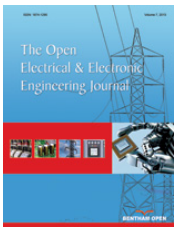
Send Orders for Reprints to reprints@benthamscience.ae

The Open Electrical & Electronic Engineering Journal, 2016, *10*, 1-10

The Open Electrical & Electronic Engineering Journal

Content list available at: www.benthamopen.com/TOEEJ/

DOI: 10.2174/1874129001610010001



A Novel Controllable BIST Circuit for embedded SRAM

Zhiting Lin, Chunyu Peng* and Kun Wang

School of Electronics and Information Engineering, Anhui University, Peoples Republic of China

Abstract: With increasingly stringent requirements for memory test, the complexity of the test algorithm is increasing. This will make BIST (Build-In-Self-Test) circuit more complex and the area of BIST circuit larger. This paper proposes a novel controllable BIST circuit. The controllable BIST circuit provides a cost-effective solution that supports a variety of March algorithms and SRAM embedded testing operation modes. It controls the test patterns with three additional input ports. And it indicates the algorithm progress, the test result and the number of fails with three output ports. To achieve test patterns generation, analy-sis and test results recording, the proposed BIST circuit contains five internal functional modules, which are Address Gener-ator, Control Generator, Data Generator, Data Comparator and Fail Accumulator. The test patterns of the proposed BIST cir-cuit are controlled by external signals. It is not only suitable for any existing march algorithms but also leaves room for ex-tension if needed.

Keywords: Controllable BIST, Hardware overhead, March algorithm, SRAM.

1. INTRODUCTION

As the size of semiconductor SRAMs becomes larger and larger, it's more and more difficult and expensive to test the memories [1, 2]. BIST is a widely used method for the detection of manufacturing defects and operational faults in DFT (Design for Testability) of memory to enhance the yield [3, 4]. BIST technology transfers external test to inside by realizing test modules, such as the test pattern generator, BIST controller and comparator.They canwrite the test pattern into the memory, read data from the memory and compare with the expected data. The basic structure of BIST circuit is shown in (Fig. **1**). Integrating the BIST circuit and the circuit under test on one chip can not only eliminate the test equipment for external test but also make the test at the same clock speed as operating state. Thus, it saves the test cost and improves the quality of the test. However, this structure also has shortcomings that mainly reflected in increasing the area and affecting timing characteristics of chips.

BIST has been extensively explored and utilized especially in the design of embedded memories as it eases the process of testing as well as enablea very complex embedded memories being tested thoroughly and efficiently [5]. The March algorithms have been widely used to test memory chips and found to be very effective for fault detection and diagnosis in memory tests [6].

However, due to the structures of memories are more diverse, and the capacitance and density of memories are increasing, variety of faults emergein the process of manufacturing.So the cost of testing memories increases rapidly with every new generation of memory chips [7]. People have been exhausting a lot of researches on March algorithms and proposing various new March algorithms special for emerging faults to test the embedded memories effectively [8 - 11]. In order to accurately detect various faults, the complexity of new test algorithms is increasing. For example, the complexity of March LSD algorithm in [12] is 75N and the complexity of March MD2 algorithm in [13] is 70N. The high complexity algorithms make the BIST circuit area grow and impact the timing performance. In addition, even though new march algorithms can test against one or several types of faults, most of these algorithms are based on the hard-wired BIST approach which is fixed once the design is completed. Once the BIST circuit is generated, it cannot detect more other faults.

* Address correspondence to this author at the School of Electronics and Information Engineering, Anhui University, Peoples Republic of China; E-mail: ztlin@mail.ustc.edu.cn

