International Journal of Engineering & Technology, 7 (3.1) (2018) 98-100

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Brief Review for Semiconductor Memory Testing Based on BIST Techniques

Nongthombam Imocha Singh[1]*, Prashant V. Joshi[2]

[1,2]School of ECE, Reva University, Bangalore-64, India.
**Corresponding author E-mail: imochamm@yahoo.com*

Abstract

With rapid growth of semiconductor industry and increase in complexity of semiconductor based memory, necessity of stringent testing methodology has become one of top most criteria for memory evaluation. This paper describes the fundamental concepts and overview of Built-In-Self-Test (BIST). It describes different functional faults modeling of RAM and flash memory. This review mentions about testing approaches for memory and illustrates BIST techniques for finding faults, power dissipation, area overhead and test time during testing, also includes research gap and future scope regarding the testing of memory.

Keywords: *BIST, fault functional modeling, flash memory, RAM, and test algorithm*.

1. Introduction

For any VLSI system, testing is important as it detects defects or faults. To verify the correctness of hardware, testing is required. With the rapid development in semiconductor technology, the complexity of VLSI system increases exponentially [1, 2]. Unfortunately, increase in circuit complexity also leads to complication in testing. Several challenges during memory testing have to be faced by design engineers with respect to power dissipation, fault coverage, testing time, cost and area overhead [3, 4, 5, 6]. Physical testing for highly advanced ICs is not possible to test by using Automatic test equipment. To overcome these problems, Built-In-Self-Test (BIST) method is introduced. By using BIST circuitry, testing cost is reducing and also testing time is also reduced. Semiconductor memories such as SRAM, DRAM, ROM, EPROM, EEPROM and Flash memory etc are the most vulnerable devices to process defects. Based on the International Technology Roadmap for Semiconductors, the percentage of memory in the CPU cores and consumer core are increasing rapidly. So, testing of semiconductor memories is very important.

This review is divided into four major sections. The introduction is presented in this section (Section - I). Section - II describes the concept of BIST. RAM and FLASH memory testing are covered in Section - III and IV respectively. Conclusion and future scope are mentioned in Section - V.

2. Concept of BIST

According to [7], “BIST is a design-for-test technique in which testing is accomplished through built-in-hardware features”. In BIST technique, additional circuits are added on the chip to test itself. The difficulty and complexity of VLSI testing is reduced effectively by using these techniques. The correctness of the design function and fabrication process is verified by the testing. Moreover, testing may also be implemented during the latter periods of the product lifecycle for detecting errors due to aging, environment or other factors.
BIST techniques in memory can be classified as

- On-line BIST
- Off-line BIST

In On-line BIST, testing can be done during the normal functional operating states. This testing takes shorter time but has little area overhead. Again, there are three types of On-line BIST:
Concurrent BIST: In On-line Concurrent BIST, testing can be done concurrently with the normal functional operation. As a result, errors can be detected and corrected instantly depending on the method used. On-line Non-Concurrent BIST is a form of
Non-Concurrent BIST: On-line Non-Concurrent BIST is a form of testing when the system is in idle mode. So, interruption of the normal function is required in Non-Current BIST techniques.
Off-line BIST is the form of testing that occurs when the system is not operated in its normal function. This testing takes longer time but has no area overhead.
The basic architecture of BIST consists of test pattern generator (TPG), circuit-under-test (CUT) and output response analyser (ORA). Fig. 1 shows the basic architecture of BIST. TPG which may be pseudo random numbers or deterministic sequence generates test pattern to address the location on CUT. ORA will collect and verify the output from CUT and give the result as pass or fail. BIST Controller controls all the operation during testing.

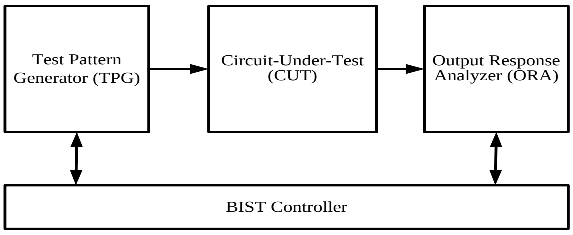

Fig. 1: Basic architecture of BIST

3. RAM Memory Testing

RAM is a volatile memory in which data can be accessed randomly. Generally it consists of read and writes logic, address decoder and array of memory cells. Fig 2 shows the typical RAM BIST architecture. It consists of BIST module, test collar (mux) and circuit-under-test memory. For handling multiple clocks or signals, test collar is used. BIST module consists of controller, pattern generator and comparator. The flow in BIST module is controlled by BIST algorithm.

