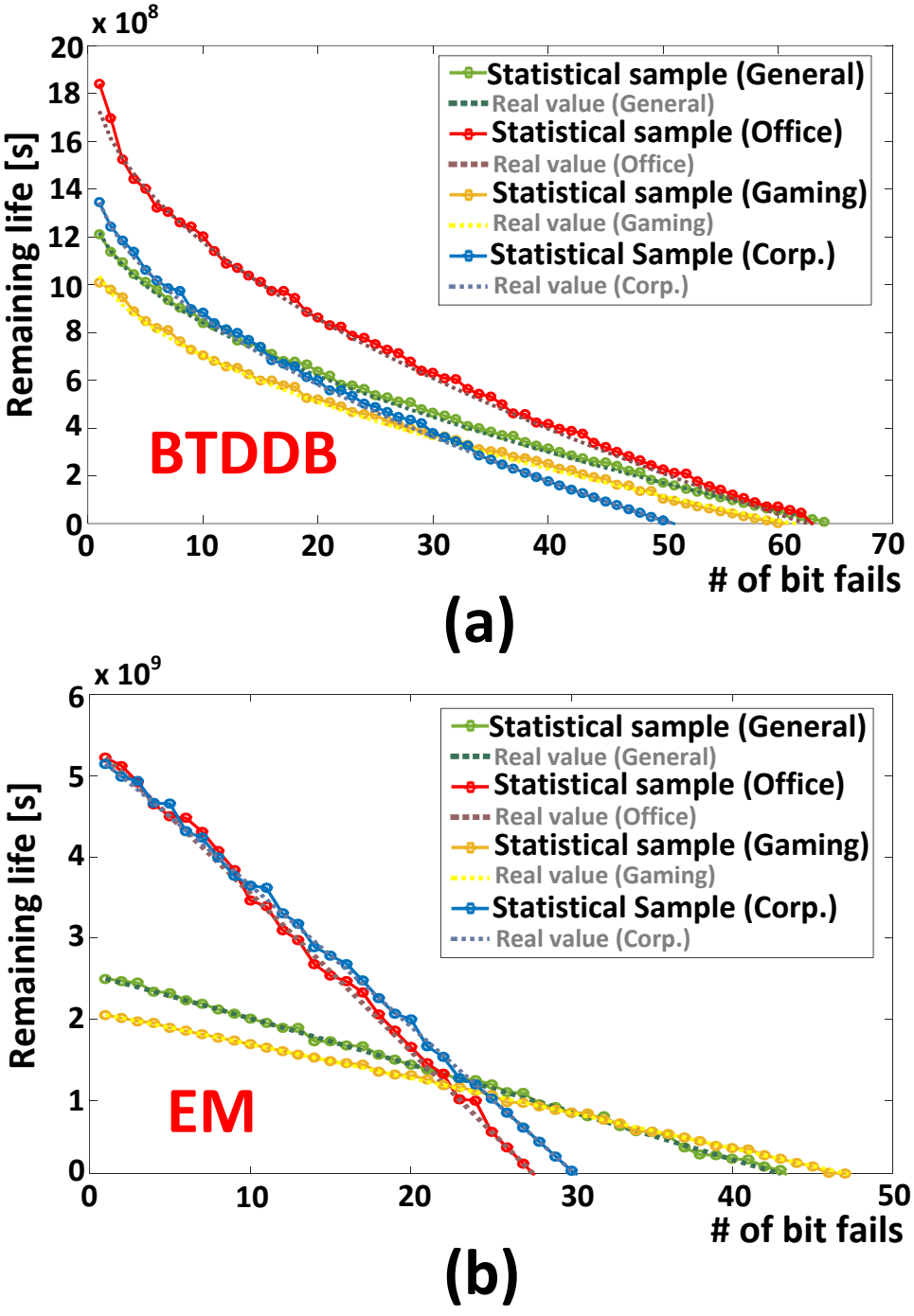

Figure 5.12 The remaining lifetime estimation from statistical failure analysis vs. the true result from simulations for (a) BTDDB mechanism and (b) for EM mechanism.

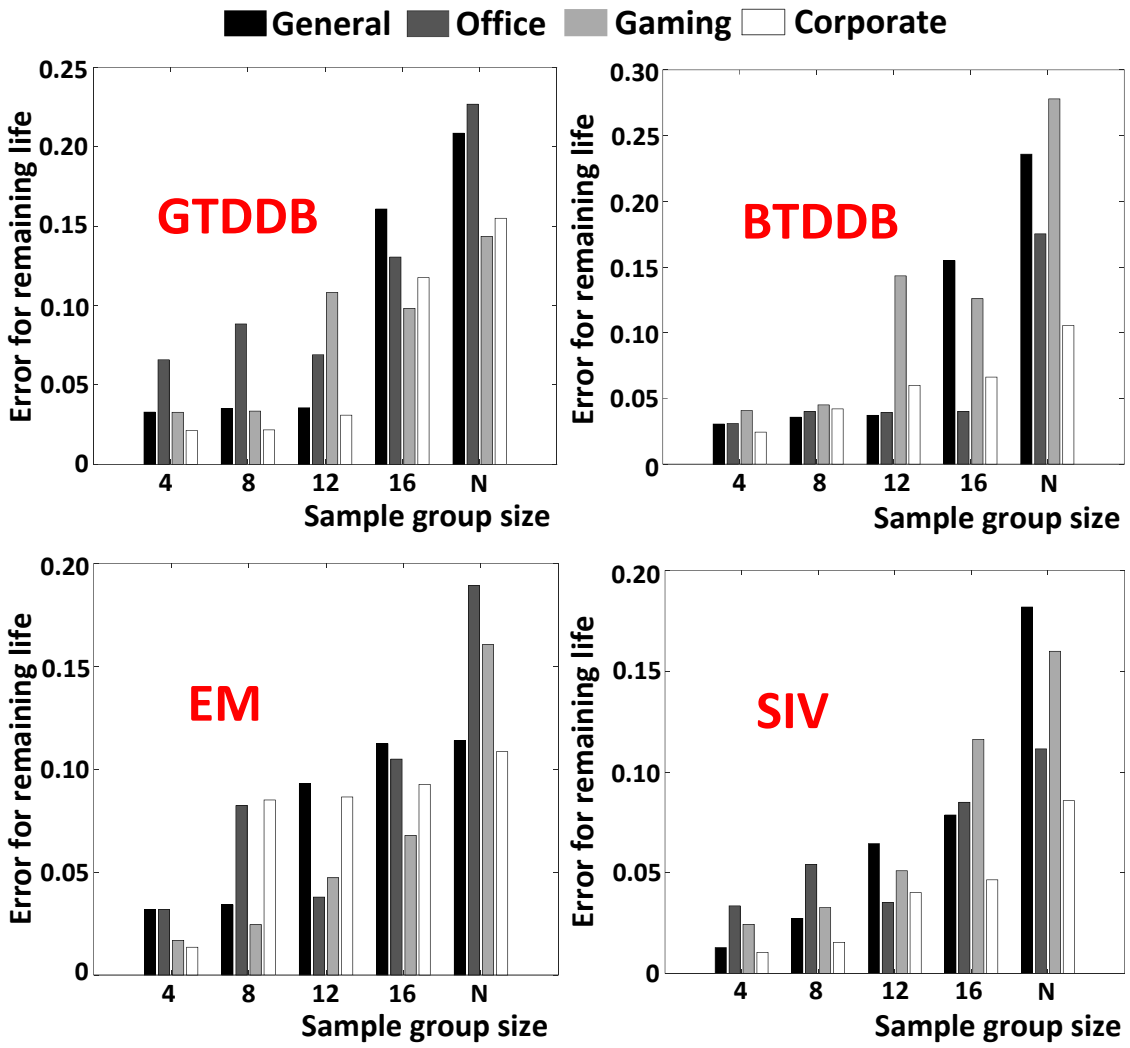

Figure 5.13 The average error for the estimation of remaining lifetime (from the initial time point when 10% lifetime remains) for different sampling group sizes for the GTDDB, BTDDB, EM, and SIV mechanisms.

5.3 Case Study: Impact of Design and Memory Parameters on the Simulation Results

For the estimation of the remaining lifetime for the processor, several quantifiable parameters, including memory array size, memory supply voltage, temperature, and process parameter variations can have an impact on the simulation results. Hence, the appropriate calibration procedures can be conducted for the simulation flow for each

critical parameter. In this section, we present a case study for the impact of these parameters on the simulation results of the remaining lifetime.

5.3.1 Impact of Memory Array Size on Estimation Result

The characteristic lifetimes of each mechanism for the entire cache cluster, η_{SRAM}, are determined by solving for the lifetime of each cell, η_j with [68]:

$$1 = \sum_{i=1}^{n=N} P_i \tag{5.8}$$

where

$$P_i = (\eta_{SRAM}/\eta_i)^{\beta_i} \ , \tag{5.9}$$

P_i is the probability of failure of ith cell and *N* is the number of memory cells in the entire processor. For a single mechanism, β_i is usually assumed to be constant. With this assumption that all cells are identical, a closed form solution is derived for the lifetime of the entire memory systems

$$\eta_{SRAM} = \eta_i/N^{1/\beta}. \tag{5.10}$$

If the total number of memory cells, *N*, increases, η_{SRAM} is reduced. This increases the difference between the time for the system failure and memory cell failures.

Fig. 5.14 presents the remaining life with different memory sizes due to BTDDB for four different usage scenarios. If the size of memory array increases, the initial lifetime of the processor also decreases. Also, as the memory size, *N*, increases, the interval between each ECC bit failure is reduced (see Equations (5.2)-(5.4)). It can be seen that more failure bits are needed to monitor system lifetime with larger memory systems (see Fig. 5.14). A larger SRAM array can lead to more frequent system monitoring and improves the accuracy and resolution of the diagnosis. Moreover, an

SRAM array that is too small may not have sufficient ECC failures, as illustrated in Fig. 5.9.