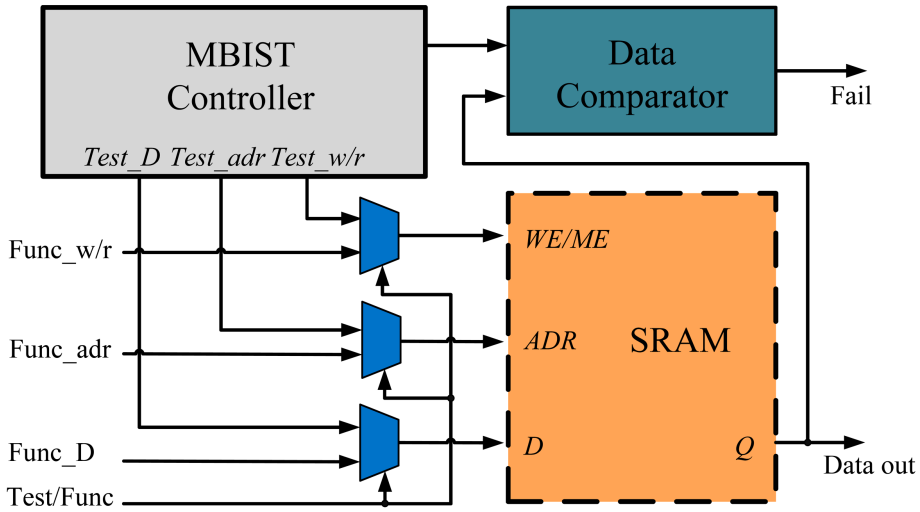


Fig. (1). Basic structure of conventional BIST.

This lacks the flexibility of design debug, especially in the early phase of the design [14, 15]. What's more, a design with multiple memories requires embedding multiple BIST circuits into the chip as shown in (Fig. **2**). This will result in greater area overhead and timing derating.

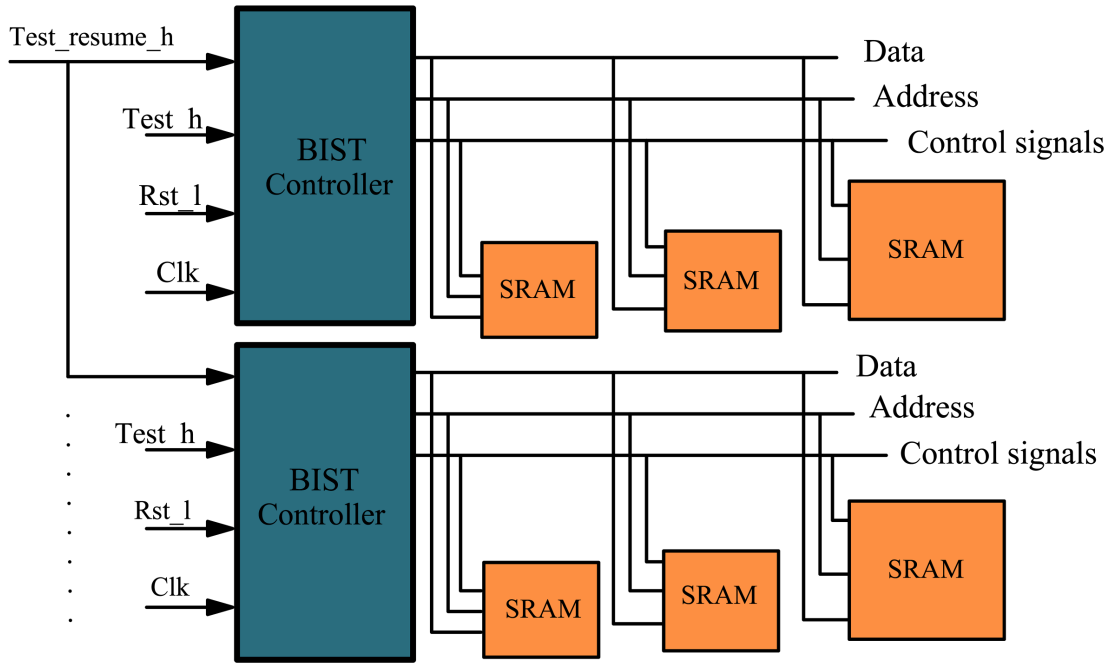


Fig. (2). Multi BISTs multi memories scheme.

These shortcomings of conventional BIST circuits limit the fault coverage capabilities. This paper proposes a novel controllable BIST circuit. The controllable BIST circuit supports a variety of March algorithms. The circuit structure of the proposed BIST circuit is different from conventional BIST circuits. It has control signals from external input ports to minimize the chip area, and supports a wide variety of test patterns.