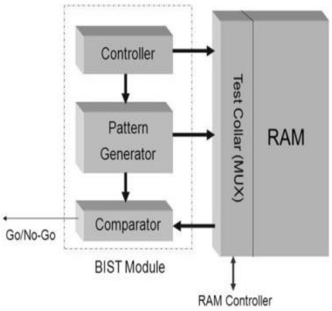

Fig. 2: Typical RAM BIST architecture

Many physical failures may occur in the read and write logic, address decoder and array of memory cells which will cause several failures in the memory function. Their causes are depended on such factors as circuit layout, component density, manufacturing method etc. Many fault models have been introduced to capture the effects of these failures in RAM.

3.1. RAM Functional Faults Model

1. Stuck-at fault (SAF): Sticking of either a cell or a line to logical value ‘1’ or ‘0’.
2. Stuck open fault (SOF): If a cell cannot be accessible due to broken line, then SOF occurs.
3. Transition fault (TF): Failing of either a cell or a line to transit from 1 to 0 or 0 to 1.
4. Coupling fault (CF): Changing of the content of a cell by the write operation of another cell. There are three types of CF:
   (a) Inversion coupling fault (CFin): A transition from ‘1’ to ‘0’ or ‘0’ to ‘1’ in one cell (coupling cell) inverts the content of other cell.
   (b) Idempotent coupling fault (CFid): A transition from logic ‘1’ to ‘0’ or ‘0’ to ‘1’ in one cell (coupling cell) forces the content of other cell to a particular value, ‘0’ or ‘1’.
   (c) State coupling fault (CFsts): Forcing of a cell to stay in particular state only when the coupling cell/line is in a given state.
5. Neighborhood pattern sensitive fault (NPSF): If the contents of some other cells in the memory influence the content of a cell or the ability to change its content, then NPSF occurs.
6. Address fault: In this condition, a cell will not be address with a certain address or simultaneously multiple cells are accessed with a particular address or multiple addresses access a particular cell or a particular cell is never accessed.

3.2. RAM Test Algorithms

The functional test of a RAM is mostly operated by read and writes operation. Many algorithms have been developed for several fault models and different test structures.

These RAM test algorithms are has two types: Classical test and March-based tests. Classical test algorithms are Checkerboard, Walking, Butterfly, etc. Each algorithm has its own merit and demerits. March-based Test Algorithm has finite sequence of march elements. This element is specified by memory operation such as read operation or write operation, data pattern and address sequence such as ascending or decreasing order. Examples of some March-based tests are MATS, MATS+, March X, March C, March C-, March Y etc. For example, ↑ (rb1, wb0) is a March element and March primitives are rb1 and wb0 [14]. wb means write operation and rb means read operation. (↑) denotes increasing order of address. (↓) denotes decreasing order of address. (↕) denotes either increasing or decreasing according to the type of algorithms. An operation can be either writing a zero or one to a cell (wb0 or wb1), or reading a zero or one from a cell (rb0 or rb1). Table 1 shows the different March Test Algorithms and table 2 shows the faults detection by different RAM test algorithms. By implementing one of these algorithms in the memory, power, area overhead and testing time can be calculated.

Table 1: RAM Test Algorithms [15, 16]

Sl. No.	Algorithm	March elements
1	MATS	{↕(wb0); ↑(rb0, wb1); ↓(rb1)}
2	MATS++	{↕(wb0); ↑(rb0, w1); ↓(rb1, wb0)}
3	March X	{↕(wb0); ↑(rb0, wb1); ↓(rb1, wb0); ↕(rb0)}
4	March C	{↕(wb0); ↑(rb0, wb1); ↑(rb1, wb0); ↓(rb0, wb1); ↓(rb1, wb0); ↕(rb0); ↓ (rb0, wb1); ↑(rb1, wb0); ↕(rb0)}
5	March C-	{↕(wb0); ↑(rb0, wb1); ↑(rb1, wb0); ↓(rb0, wb1); ↓(rb1, wb0); ↕(rb0)}
6	March B	{↕(wb0); ↑(rb0, wb1, rb1, wb0, rb0, wb1); ↑(rb1, wb0, wb1); ↓(rb1, wb0, wb1, wb0); ↓(rb0, wb1, wb0)}
7	March U	{↕(wb0); ↑(rb0, wb1, rb1, wb0); ↑(rb0, wb1); ↓(rb1, wb0, wb1, rb0, wb1); ↓(rb1, wb0)}
8	March LR	{↕(wb0); ↓(rb0, wb1); ↑(rb1, wb0, rb0, wb1); ↑(rb1, wb0); ↑(rb0, wb1, rb1, wb0); ↑(rb0)}