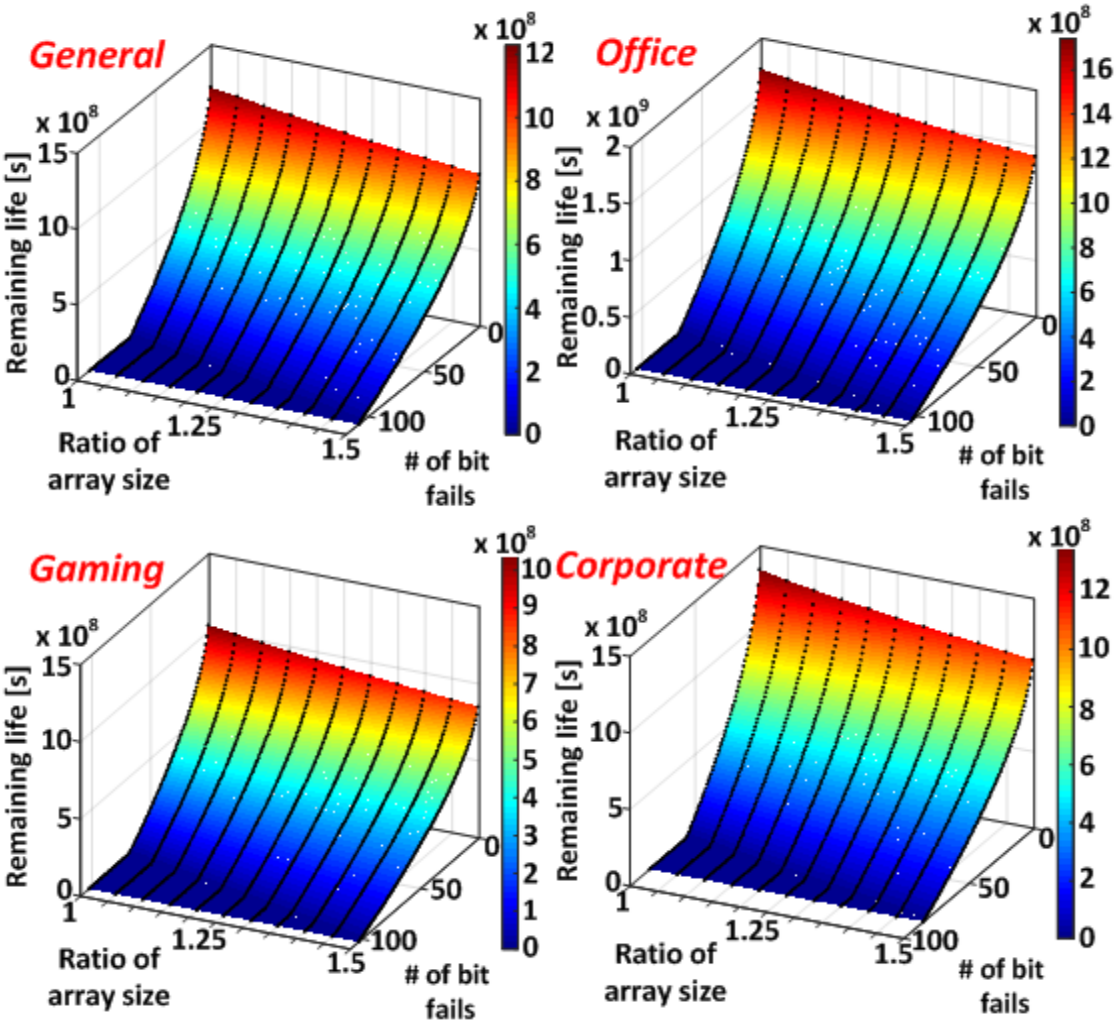

Figure 5.14 Simulation results for the remaining lifetime vs. the number of failed bits for the LEON3 for various use conditions for BTDDB with different SRAM sizes.

5.3.2 Impact of Memory Supply Voltage on the Estimation

Fig. 5.15 shows the impact of the memory supply voltage on the remaining life of the system due to the BTI mechanism. For the BTI mechanism, the limiting performance that determines the remaining lifetime is the read static noise margin (SNM) [9],[76]. With a lower memory supply voltage, the lifetime of the SRAM system, η_{SRAM},

decreases because a lower VDD reduces the read SNM [9],[76]-[78]. The lifetime of an SRAM decreases significantly when the supply voltage varies from 1.1V to 0.9V [78].

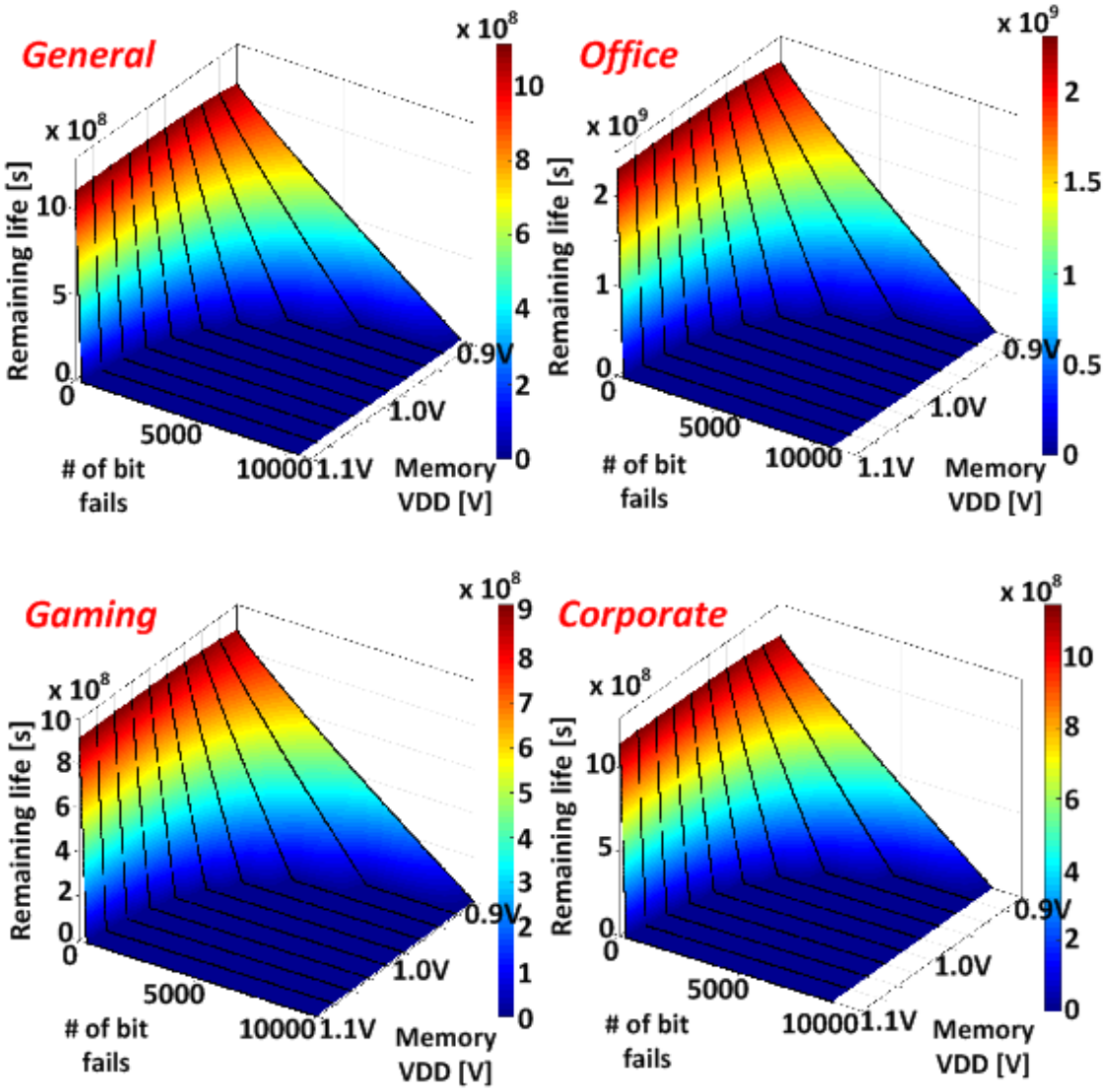

Figure 5.15 Simulation results for the remaining lifetime due to BTI mechanism with different supply voltages.

Since the SRAM lifetime is significantly reduced with a lower supply voltage, the time between each ECC fail decreases (see Equation (5.2)). Fig. 5.15 shows that if the memory supply voltage is reduced, the number of memory cell failures prior to system failure increases substantially. The lifetime of a logic block is similarly affected. Hence,

overall, the memory supply voltage also should be carefully taken into account in the simulation platform in Fig. 5.1 to enable proper calibration of the remaining life estimate as a function of the memory bit failures.

5.3.3 Impact of Temperature on the Estimation Result

Temperature can have an impact on the lifetime of each mechanism. Especially, for the BTI presented in [79], a higher temperature accelerates the threshold voltage shift, leading to a reduction of the lifetime for both logic and memory components. Fig. 5.16 shows extreme cases for the impact of temperature on the remaining lifetime results. If the operating temperature increases, both the lifetimes of logic and memory components can decrease. Also, the number of bit failures prior to the system failure also decrease.

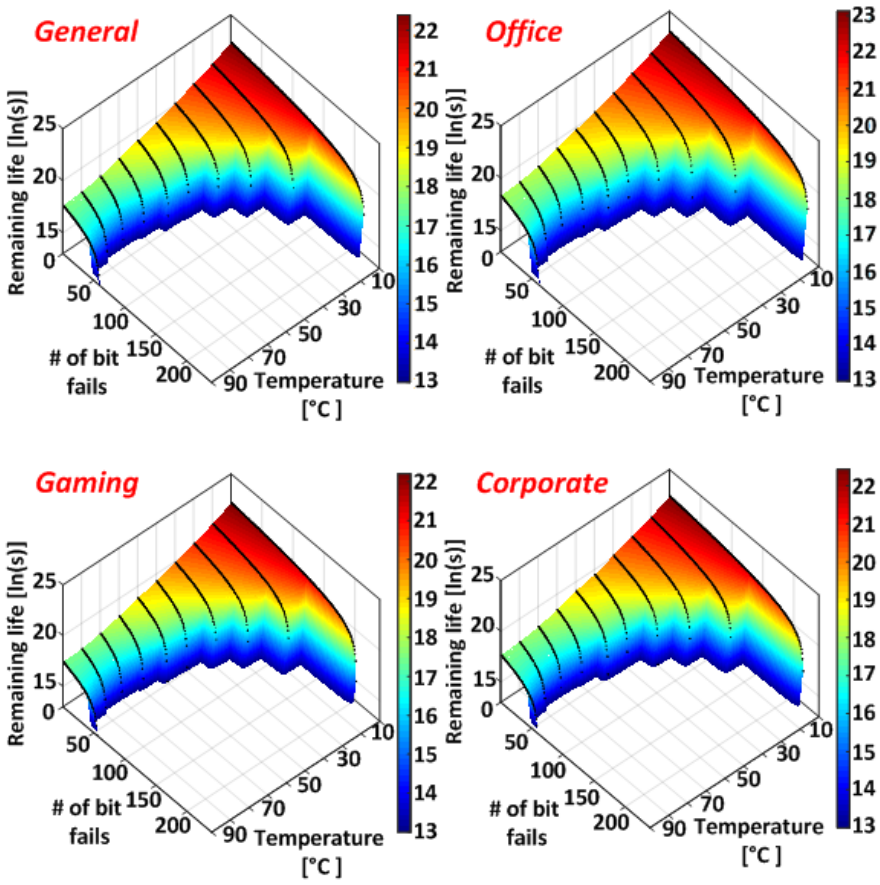

Figure 5.16 Simulation results for the remaining lifetime due to BTI mechanism with different temperatures.

5.3.4 Impact of Process Variations on the Estimation Result

Fig. 5.17 presents that process variations in the length and threshold voltage of each transistor cause variations in the remaining life profile for the BTI. We applied the extreme cases of 10% to both threshold voltage and channel length variations to analyze the variation in the remaining lifetime profile. A negative Vth shift or a negative length shift due to process variations leads to an increase in the initial processor lifetime. The opposite direction of process variations for both Vth and length accelerates the device degradation. Hence, the remaining lifetime estimate should be calibrated for process variations, which can be done through calibration with test structures, such as ring oscillators embedded in manufactured circuits.