Unlike conventional BIST circuits which require high hardware overhead, the proposed controllable BIST circuit greatly simplifies the internal test pattern generation circuit, reduces the area of BIST circuit, and improves the timing character.

2. MATERIALS AND METHODS

2.1. Controllable BIST Circuit Architecture

The Controllable BIST circuit can be divided into address generator, control generator, data comparator and fail accumulator. Specific module structure is shown in (Fig. **3**). The features of each module are as follows:

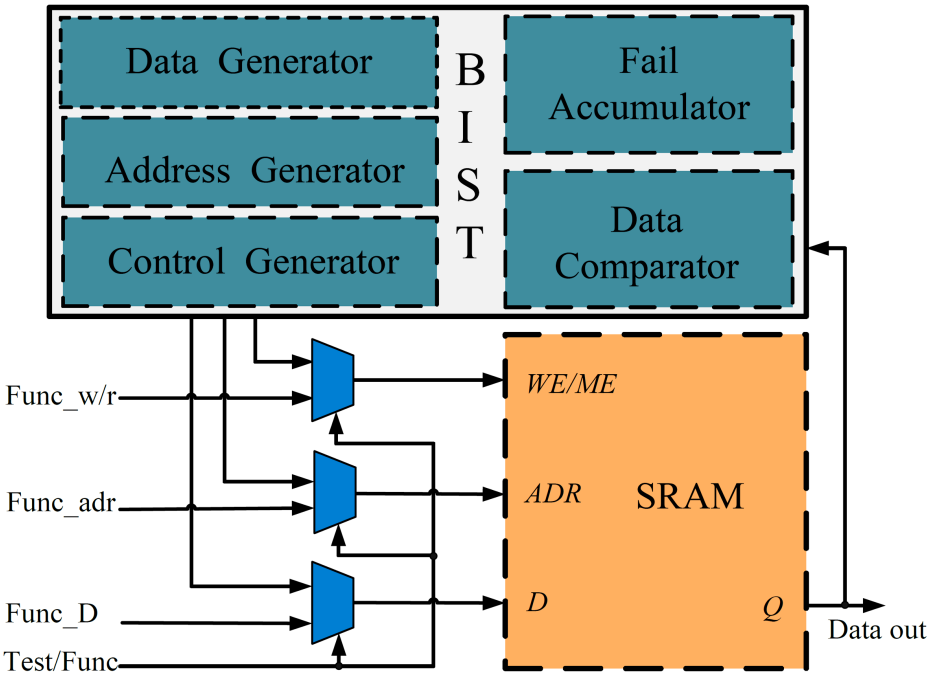


Fig. (3). Major construct of the controllable BIST circuit.

Address Generator: generate address signals for memory tests. The address generator of the proposed BIST circuit is designed to control the modes of address transformation. Ascending or descending is controlled by test_adr signals according to specific algorithm. In addition, the address generator also provides a step_done signal. When the address transformation is complete, the step_done signal is asserted to indicate that all memory units complete the read and write operations.

Control Generator: provide control signals for memory tests, including memory read and write control signals. It is controlled by the input test_we.

Data Generator: generate memory write data and BIST comparator data. The generated data is determined by the test_D signal.

Data Comparator: compare the data read from a memory and the data written before. If the comparison between the two data is consistent, it returns 0, otherwise it returns 1.

Fail Accumulator: when a fail appearance, bist_fail signal active. At the same time, it records the number of fails in a memory according to the comparison result of the data comparator, and exports a fault_num signal to show the number of fails.

2.2. Work flow of the Controllable BIST Circuit

Fig. (**4**) shows the work flow of the controllable BIST circuit. The operation of SRAM is decided by its inputs ADR, WE and D.