Table 2: Detection of Faults by Several RAM Test Algorithms

Algorithm	FAULTS				Complexity
	SAF	AF	TF	CF	
Checkerboard	D	-	-	-	4N
GALPAT	D	D	D	D	$4N^2$
BUTTERFLY	D	D	-	-	5N log N
MATS	D	-	-	-	4N
MATS+	D	D	-	-	5N
Marching 1/0	D	D	D	-	14N
MATS++	D	D	D	-	6N
March X	D	D	D	-	6N
March C	D	D	D	D	11N
March C-	D	D	D	D	10N

Here in the above table, ‘D’ means all detected; ‘-’ means not all detected; ‘N’ represents the number of cells.

4. Flash Memory Testing

Flash memory is a non-volatile memory which is based on floating-gate (FG) transistor [8, 9]. The state of memory in flash memory to be high (logic 1) or low (logic 0) is operated by charging or discharging the FG by applying high voltage. Flash memory performs read, program and erase operations. Fig 3 shows the flash memory BIST architecture. It consists of controller (CTR), test pattern generator (TPG) and MUX (test collar). The output signals from the MUX goes into the flash memory for testing.

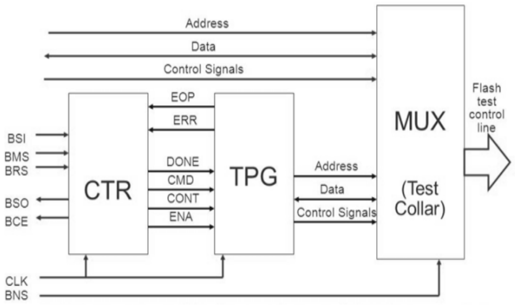

Fig. 3: Flash memory BIST architecture

Several challenges also occur with the development of new advance semiconductor technology and thereby different faults are existed.

Thus, different functional fault model of flash memory are as followed [11]:

1. Word-line Program Disturbance (WPD): Programming of a cell if another cell on that word-line is happened to program.
2. Word-line Erase Disturbance (WED): Erasing of a cell if another cell on that word-line is happened to program.
3. Bit-line Program Disturbance (BPD): Programming of a cell if another cell on that bit-line is happened to program.
4. Bit-line Erase Disturbance (BED): Erasing of a cell if another cell on that bit-line is happened to program..
5. Over Erase (OE): Reduction of cell threshold voltage and bit-line leakage if that cell is overly erased.
6. Read Disturbance (RD): Changing of threshold voltage for a particular cell, when that cell is frequently because the bias conditions for programming are same as for reading operation.

Functional fault models such as stuck-at fault (SAF), transition fault (TF), state-coupling fault (CFst), address decoder fault (AF), and stuck-open fault (SOF) which are occur in RAM, also occur in flash memory[10]

4.1. Bit-Oriented Test Algorithm

Traditional March tests cannot detect most of flash specific faults [2]. March FT [12] algorithm for flash memory testing:

{(f); ↓ (r1, w0, r0); ↕ (r0); (f); ↑ (r1, w0, r0); (r0); ↕ (r0)}Where f denotes erase/flash; w0 means program; r1 or r0 denotes read 1 or read 0

4.2. Word-Oriented Test Algorithm

Intra-word faults are occurred in Word-oriented memory. For testing intra-word faults in flash memory, simple test with multiple standard backgrounds are added. In this case, background is a multiple bit operation.

The algorithm for intra-word faults in flash memory is

{(f); ↕ (waa, raa); (f); ↕ (wbb,rbb)}
Where f denotes erase or flash; w0 means program; r1 or r0 denotes read 1 or read 0 ‘aa’ and ‘bb’ are background. For this algorithm, number of backgrounds is log2(m) +1
where ‘m’ denotes word width.

5. Conclusion and Future Scope

As semiconductor technology has grown exponentially and 3D layered memory architecture has evolved rapidly. New faults are observed due to complexity in technology. As power dissipation in testing mode is more than the normal mode [13], reduction of power dissipation is also important during test mode. It is seen that testing time, cost, area overhead and delay of the memory BIST architecture increases with new evolving technology. But, all the existing algorithms are not stringent enough to find possible faults which may be present due to fabrication errors or environmental disturbance [3]. Regarding the above challenges, there is a need to design efficient BIST algorithm considering power dissipation, area overhead, testing time and high fault coverage. This review helps us in building a way for future scope.

Acknowledgement

The authors thank to Reva University for providing a platform for the research.