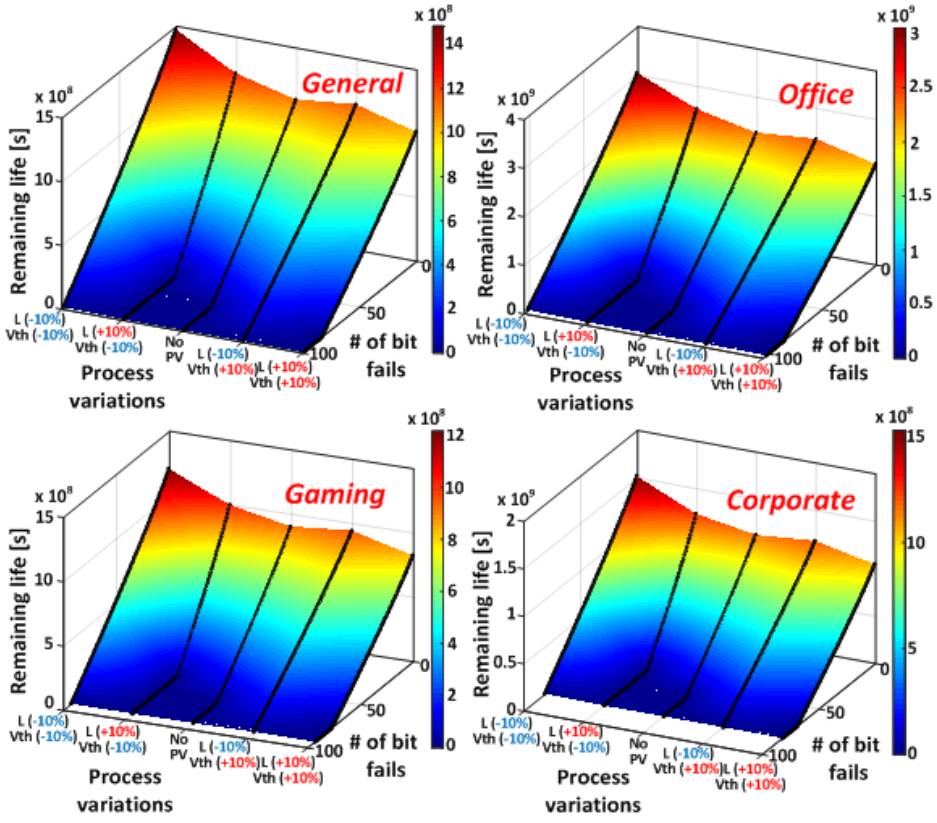

Figure 5.17 Simulation results for the remaining lifetime for BTI mechanism with process variations in channel length (+-10% corners) and threshold voltage (+-10% random variations) for four different usage scenarios.

5.3.5 Impact of Parameters on Ratio between Failure Time for Processor and Memory

Fig. 5.18 shows the ratios between the time to system failure and the first five ECC bit failures as a function of different parameters, including memory array size, memory supply voltage, operating temperature, and process variations. This metric can be used to determine if there are a sufficient number of memory bit failures prior to system failure to enable the use of ECC bit fails as the indicator for the remaining life for the processor. Hence, the memory specifications and design parameters can be defined and controlled properly based on the correlations between the parameters and the ratio.

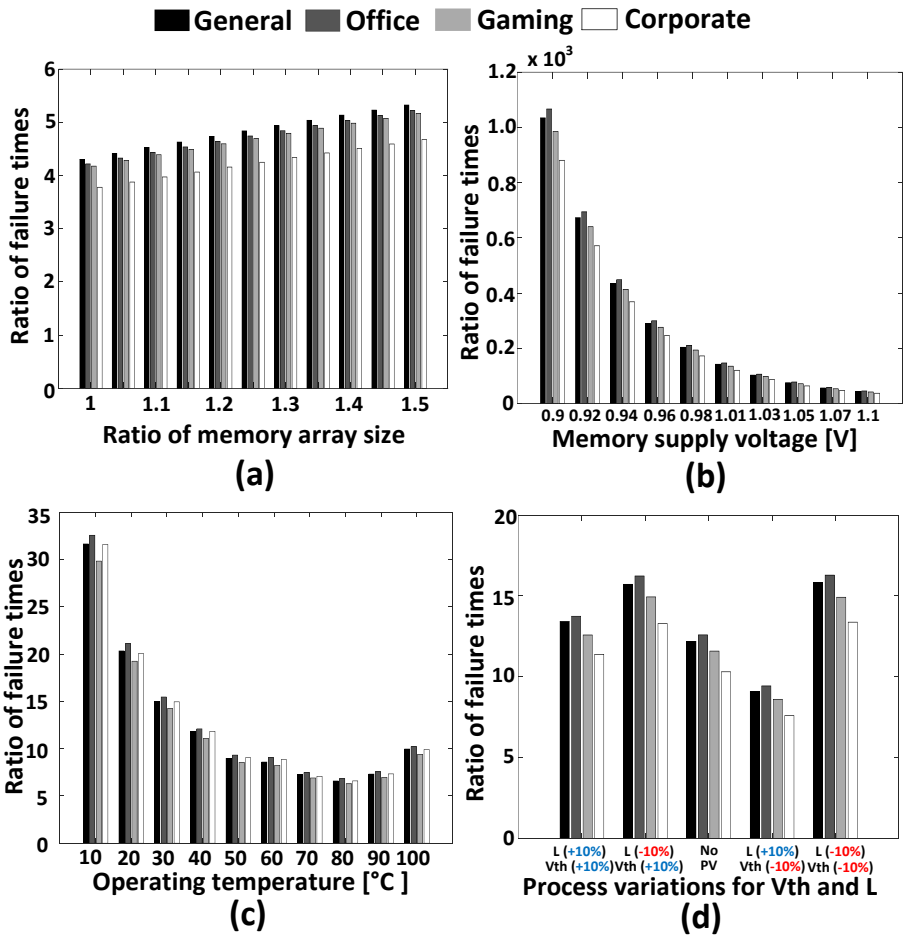

Figure 5.18 Simulation results for the ratio between the time to failure for the LEON3 and the first five ECC bit failures for four different usage scenarios with (a) different memory sizes for BTDDB, (b) different memory supply voltages for BTI, (c) different operating temperatures for BTI, and (d) process variations for BTI.

CHAPTER 6

3D DRAM DESIGN FOR THE OPTIMIZATION OF RELIABILITY, POWER, AND PERFORMANCE

The goal of the project is to investigate the optimized solution for a 3D DRAM system, regarding the reliability issues induced by TSVs with power, performance, area, and cost requirements. We propose new cell/logic partitioning methodology and design schemes for the 3D DRAM and compare the critical metrics for different design styles.

6.1 Design Schemes for Different Cell/Logic Partitioning Methods

We propose a cell/logic-split design which incorporates 5 tiers of DRAM dies that altogether provide 32 Gb of DDR3 memory (see Fig. 1.3(b)). Our design is based on 20nm technologies. We used two poly layers, i.e., bitline poly and wordline poly, and three metal layers in the DRAM arrays. The TSVs used in this 3D DRAM stacking are via-last with 10um diameter and 60um pitch [80]. Each contains 656 signal TSVs that are located in the middle and 100 power/ground (P/G) TSVs on both the top and bottom. In the master die, we add additional 60 P/G pads each in the top and the bottom for 3D power noise reduction as presented in [80]. Each slave die contains a 8Gb DRAM array. The data rate of the 3D DRAM is 1,600Mbps based on the burst length of 8. The Vdd for the slave die is 1.5V and for the master die is 1.3V.

The bottom master die consists of peripheral circuits, I/O pads/circuits, buffers, and serializer/deserializers (see Fig. 6.1(b)). We move most peripheral circuits between GIO drivers and I/O circuitry to the bottom die to reduce the total TSV usage, chip area, and reliability impact. We define the peripheral circuits for one DQ as the *DQ Peripheral Unit* (DQPU). Each DQPU handles the communication between GIO drivers and one I/O

pad. We also have empty space available for extra logic in the master die. Using the advanced process technology in this logic only master die [81], we can use transistors with shorter channel lengths and low Vth to optimize design quality further.

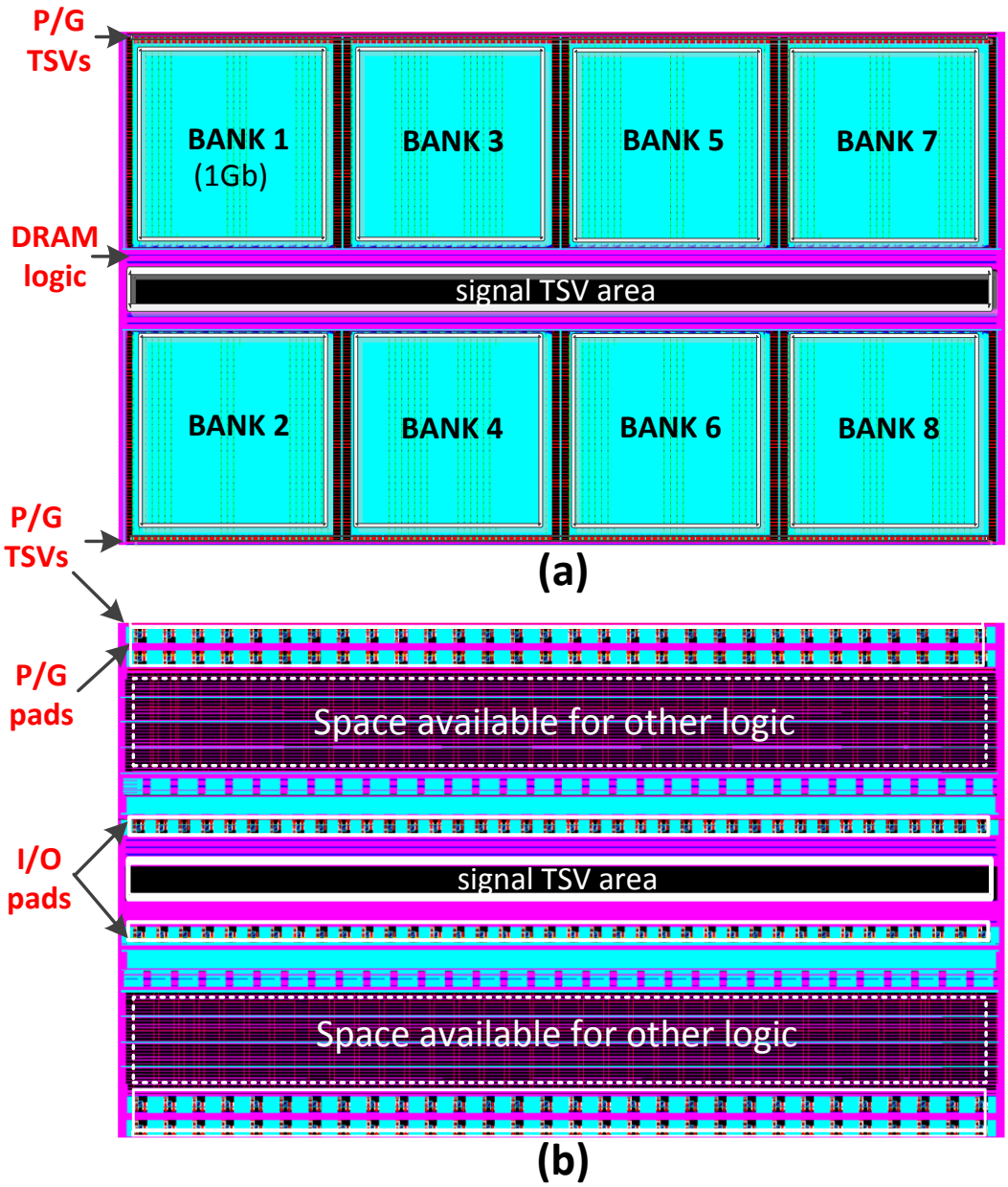

Figure 6.1 Full-chip layouts (a) slave die of cell/logic-split design, (b) master die of cell/logic-split design [20].

Our related experiments show that with high-speed logic and reduced RC parasitic effects on the data paths, we were able to reduce the size of the peripheral

circuits significantly (up to 27%) and use a lower supply voltage (1.3V) for the bottom die. Then, this can reduce power consumption significantly (by 23.6% for a write operation and 27.3% for a read operation) and leads to *tRCD*write reduction of 1.9ns (15.6%). More details are provided in section 6.3. The top four slave dies contain DRAM cells, decoders, sense amps, parts of logic, and GIO drivers (see Fig. 6.1(a)). The logic portions located in the slave dies are mostly logic devices with very small metal pitches used to drive DRAM cell cores and decoders.