Address Generator generates address signals ADR for SRAM under testing. During testing, if test_adr 0 =0, the address of SRAM will change ascendingly, and if test_adr[0]=1, it will change descendingly. At the rising edge of clock signal CLK, when test_adr[1]=0, the address keeps unchanged, and when test_adr[1]=1, it changes according to specific address direction determined by test_adr[0]. When the address transformation is complete, the step_done signal is asserted. Data Generator provides write data D for SRAM. When test_D is set to 1, the data for input ports D are a

series of 1s (the number of 1s is determined by the specification of SRAM), and similarly when test_D is 0, the data are a series of 0s. Control Generator generates the read/write control signal WE. When test_we is set to 1, the WE signal of the SRAM is 1, *i.e.* the BIST circuit performs write operation to the SRAM, and otherwise it performs read operation. Data Comparator performs bitwise comparison between the data read from the ports Q of the SRAM and the data written to ports D of the SRAM. If the comparison result is correct, it returns a 0 to Fail Accumulator, else it returns a 1. Fail Accumulator will plus the output from Data Comparator to get the number of fails.

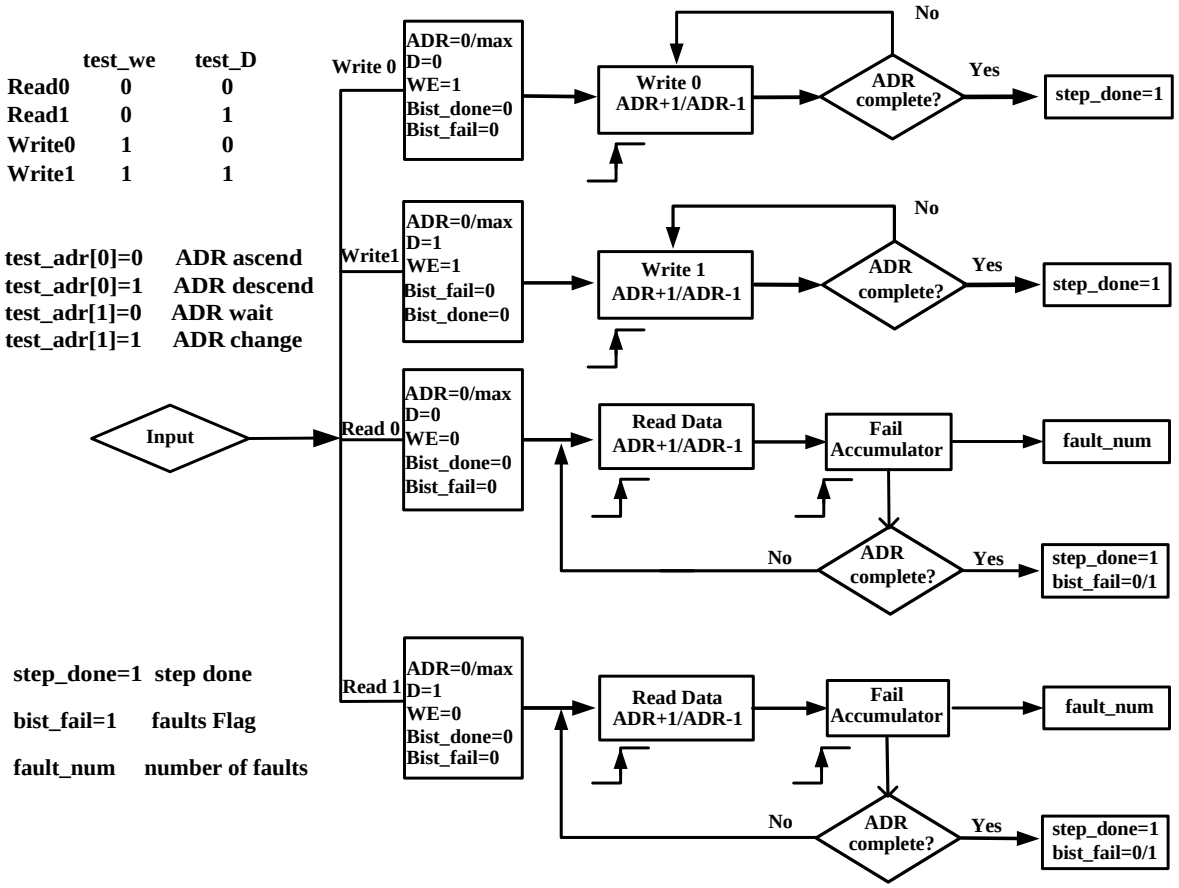


Fig. (4). Work flow of the BIST circuit.