References

[1] Jen-Cheng Ying, Wang-Dauh Tseng, Wen-Jiin Tsai, “Asymmetry dual-LFSR reseeding for low power BIST”, Integration, the VLSI journal, Elsevier, vol. 60, 2018, pp. 272-276.

[2] The international technology roadmap for semiconductors, 2005 edition (ITRS 2005), Semiconductor Industry Association, 2005. Available from: <http:// public.itrs.net/>.

[3] Aditya Kumar Singh Pundir and Om Prakash Sharma, “CHECKER-MARC: A Modified Novel Memory-Testing Approach for Bit-Oriented SRAM”, International Journal of Applied Engineering Research, vol. 12, no. 12, 2017, pp. 3023-3028.

[4] Michał Filipek, Grzegorz Mrugalski, Nilanjan Mukherjee, Benoit Nadeau-Dostie, Janusz Rajski, Jedrzej Solecki, and Jerzy Tyszer, “Low-Power Programmable PRPG With Test Compression Capabilities”, IEEE Transactions On Very Large Scale Integration (VLSI) Systems, vol. 23, no. 6, Jun. 2015, pp. 1063-1076.

[5] Feng Liang, Luwen Zhang, Shaochong Lei, Guohe Zhang, Kaile Gao, and Bin Liang, “Test Patterns of Multiple SIC Vectors: Theory and Application in BIST Schemes”, IEEE Transactions On Very Large Scale Integration (VLSI) Systems, vol. 21, no.4, April 2013 , pp 614-623.

[6] Gert Jervan, Elmet Orasson, Helena Kruus, Raimund Ubar, “Hybrid BIST optimization using reseeding and test set compaction”, Microprocessors and Microsystems, vol. 32, no. 5-6, 2008, , pp. 254-262.

[7] V. Agrawal, C. Kime, K. Saluja, A tutorial on built-in self test, in: IEEE Design & Test of Computers, IEEE Computer Soc. Press, Silver Spring, MD, March 1993, pp. 73-80 and June 1993, pp. 69-77.

[8] S. Aritome, R. Shirota, G. Hemink, T. Endoh, F. Mausouka, “Reliability Issues of Flash Memory Cells”, Proc. of the IEEE, vol. 81, no. 5, May 1993, pp. 776-788.

[9] P. Pavan, R. Bez, P. Olivo, E. Zanoni, “Flash Memory Cells - An Overview”, Proc. of the IEEE, vol. 85, no. 8, Aug, pp. 1248-1271, 1997.

[10] A.J. van de Goor, “Testing semiconductor memories: Theory and Practice”, John Wiley & Sons, Chichester England, 1991.

[11] Martirosyan, S., G. Harutyunyan, S. Shoukourian, and Y. Zorian, “An efficient testing methodology for embedded flash memories”, In East-West Design & Test Symposium (EWDTS), IEEE, 2017, pp. 1-4.

[12] J.-C. Yeh, C.-F. Wu, K.-L. Cheng, Y.-F. Chou, C.-T. Huang, C.-W. Wu, “Flash memory built-in self-test using march-like algorithms”, Proc. Int. Workshop on Electronic Design Test and Applications (DELTA), Jan 2002, pp. 137-141.

[13] Y. Zorian, “A Distributed BIST Control Scheme for Complex VLSI Devices”, Proc. VLSI Test Symp., 1993, pp. 4-9.

[14] Balwinder Singh, Sukhleen Bindra Narang, Arun Khosla, “Modeling and Simulation of Efficient March Algorithm for Memory Testing”, In International Conference on Contemporary Computing. Springer Berlin Heidelberg, 2010, pp. 96-107.

[15] Kuo-Liang Cheng, Chia-Ming Hsueh, Jing-Reng Huang, Jen-ChiehYeh, Chih-Tsun Huang, and Cheng-Wen Wu, Automatic Generation of Memory Built-In Self -Test Cores for System on Chip. IEEE conference 2001; (1) pp. 91-96.

[16] Viswani D Agarwal and Michael L Bushnell, Essentials of Electronic Testing of Digital, Memory and Mixed Signal VLSI Circuits. 3rd ed. New York: Kluwer Academic Publishers; 2000.

[17] Author,”Title of the Paper”, *Proceedings of the conference name*, Vol.X, No.X, (200X), pp:XX-YY, http://dx.doi.org/10.1109/MMM.2013.2248651.

[18] G. Ramprabu, S. Nagarajan, “Design and Analysis of Novel Modified Cross Layer Controller for WMSN”, Indian Journal of Science and Technology, Vol 8(5), March 2015, pp.438-444.