In the cell/logic-mixed partitioning style [80], on the other hand, the four dies are almost identical except for the bottom (= master die) that consists of I/O pads and interface circuits (see Fig. 6.2). Each slave die contains 8 Gb DRAM cells, 400 signal TSVs and 100 P/G TSVs. In the master die, the I/O pads and interface circuitry occupy a large area. There are two major problems with this style. First, the large area of I/O pads/buffers is expected to become more critical with today's 20-30nm DRAM process technology, since their size may not scale with DRAM cell technology. Second, the package bumps below I/O pads can cause a non-trivial reliability problem in DRAM cells. This is mainly induced by the CTE (co-efficient of thermal expansion) mismatch among various materials in that area, including the chip/package substrate, micro-bumps, and underfill, leading to a highly compressive stress on DRAM cells [82],[83] (see Fig. 2.1). However, in our cell/logic-split design, this compressive stress does not influence DRAM cells since we separate I/O pads/interface circuits and package bumps from the dies that contain DRAM cells.

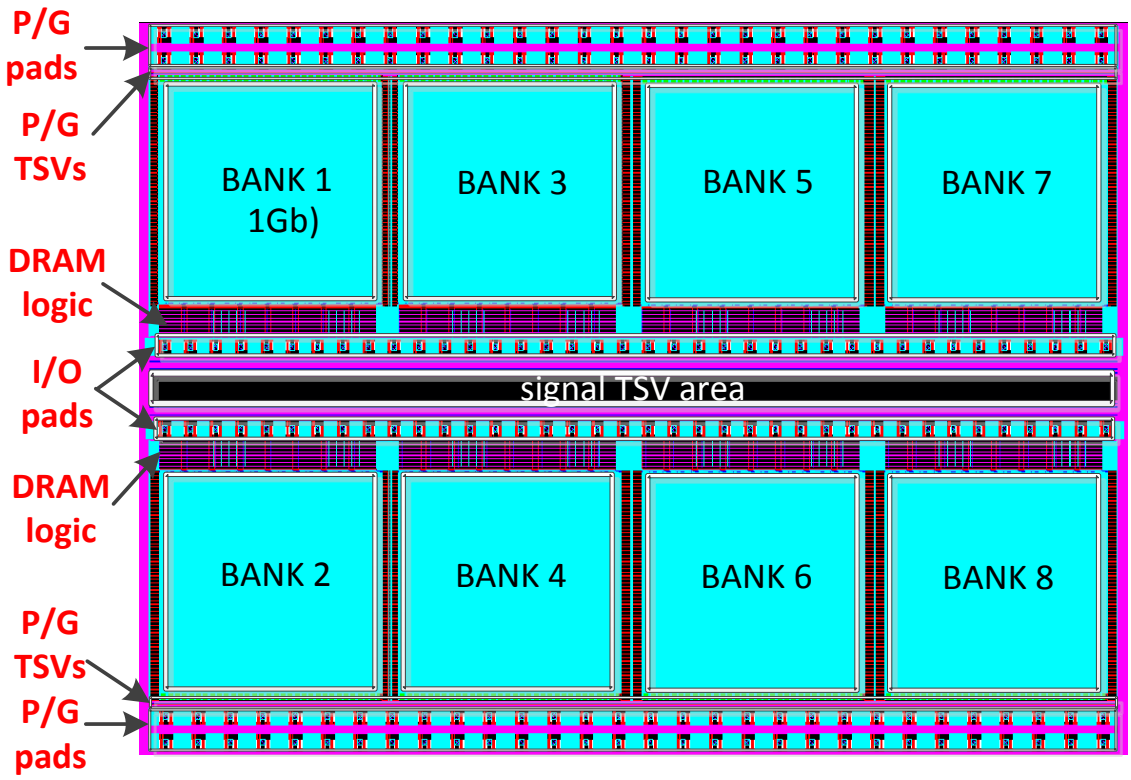

Figure 6.2 Full-chip layout of master die of cell/logic-mixed design.

6.2 Design Solutions For TSV Reduction

In cell/logic-mixed design [80], each bank uses 64 DQPUs to handle 8 DQ signals in 8 burst mode. Fig. 6.3(a) presents this structure. Hence, 512 DQPUs are placed with 8 DRAM banks in each die. Then, TSVs are used for the connections among DQPUs in all slave dies and to I/O pads in the bottom die. Note that these TSVs are *time shared* among 4 DRAM dies. 256 DQPUs on the left half of the die share 128 TSVs and another 256 DQPUs on the right half share another set of 128 TSVs. Thus, the total number of DQ TSVs designed in each die is 256. In addition to these TSVs used for DQ paths, each die contains 144 signal TSVs that are used for address and control signals. The summary is presented in Table 6.1.

In our cell/logic-split design, however, all of the DQPUs and I/O pads are located in the master die. In this case, each data line between a DRAM bank and its DQPU requires a dedicated connection and must distinguish between read and write operations. Thus, 4096 *non-shared* TSVs (= 2 x 8 DQs x 8 burst length x 8 banks x 4 dies) are utlized in the master die, where 75% of them are "feed-through TSVs" that provide connections between the master and other slave dies. Fig. 6.3(b) presents this scheme. The high TSV usage poses challenges in area and reliability issues. In this section, we propose two solutions to solve this problem.

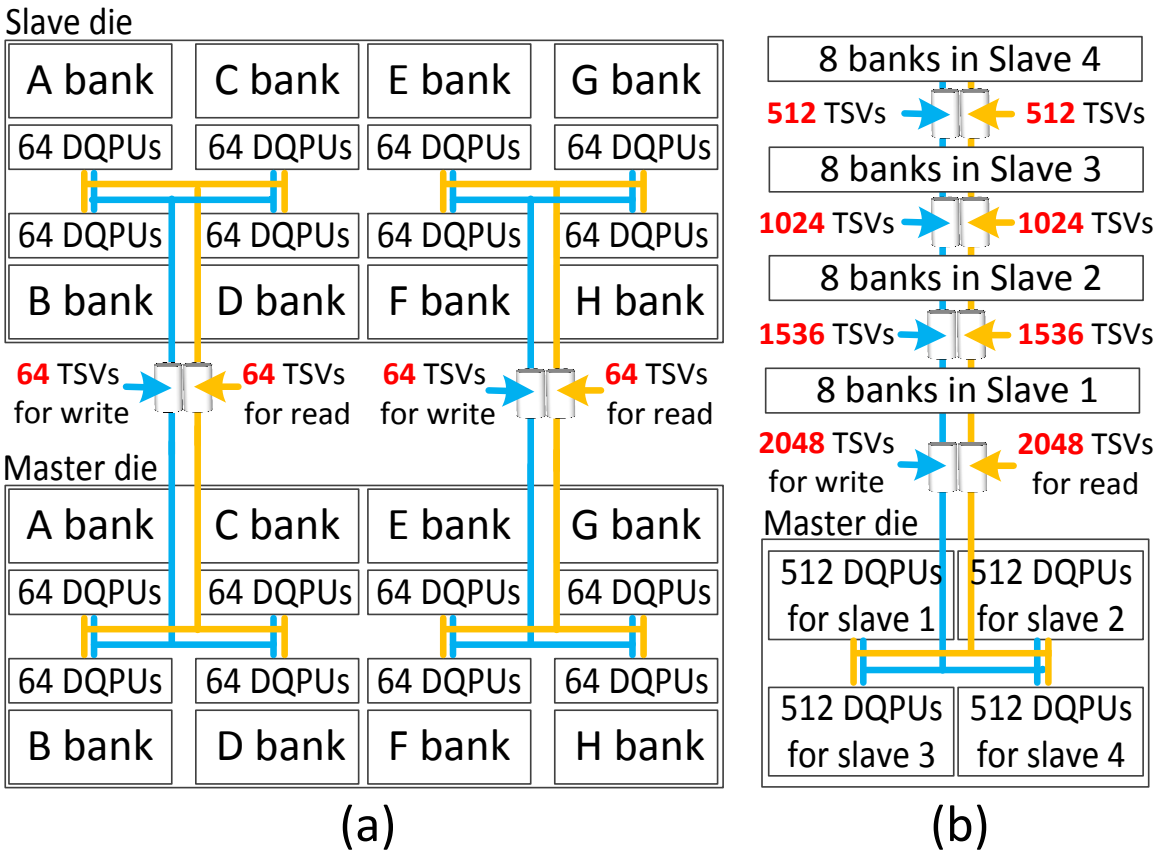

Figure 6.3 DQ TSVs and DQ peripheral unit usages (a) cell/logic mixed design [80], (b) cell/logic-split design w/o TSV reduction.

TABLE 6.1 COMPARISON OF SIGNAL TSV AND DQPU USAGE ON PER DIE BASIS

Any die of cell/logic-mixed design				
	DQ TSVs	Other TSVs	Signal TSVs	DQPU
no optimization	256	144	400	512
Master die of cell/logic-split design				
	DQ TSVs	Other TSVs	Signal TSVs	DQPU
no optimization	4096	144	4340	2048
bank-level sharing	2048	144	2192	1024
die-level sharing	1024	144	1468	512
both solutions	512	144	656	256

- **Bank-level DQPU Sharing:** DQPUs between a pair of an active and an inactive bank can be shared as presented in Fig. 6.4(a). Note that the advanced process technology for the peripheral circuits is used in the master die of cell/logic split design. This leads our DQPUs in the master die to be able to drive larger loads. In addition, we add switches in the GIO drivers between a DQPU and its two banks so that we can disconnect the loads from the inactive bank and its data paths from the DQPU. Hence, our DQPUs need to drive the loads from active banks. This bank-level sharing scheme also leads to a significant reduction in both DQ TSV and DQPU counts by 2x. Table 6.1 presents details on the savings.

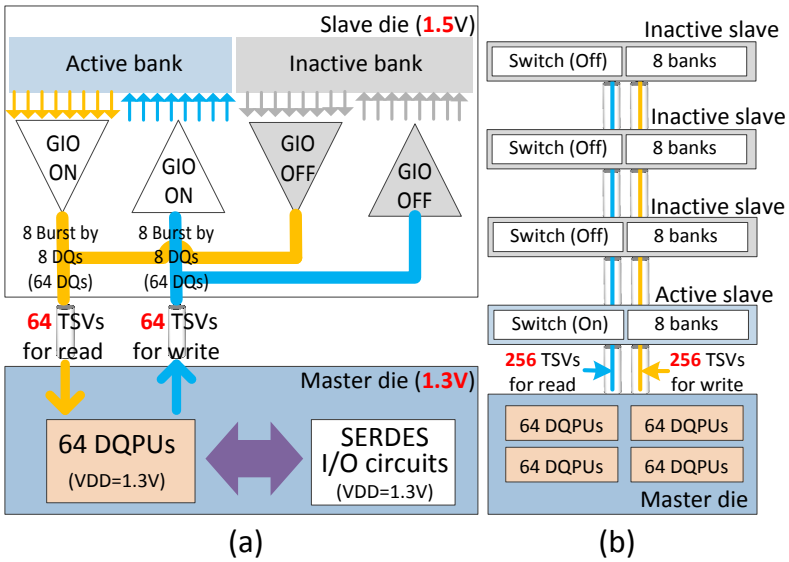

Figure 6.4 Illustration of our TSV reduction solutions (a) bank-level DQPU sharing, (b) die-level DQPU sharing.