The connection of BIST circuit and SRAM is shown in (Fig. **5**). We add three input ports test_adr, test_D and test_we to the BIST. The three signals control the address transformation, the testing data, and the operation selection respectively. We also add three output ports step_done, bist_fail and fault_num. The step_done signal is used to indicate the algorithm progress, the bist_fail is used to show that there are fails appearance and fault_num signal is used to count the number of fails.

2.3. Analysis

Without loss of generality, a march algorithm has *m* steps denoted by *S1*, *S2*, ..., and *Sm*, and each step has at most *k* read/write operations. It can be mathematically expressed as:

$$< \text{march algorithm} > \; = \{<Si> \mid \; i = 1, 2, ..., m\} \tag{1}$$

$$< Si > \; = \{ < \text{address order} > (< \text{opj} > \mid \; j \; = \; 1, \; ..., \; k)\} \tag{2}$$

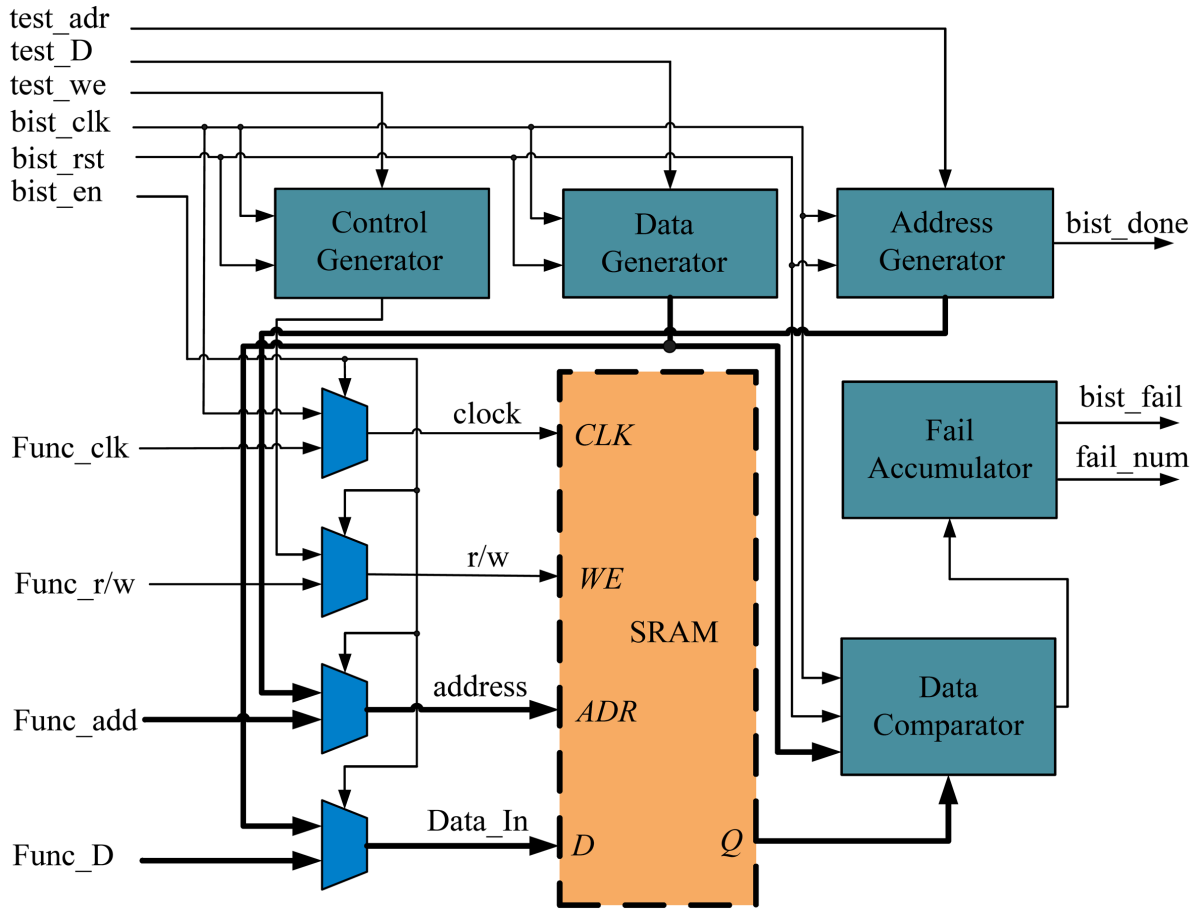


Fig. (5). Connection of BIST circuit and SRAM.

$$< \text{address order} > \in \{ \text{ascend}(\Uparrow), \text{descend}(\Downarrow), \text{arbitrary}(\Updownarrow) \} \quad \textbf{(3)}$$

$$< \text{opj} > = \{<\text{behavior}> <\text{data}> \mid \text{behavior} \in \{w, r\},\ \text{data} \in \{0,1\} \} \quad \textbf{(4)}$$

The conventional BIST hardware overhead (bits) is expressed as follows:

$$< \text{hardware overhead} >= (1+(1+[\log_2 d])*k)*[\log_2 adr]*m \quad \textbf{(5)}$$

, where the terms of "1", "1+$\log_2 d$" and "$\log_2 adr$" are the bit numbers required to save the "ascending/descending address order", "the sum of 1 bit for read/write operation and data length" and "address length"respectively,and the terms of "$\log_2 d$" and "$\log_2 adr$" are the bit numbers of data and address.

Because of the test patterns of proposed controllable BIST circuit are controlled by outside signals, which means the algorithms is determined by outside signals. It only retains test patterns generation circuits, and reduces the hardware overhead of test pattern storing circuits.The proposed controllable BIST hardware overhead (bits) is:

$$< \text{hardware overhead} >= (1+1+[\log_2 d])*[\log_2 adr] \quad \textbf{(6)}$$

It can be seen that the controllable BIST hardware overhead is only depend on the specification of SRAM, nothing to do with specific algorithms.

Take March1 algorithm on 8192x32 SRAM for example. There are six steps in the March1 algorithm as illustrated in (Table **1**). The conventional BIST hardware overhead of March1 algorithm is (1 + (1 + $\log_2$32)*2)*$\log_2$8192* 6 =1014 bits, while the controllable BIST hardware overhead is (1 + 1 + $\log_2$32) * $\log_2$8192 = 91 bits.

Table 1. The characteristics of March1 algorithm.

Step	S_1	S_2	S_3	S_4	S_4	S_6
Order	⇑	⇑	⇑	⇓	⇓	⇓
Operation	*w0*	*r0w1*	*r1w0*	*r0w1*	*r1w0*	*r0*
op length	1	2	2	2	2	1

Similarly, the hardware overhead of March LSD algorithm, whose algorithm complexity is 75N in [12], is (1 + (1 + $\log_2$32)) * 18) * $\log_2$8192 * 7 = 9919bits and the hardware overhead of March MD2 algorithm, whose algorithm complexity is 70N in [13], is (1 + (1 + $\log_2$32) * 17) * $\log_2$8192 *6 = 8034 bits. However, the hardware overhead of controllable BIST is still 91 bits.

If a design contains multiple BIST circuits, the hardware overhead in traditional structures is the sum of the overhead of every single BIST. But the proposed controllable BIST circuit supports a variety of March algorithms, so it need not embed multiple BIST circuits into a design. In other words, it sharply reduces the hardware overhead.

3. RESULTS AND DISCUSSION

Here is a simulation of the controllable BIST circuit implementing March1 algorithm on 8192x32 SRAM shown in (Fig. **6**). We can clearly see the six steps of March 1 algorithm. After completing every step, the step_done signal jump to 1 for a while. And after completing the March 1 algorithm, bist_done signal jump to 1.