- **Die-level DQPU Sharing**: DQPUs among the DRAM banks are shared in different tiers. Fig. 6.4(b) presents this scheme. In the original cell/logic-split design, each bank is connected to 64 DQPUs in the master die as presented earlier. However, only one die is activated during a read/write operation. This means that a group of 64 DQPU sets are shared among 4 banks in 4 slave dies so that we can disconnect 3 inactive dies using switches in those dies and drive only one from the active die. This leads to 4x savings in both DQPU and DQ TSV counts. Table 6.1 shows details on the savings. We note from Table 6.1 that with both solutions combined, the total DQ TSV usage is reduced from 4,096 to 512 and DQPU usage is reduced from 2,048 to 256. This corresponds to 2x worse DQ TSV usage (512 for split design vs 256 for mixed design) and 1.64x worse signal TSV usage (656 for split design vs 400 for mixed design). In case of DQPU savings, our split design uses 256 DQPUs in the entire 5 dies, whereas the mixed style uses 512 DQPUs in each die. Hence, the total DQPU count is 2048 in the cell/logic-mixed style, which leads to 8x savings with our split style.

6.3 Simulation Results

We merge GDSII files for both analog and digital circuit parts using Virtuoso and perform sign-off analysis using HSPICE and Synopsys PrimeTime for timing and power calculations. PrimeTime is built for 2D IC analysis, and we have extended it to handle 3D DRAM. We also use the full-chip mechanical stress and mobility variation analysis tools studied in [82].

6.3.1 Reliability Simulation

Fig. 6.5(a) presents simulation results of mechanical stress in the S11-direction for cell/logic-mixed design. The significant mechanical stress induced by CTE (co-

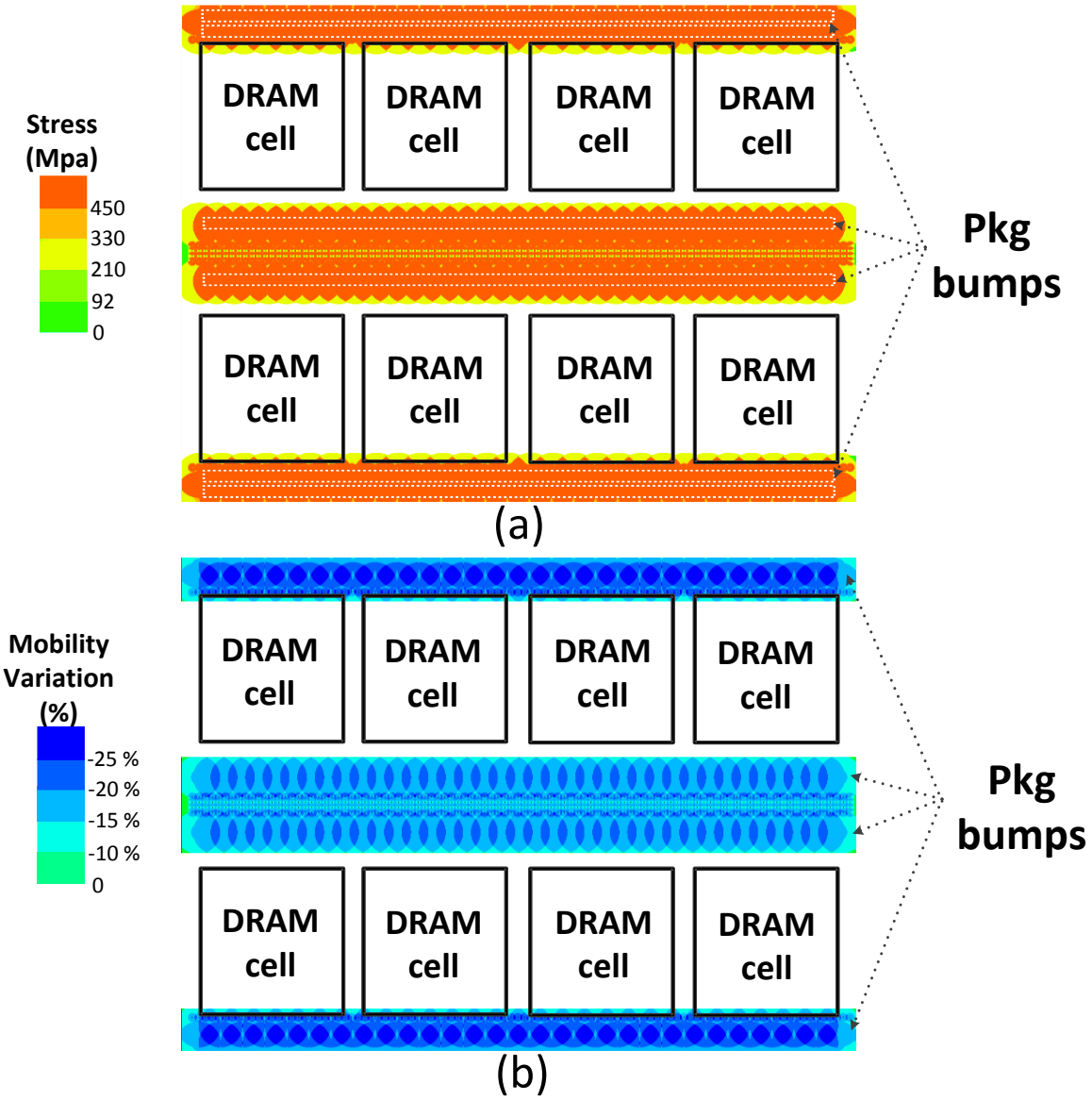

Figure 6.5 Reliability simulation for master die of cell/logic-mixed design with 20um Keep-Out-Zone (a) full-chip analysis for mechanical stress, and (b) full-chip analysis for mobility variations.

efficient of thermal expansion) mismatch among package bumps, micro-bumps, and TSVs mostly affects the area near the TSV arrays located in the middle, top, and bottom of the die. The mechanical stress may cause serious structural damage, such as cracks in the substrate and TSVs, delamination of the TSV liner, and TSV protrusion [84]-[86]. These issues in turn affect the overall yield, because the chips may not meet the

performance specification and/or may have mechanical faults. Also, the mechanical stress induced by TSVs decreases electron mobility of DRAM cell transistors near the top and bottom edges as presented in Fig. 6.5(b) [87]. The variation of electron mobility introduces undesirable timing violations and may lead to read/write failures.

TABLE 6.2 RELIABILITY COMPARISON

	CELL/LOGIC-MIXED	CELL/LOGIC-SPLIT
MECHANICAL STRESS		
AREA OVER 450MPA	36.8%	4.37%
MAXIMUM STRESS	1350.4MPA	688.1MPA
MOBILITY VARIATION		
AREA OVER 15%	34.8%	5.01%
MAXIMUM VARIATION	55.2%	37.7%

In Table 6.2 we present a comparison of mechanical reliability and mobility variation between cell/logic-mixed vs cell/logic split design styles. We have focused on the area with more than 450MPa mechanical stress and 15% mobility variation. We find that our cell/logic-split design presents a lower mechanical stress and mobility variation impact. Since there are no package bumps under the substrate that consists of DRAM arrays in cell/logic-split design, mechanical stress is only due to TSVs and micro bumps. This significantly alleviates mechanical stress and electron mobility variation compared with those for cell/logic-mixed design [83]. The maximum stress is smaller (688.1Mpa vs 1350.4Mpa) in the cell/logic-split design (see Fig. 6.6).

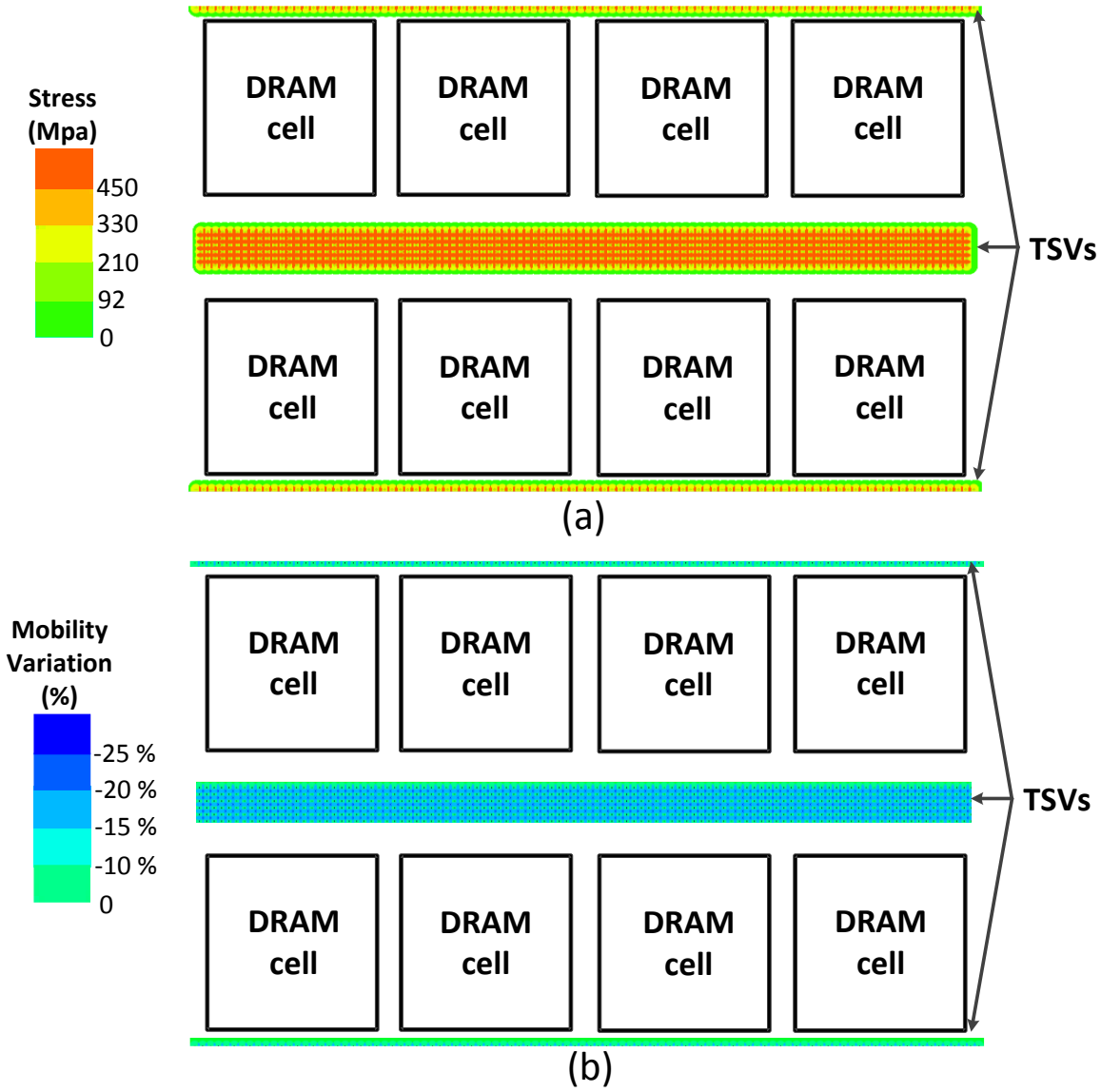

Figure 6.6 Reliability simulation for slave die of cell/logic-split design with 20um Keep-Out-Zone (a) full-chip analysis for mechanical stress, (b) full chip analysis for mobility variations.

6.3.2 Power Consumption Simulation

Using a more advanced process technology in the master die of the cell/logic-split design, the size of logic devices (up to 27%) can be reduced and operated at a lower Vdd (= 1.3V) as shown in Fig. 6.4(a). Our power analysis in Table 6.3 and Fig. 6.7 presents that our device scaling and low supply voltage (1.3V) together improve the power

consumption of DQPUs and I/O circuits for both read and write operations. This leads to a total power consumption reduction by 23.6% for write operations and 27.3% for read operations in our cell/logic split design at 1.3V Vdd. The power values are comparable to those of cell/logic-mixed design [80].