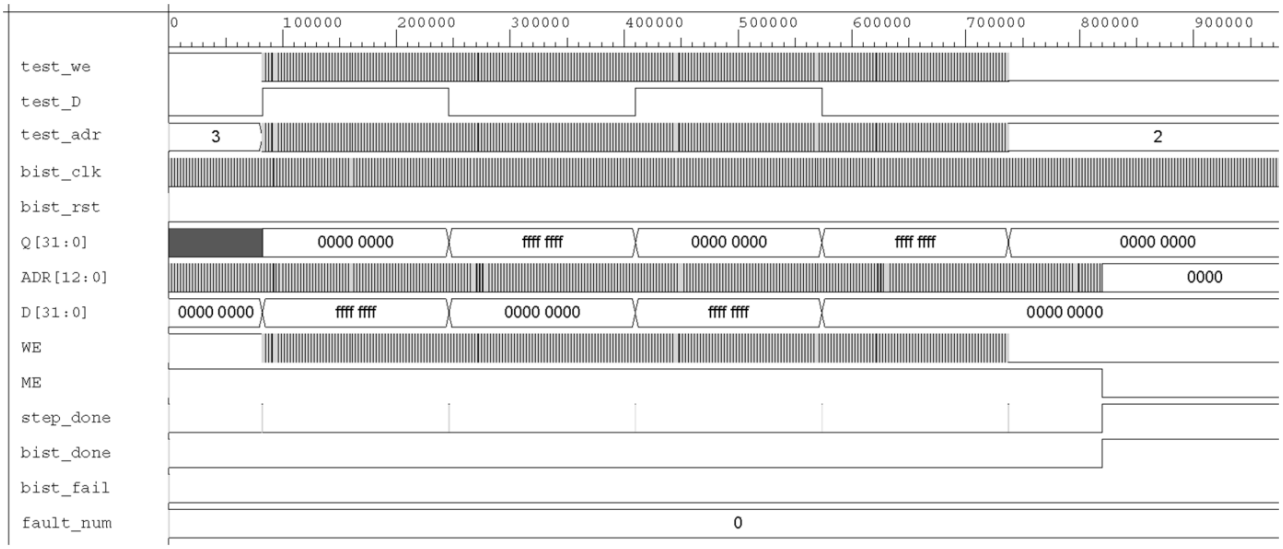
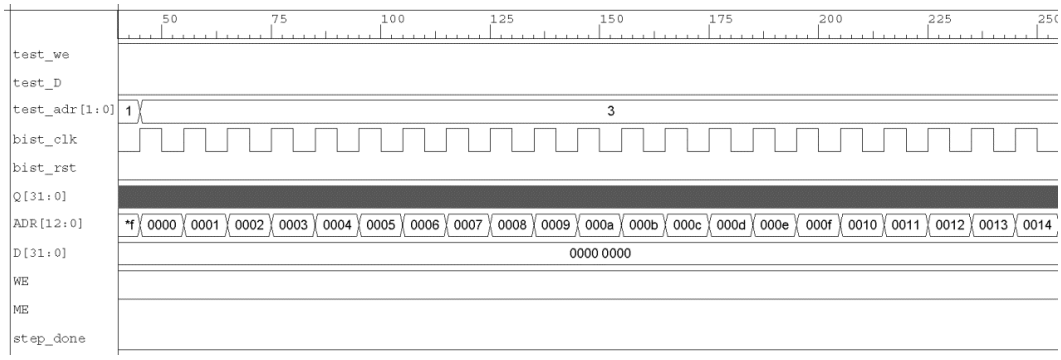


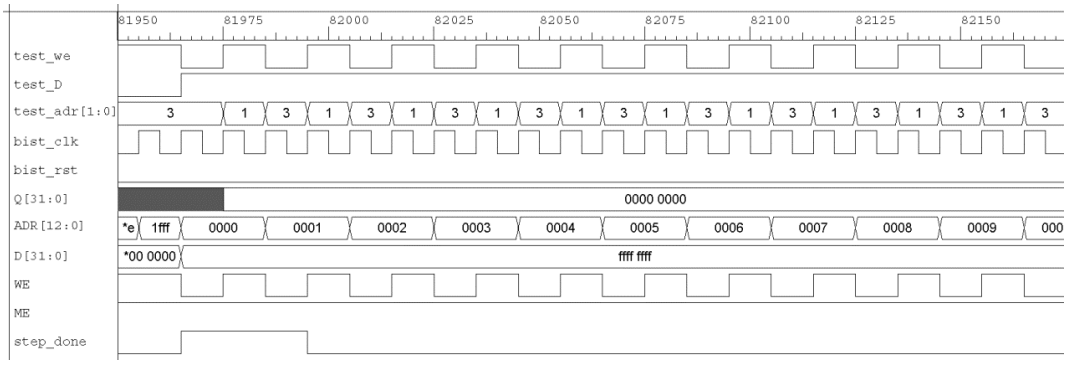
Fig. (6). March 1 algorithm on 8192x32 SRAM implemented by controllable BIST.