TABLE 6.3 POWER ANALYSIS FOR DQ DATAPATH ELEMENTS

Write operation			
	Cell/logic-mixed (1.5V)	Cell/logic-split (1.5V)	Cell/logic-split (1.3V)
8 DQPUs	7.05 mW	4.58 mW	3.48 mW
1 I/O SERDES	8.45 mW	7.25 mW	5.87 mW
8 GIO drivers	8.91 mW	9.11 mW	9.29 mW
Total	24.41 mW	20.94 mW	18.64 mW
Read operation			
	Cell/logic-mixed (1.5V)	Cell/logic-split (1.5V)	Cell/logic-split (1.3V)
8 DQPUs	12.9 mW	7.14 mW	5.36 mW
1 I/O SERDES	12.2 mW	10.8 mW	8.39 mW
8 GIO drivers	14.6 mW	15.1 mW	15.1 mW
Total	39.70 mW	33.04 mW	28.85 mW

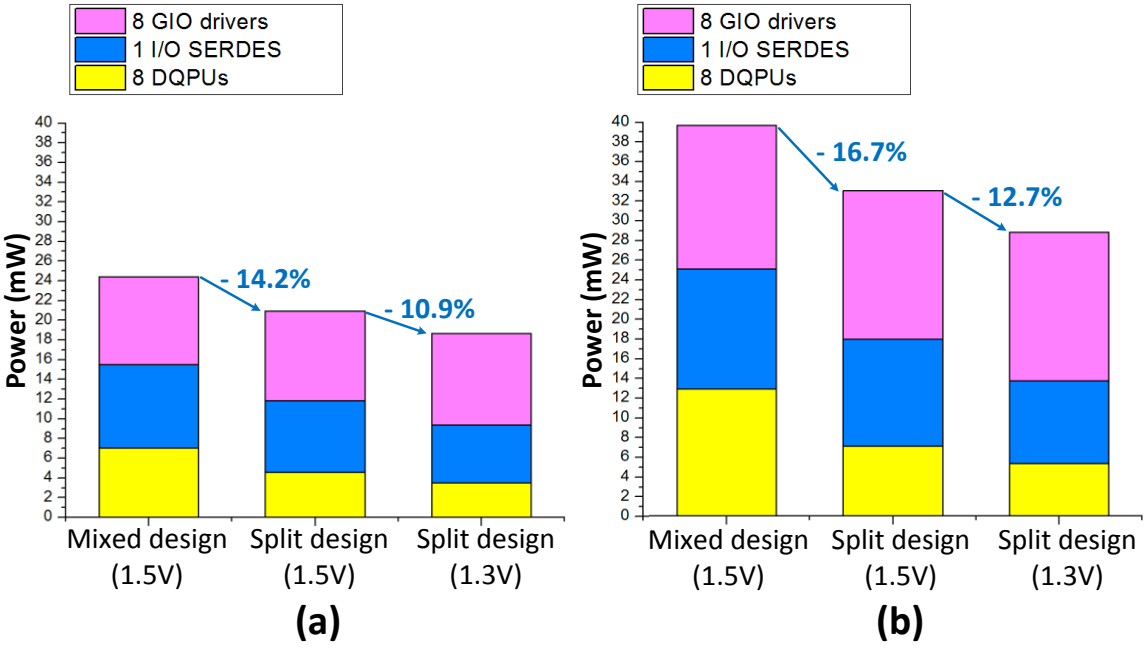

Figure 6.7 Power simulation comparison for (a) write operation, (b) read operation for both design styles.

6.3.3 Performance Simulation

Fig. 6.8 presents HSPICE simulations of the DQ peripheral circuit for the write operation. Using the advanced logic process in the master die of the split design, DQPUs in that die can be designed with transistors with shorter channel lengths and low Vth, leading the DQPU units to handle the load even with bank-level and die-level DQPU sharing schemes effectively. All of these benefits lead to a *tRCD*write reduction of 1.9ns (15.6%) on a DQ data line, as presented in Fig. 6.8.

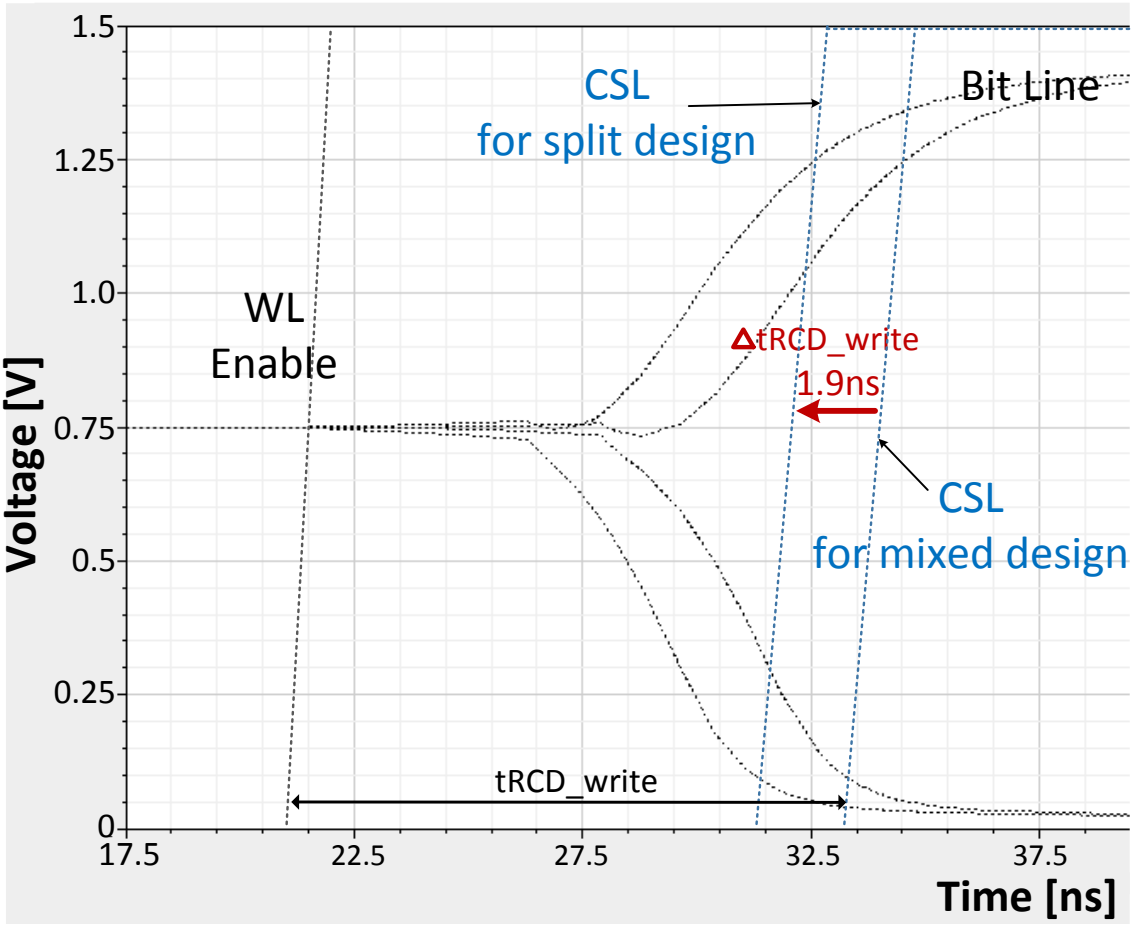

Figure 6.8 HSPICE simulations for write operation (*tRCD*write) with split design and mixed design.

6.3.4 Yield and Cost Analysis

The definition of yield (Y) is the number of good chips divided by the number of chips that are manufactured. Most of yield analysis has focused on wafer probe yield (Y_{probe}) because most yield loss appears at wafer probe. Y_{probe} is defined as the portion of chips which pass tests and can be modeled by two yield parameters (see equation (6.1)). The first parameter is random yield (Y_{random}) that depends on randomly placed defects. The second metric is systematic yield (Y_{sys}) which includes all other causes for yield loss at wafer probe. We focus on the Y_{random} parameter for our yield analysis of 3D DRAM because Y_{sys} has generally been considered to be approximately equal to one [88],[89].

Y_{random} is modeled by the Poisson yield model as presented in equation (6.2). The Poisson yield model is based on an assumption that particles can be randomly distributed throughout a wafer. The probability that a defect kills the chip for each layer is λ_i in equation (6.3) and is a function of the defect density, D_i, the vulnerable area, $A_i(r)$, and the defect size distribution, $f_i(r)$, for the ith failure mechanism [88]. We assume that $f_i(r)$ and D_i are the same in the same wafer.

$$Y_{probe} = Y_{sys} Y_{random} \tag{6.1}$$

$$Y_{random} = \exp(-\lambda_i) \tag{6.2}$$

$$\lambda_i = D_i \int_0^\infty A_i(r)\, f_i(r) dr \tag{6.3}$$

$A_i(r)$ is the critical area that is vulnerable due to a defect with radius r. Here, when there are different design rules for space, because typically $f_i(r) = k/r^3$, the narrow spaces will dominate the calculation of $\int_0^\infty A_i(r)\, f_i(r) dr$, unless there is much more area with the wide spaces, or if the narrow space part has enough redundancy to

tolerate defects [88],[90]. In 3D DRAM, the DRAM core area has a significantly smaller feature size and is consequently much more vulnerable to be damaged by defects during manufacturing processes. On the other hand, peripheral circuit parts and control circuits with a larger feature size are much more tolerant to the defects. Since the DRAM cell area is the same in each design style, we can approximately make the assumption that $\int_0^\infty A_i(r)\, f_i(r) dr$ is the same in each style. The equation (6.3) shows that the random yield for each layer is mainly determined by D_i with the assumption that $\int_0^\infty A_i(r)\, f_i(r) dr$ and $f_i(r)$ are the same in each die.

The defect density (*D*) is presented in equation (6.4) where A_w is total inspected area on a wafer and λ_k is the average number of killer defects [91]-[93].

$$D = \lambda_k / A_w \tag{6.4}$$

There are three possible types of defects in a wafer: killer defects which cause failures in the circuit, latent defects which are either too small or inappropriately distributed to cause an immediate circuit failure, and defects which do not cause any failure because of their size and/or composition. The ratio of the average number of latent defects to the average number of killer defects is a function of process technology and inspection methodology. For our DRAM technology, because yield loss is dominated by the DRAM core area, the defect density should be calculated considering only the DRAM core area. Nevertheless, it is not easy to distinguish killer and nonkiller defects with in-line inspection. As an alternative, wafer probe can detect only killer defects. If A_{core} is the area of the DRAM core, if there is no redundancy, defect density, *D*, is computed as $D = -\ln(Y)/A_{core}$. When there is redundancy, this equation substantially underestimates

the defect density. Specifically, when there is a capability to correct n defects, then the defect density is the solution of $Y = \sum_{x=0}^{n} DA_{core}{}^{x} \exp(-DA_{core}) / x!$ [88],[91],[94].

TABLE 6.4 COMPARISON OF AREA AND # OF MANUFACTURED CHIPS

	Mixed design	Split design
# of chips in 12-inch wafer	1064	1342
Peripheral component in slave die	45.3%	29.1%

Note that profit is a function of the total number of chips that are sold, which is a product of the yield and the number of manufactured chips per wafer [88]. The total profit depends on the yield, the number of manufactured chips, bonding costs, and cost of additional logic die for the split design. Table 6.4 shows that the smaller footprint of split design leads an increase in the number of chips that can be manufactured per wafer. For a set of N wafers, the cell/logic mixed design produces $1064NY$ good chips. Since each product requires 4 good DRAM chips, mixed design produces $266NY$ products. On the other hand, cell/logic split design requires five chips, of which four will have the DRAM core. Hence, the same N wafers can produce $268.4N(4Y + 1)$ good chips and $268.4NY$ good products. Hence, cell/logic split design produces on average $2.4Y$ more good product per wafer. On the other hand, since the yield is higher for the master die for split design, it becomes possible to allocate more wafers to the slave die. Specifically, for every M master die, we need $4M/Y$ slave die for the split design. Then, N wafers produce $1342N/(1+ 4/Y)$ good products with the split design style. Hence, when the yield drops below 100%, the number of good products produced per wafer increases. For example, when the yield is 50% for the DRAM core, then split design produces 16 more products per wafer than mixed design.

CHAPTER 7

CONCLUSION

The object of the proposed research is to develop comprehensive methodologies, including circuit design, new test methodologies, and statistical failure analysis, to implement reliable microprocessor and main memory systems. For a microprocessor, we have focused on the reliability issues in the embedded cache, since SRAMs are designed with the tightest design rules, and high performance processors are expected to consist of a large embedded memory. Also, to solve the scaling challenges for the main memory system, we have studied optimized design schemes for the 3D DRAM system, to achieve better performance, reliability, cost, and power.

To implement a reliable microprocessor, this research has focused on wearout mechanisms, namely BTI, GTDDB, EM, SIV and BTDDB, in the embedded cache systems. The research has presented built-in self-test and statistical analysis methodologies for electrical detection and diagnosis of wearout mechanisms in an SRAM to improve the manufacturing process. Also, based on the diagnosis result, this research work has proposed to use the ECC failure bits as the mileage monitor for the remaining lifetime of the processor.

Although 3D DRAM had been proposed as a feasible candidate for the main memory system, the reliability issues and area overhead induced by TSVs with the limited budgets of performance and power were regarded as one of the critical bottlenecks for mass production. In this dissertation, we have proposed the optimized design solutions to provide the solution for the tradeoff relationship between the critical parameters.

APPENDIX A

PUBLICATIONS

This dissertation is based on and/or related to the works presented in the following publications:

[1] **W. Kim**, C.-C. Chen, D.-H. Kim, and L. Milor, "Built in self test methodology with statistical analysis for electrical diagnosis of wearout in a static random access memory array," *IEEE Trans. VLSI*.

[2] **W. Kim**, C-C. Chen, T. Liu, and L. Milor, "Dynamically Monitoring System Health Using On-Chip Caches as a Wearout Sensor," *IEEE Trans. VLSI* (under review).

[3] **W. Kim**, C.-C. Chen, S. Cha, and L. Milor, "MBIST and statistical hypothesis test for time dependent dielectric breakdowns due to GOBD vs. BTDDB in an SRAM array," Proc. IEEE VLSI Test Symposium, 2015.

[4] **W. Kim** and L. Milor, "Built-in self test methodology for diagnosis of backend wearout mechanisms in SRAM cells," Proc. IEEE VLSI Test Symposium, 2014.

[5] **W.Kim**, D.-H. Kim, H. Hong, L. Milor, and S. Lim. "Impact of die partitioning on reliability and yield of 3D DRAM," Proc. IEEE International Interconnect Technology Conference/Advanced Metallization Conference (IITC/AMC), 2014.

[6] **W. Kim**, C-C. Chen, T. Liu, and S. Cha, "Estimation of remaining life using embedded SRAM for wearout parameter extraction." Proc. IEEE Int. Workshop on Advances in Sensors and Interfaces, 2015.

[7] **W. Kim**, S. Cha, and L. Milor, "Memory BIST for On-Chip Monitoring of Resistive-Open Defects due to Electromigration and Stress-Induced Voiding in an SRAM Array," Proc. Conf. on Design of Circuits and Integrated Systems, 2014.

[8] **W. Kim**, C.-C. Chen, and L. Milor, "Diagnosis of resistive-open defects due to electromigration and stress-induced voiding in an SRAM array," Proc. International Integrated Reliability Workshop (IIRW), 2014.

[9] **W. Kim**, D.-H. Kim, H. Zhou, and L. Milor, "Numerical Optimization of Stress Accelerated Test Plans for Diagnosis of Wearout in On-Chip Caches", *IEEE Trans. on Reliability* (under review).

REFERENCES

[1] S.-K. Lu, C.-L. Yang, Y.-C. Hsiao, and C.-W. Wu, “Efficient BISR techniques for embedded memories considering cluster faults,” *IEEE Trans. VLSI*, vol. 18, no. 2, pp. 184-193, Feb. 2010.

[2] R. Alves Fonseca, L. Dilillo, A. Bosio, P. Girard, S. Pravossoudovitch, A. Virazel, and N. Badereddine., "Analysis of resistive-bridging defects in SRAM core-cells: A comparative study from 90nm down to 40nm technology nodes," Proc. IEEE European Test Symp., 2010, pp. 132-137.

[3] L. Dilillo, P. Girard, S. Pravossoudovitch, A. Virazel, S. Borr, and M. Hage-Hassan., "Resistive-open defects in embedded-SRAM core cells: Analysis and March test solution," Proc. Asian Test Symp., 2004.

[4] C.-C. Chen and L. Milor, "Microprocessor aging analysis and reliability modeling due to back-end wearout mechansism," *IEEE Trans. VLSI*, 2015.

[5] C.-C. Chen, F. Ahmed, and L. Milor, "A comparative study of wearout mechanisms in state-of-art microprocessors," IEEE Int. Conf. Computer Design, 2012.

[6] C.-C. Chen and L. Milor, "System-level modeling and microprocessor reliability analysis for backend wearout mechanisms," Design Automation and Test in Europe, 2013.

[7] C.-C. Chen and L. Milor, "System-level modeling and reliability analysis of microprocessor systems," IEEE Int. Workshop on Advances in Sensors and Interfaces, 2013.

[8] C.-C. Chen, F. Ahmed, and L. Milor, "Impact of NBTI-PBTI on SRAMs within microprocessor systems: modeling, simulation, and analysis," *Microelectronics Reliability*, vol. 53, no. 9-11, pp. 1183-1188, Sept.-Nov. 2013.

[9] C.-C. Chen, T. Liu, S. Cha, and L. Milor, "System-level modeling of microprocessor reliability degradation due to BTI and HCI," Int. Reliability Physics Symp., 2014.

[10] C.-C. Chen, S. Cha, and L. Milor, "System-level modeling of microprocessor reliablity degradation due to TDDB," Design of Circuits and Integrated Systems, 2014.

[11] J. Blome, S. Feng, S. Gupta, and S. Mahlke, "Online timing analysis for wearout detection," *Workshop on Architectural Reliability, 2006.*

[12] A. Tiwari and J. Torrellas, “Facelift: Hiding and slowing down aging in multicores,” *IEEE/ACM Int. Symp. on Microarchitecture*, 2008.

[13] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, “Exploiting structural duplication for lifetime reliability enhancement,” *ACM SIGARCH Computer Architecture News*, vol. 33, no. 2, pp. 520-531, May. 2005.

[14] LEON3 processor: www.gaisler.com.

[15] S.-Y. Kuo and W.K. Fuchs, “Efficient spare allocation in reconfigurable arrays,” *IEEE Design & Test of Computers*, vol. 4, no. 1, pp. 24-31, Feb. 1987.

[16] B. Sklar and F.J. Harris, “The ABCs of Linear Block Codes,” IEEE Signal Processing Magazine, pp. 14-35, July 2004.

[17] Mibench benchmark: http://www.eecs.umich.edu/mibench.

[18] R. Kwasnick, A.E. Papathanasiou, M. Reilly, A. Rashid, B. Zaknoon, and J. Falk, “Determination of CPU use conditions,” Proc. Int. Reliability Physics Symp., 2011, pp. 2C.3.1-2C.3.6.

[19] U. Kang, et al., "8Gb 3D DDR3 DRAM using through-silicon-via technology," IEEE International Solid-State Circuits Conference-Digest of Technical Papers, 2009.

[20] W.Kim, et al., "Impact of die partitioning on reliability and yield of 3D DRAM," IEEE International Interconnect Technology Conference/Advanced Metallization Conference (IITC/AMC), 2014.

[21] O. Mutlu, "Memory scaling: A systems architecture perspective," IEEE International Memory Workshop (IMW), 2013.

[22] Y. Huai, "Spin-transfer torque MRAM (STT-MRAM): Challenges and prospects," *AAPPS Bulletin*, vol. 18, no. 6, pp. 33-40, 2008.

[23] R. Fackenthal, et al., "19.7 A 16Gb ReRAM with 200MB/s write and 1GB/s read in 27nm technology," IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014.

[24] D. Lee, et al., "25.2 A 1.2 V 8Gb 8-channel 128GB/s high-bandwidth memory (HBM) stacked DRAM with effective microbump I/O test methods using 29nm process and TSV," IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014.

[25] T. Kgil, et al., "PicoServer: using 3D stacking technology to enable a compact energy efficient chip multiprocessor," *ACM SIGARCH computer architecture news*, vol. 34, no. 5, pp. 117-128, 2006.

[26] C. Liu, et al., "Bridging the processor-memory performance gap with 3D IC technology," *IEEE Design & Test of Computers*, vol. 22, no. 6, pp. 556-564, Nov. 2005.

[27] T. Oshima, K. Hinode, H. Yamaguchi, H. Aoki, K. Tori, T. Saito, K. Ishikawa, J. Noguchi, M. Fukui, T. Nakamura, S. Uno, K. Tsugane, J. Murata, K. Kikushima, H. Sekisaka, E. Murakami, K. Okuyama, and T. Iwasaki, "Suppression of Stress-Induced Voiding in Copper Interconnects," Int. Electron Devices Meeting, 2002.

[28] R. Wang, C.C. Lee, L.D Chen, K. Wu, and K.S. Chang-Liao, "A study of Cu/Low-k stress-induced voiding at via bottom and its microstructure effect," *Microelectronics Reliability*, vol. 46, no. 9, pp. 1673-1678, Oct. 2006.

[29] K. Yoshida, et al., "Stress-induced voiding phenomena for an actual CMOS LSI interconnects," IEEE International Electron Devices Meeting, 2002.

[30] A. H. Fischer, et al., "Electromigration failure mechanism studies on copper interconnects," Proc. IEEE International Interconnect Technology Conference, 2002.

[31] Z. Guan, et al., "SRAM bit-line electromigration mechanism and its prevention scheme," IEEE International Symposium Quality Electronic Design (ISQED), 2013.

[32] H. Tsuchiya, and Y. Shinji, "Electromigration lifetimes and void growth at low cumulative failure probability," *Microelectronics Reliability*, vol. 46, no. 9-11, pp. 1415-1420, 2006.

[33] C. J. Christiansen, et al., "Via-depletion electromigration in copper interconnects," *IEEE Trans. Device and Materials Reliability*, vol. 6, no. 2, pp. 163-168, 2006.

[34] B. Li, et al., "Line depletion electromigration characterization of Cu interconnects," *IEEE Trans. Device and Materials Reliability*, vol. 4, no. 1, pp. 80-85, 2004.

[35] D. Li, M. Malgorzata, and S. Nassif, "A method for improving power grid resilience to electromigration-caused via failures," *IEEE Trans. VLSI*, vol. 23, no. 1, pp. 118-130, Jan. 2015.

[36] B. Kaczer, et al., "Impact of MOSFET gate oxide breakdown on digital circuit operation and reliability," *IEEE Transactions on Electron Devices*, vol. 49, no. 3, pp. 500-506, 2002.

[37] R. Rodriguez, et al, "The impact of gate-oxide breakdown on SRAM stability," *IEEE Electron Device Letters*, vol. 23, no. 9, pp. 559-561, 2002.

[38] B. Kaczer, et al., "Analysis and modeling of a digital CMOS circuit operation and reliability after gate oxide breakdown: a case study," *Microelectronics Reliability* vol. 42, no. 4-5, pp. 555-564, 2002.

[39] L. Milor, and C. Hong, "Backend dielectric breakdown dependence on linewidth and pattern density," *Microelectronics Reliability*, vol. 47, no. 9, pp. 1473-1477, 2007.

[40] C. Hsu, et al., "Improvement of TDDB reliability, characteristics of HfO 2 high-k/metal gate MOSFET device with oxygen post deposition annealing," *Microelectronics Reliability*, vol. 50, no. 5, pp. 618-621, 2010.

[41] S. Drapatz, G. Georgakos, and D. Schmitt-Landsiedel, "Impact of negative and positive bias temperature stress on 6T-SRAM cells," *Advances in Radio Science*, vol. 7, pp. 191-196, 2009

[42] S. Bhardwaj, et al., "Predictive modeling of the NBTI effect for reliable design," IEEE Custom Integrated Circuits Conference, 2006.

[43] C.-C. Chen, F. Ahmed, and L. Milor, "Impact of NBTI-PBTI on SRAMs within microprocessor systems: modeling, simulation, and analysis," *Microelectronics Reliability*, vol. 53, no. 9-11, pp. 1183-1188, Sept.-Nov. 2013.

[44] Muhammad Bashir and Linda Milor, "Backend low-k TDDB chip reliability simulator," 2011 IEEE International Reliability Physics Symposium (IRPS).

[45] F. Ahmed and L. Milor, "Analysis of on-chip monitoring of gate oxide breakdown in SRAM cells," *IEEE Trans. VLSI*, vol. 20, no. 5, pp. 855-864, May 2012.

[46] F. Ahmed and L. Milor, "NBTI resistant SRAM design," in Proc. Int. Workshop on Advances in Sensors and Interfaces, 2011, pp. 82-87.

[47] F. Ahmed and L. Milor, "Reliable Cache Design with On-Chip Monitoring of NBTI Degradation in SRAM Cells using BIST," Proc VLSI Test Symp, 2010, pp. 63-68.

[48] T. Kawagoe, J. Ohtani, M. Niiro, T. Ooishi, M. Hamada, and H. Hidaka, "A built-in self-repair analyzer (CRESTA) for embedded DRAMs," IEEE Int. Test Conf., 2000.

[49] C.-T. Huang, C.-F. Wu, J.-F. Li, and C.-W. Wu, "Built-in redundancy analysis for memory yield improvement," *IEEE Trans. Reliability*, vol. 52, no. 4, pp. 386-399.

[50] S. Naik, F. Agricola, and W. Maly, "Failure analysis of high-density CMOS SRAMs," *IEEE Design & Test of Computers*, vol. 10, no. 2, pp. 13-23, June 1993.

[51] N.S. Kim, T. Austin, D. Baauw, T. Mudge, K. Flautner, J.S. Hu, M.J. Irwin, M. Kandemir, and V. Narayanan, "Leakage current: Moore's law meets static power," *Computer*, vol. 36, no. 12, pp. 68-75, 2003.

[52] J.B. Khare, W. Maly, S. Griep, and D. Schmitt-Landsiedel, "Yield-oriented computer-aided defect diagnosis," *IEEE Trans. Semiconductor Manufacturing*, vol. 8, no. 2, pp. 195-206, May 1995.

[53] H. Balachandran and D. Walker, “Improvement of SRAM-based failure analysis using calibrated Iddq testing,” Proc. VLSI Test Symp., 1996.

[54] P. Ohler, S. Hellebrand, H.-J. Wunderlich, “An integrated built-in test and repair approach for memories with 2D redundancy,” IEEE European Test Symp., 2007.

[55] W. Jeong, I. Kang, and S. Kang, “A fast built-in redundancy analysis for memories with optimal repair rate using a line-based search tree,” *IEEE Trans. VLSI*, vol. 17, no. 12, pp. 1665-1678, Dec. 2009.

[56] S.-K. Lu, Y.-C. Tsai, C.-H. Hsu, K.-H. Wang, and C.-W. Wu, “Efficient built-in redundancy analysis for embedded memories with 2-D redundancy,” *IEEE Trans. VLSI*, vol. 14, no. 1, pp. 31-42, Jan. 2006.

[57] S.-K. Lu, C.-L. Yang, Y.-C. Hsiao, and C.-W. Wu, “Efficient BISR techniques for embedded memories considering cluster faults,” *IEEE Trans. VLSI*, vol. 18, no. 2, pp. 184-193, Feb. 2010.

[58] I. Kim, et al., "Built in self repair for embedded high density SRAM,", IEEE Int. Test Conference, 1998.

[59] Y. Zorian, "Embedded memory test and repair: infrastructure IP for SOC yield," , IEEE Int.Test Conference , 2002.

[60] A. González, F. Latorre, and G. Magklis, "Processor microarchitecture: An implementation perspective," Synthesis Lectures on Computer Architecture 5.1, Morgan and Claypool eBooks, 2010.

[61] J. Pille, et al., "A 32kB 2R/1W L1 data cache in 45nm SOI technology for the POWER7TM processor," IEEE Int. Solid-State Circuits Conf., 2010.

[62] L. Denq and C. Wu, "A Hybrid BIST Scheme for Multiple Heterogeneous Embedded Memories," IEEE Asian VLSI Test Symp., 2007.

[63] X. Vera, et al. "Dynamically estimating lifetime of a semiconductor device." U.S. Patent No. 8,151,094. 3 Apr. 2012.

[64] M. Jung. "Low power and reliable design methodologies for 3D ICs." (2014).

[65] G. Apostolidis, D. Balobas, and N. Konofaos, “Design and simulation of 6T SRAM cell architectures in 32nm technology,” PACET 2015.

[66] W. Kim, C.-C. Chen, D.-H. Kim, and L. Milor, "Built in self test methodology with statistical analysis for electrical diagnosis of wearout in a static random access memory array," *IEEE Trans. VLSI.*

[67] W. Kim, S. Cha, and L. Milor, “Memory BIST for On-Chip Monitoring of Resistive-Open Defects due to Electromigration and Stress-Induced Voiding in an SRAM Array,” Proc. Conf. on Design of Circuits and Integrated Systems, 2014.

[68] W. Kim, C.-C. Chen, S. Cha, and L. Milor, “MBIST and statistical hypothesis test for time dependent dielectric breakdowns due to GOBD vs. BTDDB in an SRAM array,” Proc. IEEE VLSI Test Symposium, 2015.

[69] W. Kim, C.-C. Chen, and L. Milor, “Diagnosis of resistive-open defects due to electromigration and stress-induced voiding in an SRAM array,” Proc. International Integrated Reliability Workshop (IIRW), 2014.

[70] W. Kim and L. Milor, "Built-in self test methodology for diagnosis of backend wearout mechanisms in SRAM cells," Proc. VLSI Test Symposium., 2014.

[71] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 12, pp. 1859-1880, Dec 2005.

[72] B. K. Kannan, and S.N. Kramer, "An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design," *Journal of mechanical design*, vol. 116, no. 2, pp. 405-411, 1994.

[73] W. Kim, C-C. Chen, T. Liu, and S. Cha, "Estimation of remaining life using embedded SRAM for wearout parameter extraction," Proc. IEEE Int. Workshop on Advances in Sensors and Interfaces, 2015.

[74] W. Kim, C-C. Chen, T. Liu, and L. Milor, "Dynamically Monitoring System Health Using On-Chip Caches as a Wearout Sensor." *IEEE Trans. VLSI (submitted).*

[75] V.A. Vardanian and Y. Zorian, " A march-based fault location algorithm for static random access memories," Proc. of IEEE International On-Line Testing Workshop, 2002.

[76] T. Liu, C-C. Chen, W. Kim , and L. Milor, “Comprehensive reliability and aging analysis on SRAMs within microprocessor systems," *Microelectronics Reliability*, 2015.

[77] S. Drapatz, G. Georgakos, and D. Schmitt-Landsiedel, “Impact of negative and positive bias temperature stress on 6T-SRAM cells,” *Advances in Radio Science*, vol. 7, pp. 191-196, 2009.

[78] A. Bansal, R. Rao, J. Kim, S. Zafar, J. Stathis, and C. Chuang, “Impacts of NBTI and PBTI on SRAM static/dynamic noise margins and cell failure probability,” *Microelectronics Reliability*, vol. 49, no. 6, pp. 642-649, 2009.

[79] S. Cha, C.-C. Chen, and L. Milor, "System-level estimation of threshold voltage degradation due to NBTI with I/O measuremetns," Proc. IEEE Int. Reliablity Physics Symp., 2014.

[80] U. Kang, et al., "8 Gb 3-D DDR3 DRAM using through-silicon-via technology," *IEEE Journal of Solid-State circuits*, vol. 45, no. 1, pp. 111-119, December 2010.

[81] G. Loh, "3D-stacked memory architectures for multi-core processors," *ACM SIGARCH computer architecture news*, vol. 36, no. 3, pp. 453-464, Jun. 2008.

[82] M. Jung, Z.P. David, and S. Lim, "Chip/package co-analysis of thermo-mechanical stress and reliability in TSV-based 3D ICs," Proc. Design Automation Conference, 2012.

[83] M. Nakamoto, et al., "Simulation methodology and flow integration for 3D IC stress management," Proc. IEEE Custom Integrated Circuits Conference, 2010.

[84] X. Liu, et al., "Failure mechanisms and optimum design for electroplated copper through-silicon vias (TSV)," Proc. IEEE Electronic Components and Technology Conference, 2009.

[85] S.-K Ryu, K.-H Lu, X. Zhang, J. Im, P. Ho, and R. Huang, "Impact of near-surface thermal stresses on interfacial reliability of through-silicon vias for 3-D interconnects," *IEEE Trans. Device and Materials Reliability*, vol. 11, no. 1, pp. 35-43, Aug. 2010.

[86] S.R. Vempati, et al., "Development of 3-D silicon die stacked package using flip chip technology with micro bump interconnects," Proc. IEEE Electronic Components and Technology Conference, 2009.

[87] J. Yang, K. Athikulwongse, Y. Lee, S. Lim, and D.Z. Pan. "TSV stress aware timing analysis with applications to 3D-IC layout optimization." Proc. Design Automation Conference, 2010.

[88] L. Milor, "A Survey of Yield Modeling and Yield Enhancement Methods," *IEEE Trans. Semiconductor Manufacturing*, vol. 26, no. 2, pp. 196-213, May. 2013.

[89] Y. Zenda, K. Nakamae, and H. Fujioka, "Cost optimum embedded DRAM design by yield analysis," IEEE International Workshop on Memory Technology, Design and Testing, 2003.

[90] Y. Fei, P. Simon, and W. Maly, "New yield models for DSM manufacturing," IEEE Electron Devices Meeting, 2000.

[91] T.S. Barnett, A.D. Singh, and V.P. Nelson, "Extending integrated-circuit yield-models to estimate early-life reliability," *IEEE Trans. Reliability*, vol. 52, no.3, pp. 296-300, Sep. 2003.

[92] C. Hess and L.H. Weiland, "Extraction of wafer-level defect density distributions to improve yield prediction," *IEEE Trans. Semiconductor Manufacturing*, vol. 12, no. 2, pp. 175-183, May. 1999.

[93] R.B. Miller and W.C. Riordan, "Unit level predicted yield: a method of identifying high defect density die at wafer sort," Proc. IEEE International Test Conference, 2001.

[94] T.N. Barbour, T.S. Barne, M.S. Grady, and K.G. Purdy, "Method of statistical binning for reliability selection," U.S. Patent No. 6,789,032. 7 Sep. 2004.