Fig. **7** respectively shows the six steps of March1 algorithm in detail. In Fig. **7** (a), WE signal always keeps at 1, D signals always keep at 32'h00000000, and ADR signals change ascendingly. At every address, when a positive edge of clock signal bist_clk arrives, it performs a write 0 operation to SRAM.

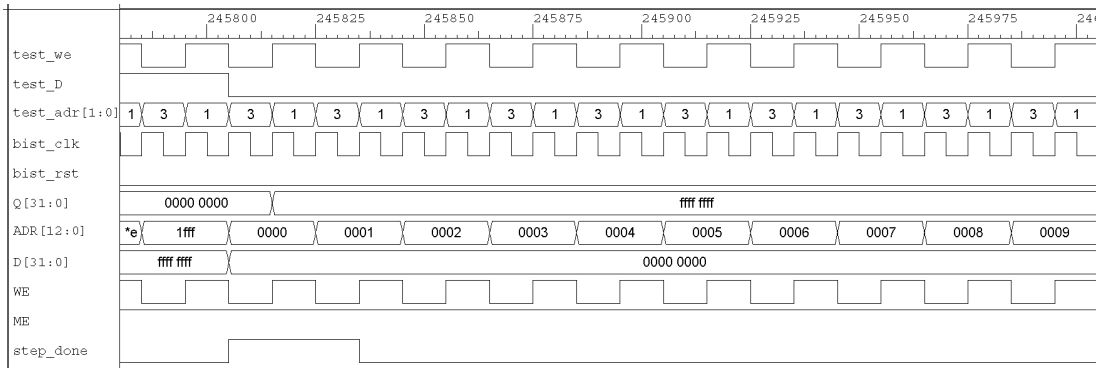
In Fig. **7** (b), D signals always keep at 32'hffffffff, the address signals ADR change ascendingly. At every address, the WE signal keeps at 0 for one clock period, and then keeps at 1 for one clock period. It performs a read operation to SRAM at the first positive edge of bist_clk signal, and then it performs a write 1opearation at the second positive edge of bist_clk signal.



(a) S1: ⇑ w0



(b) S2: ⇑ r0w1



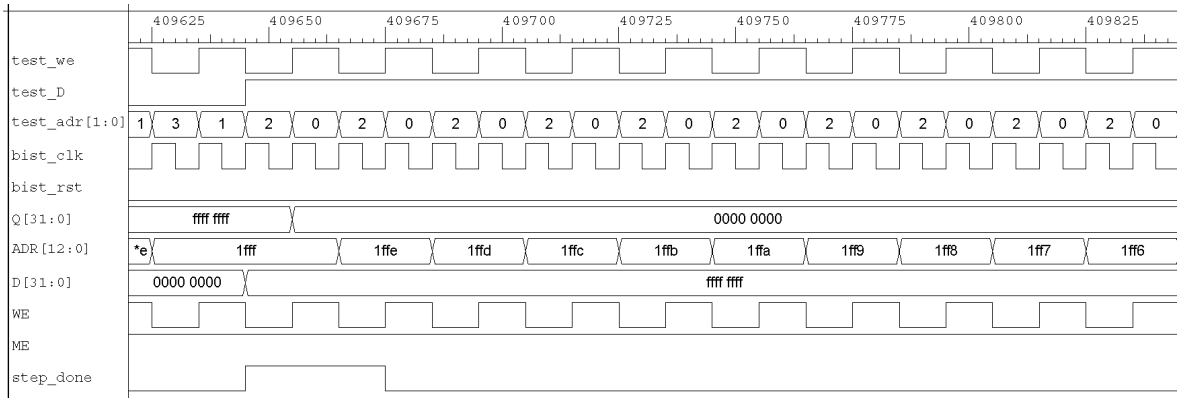
(c) S3: ⇑ r1w0

Similarly in Fig. **7** (c), address signals ADR change ascendingly. At every address, it firstly performs a read operation to SRAM, and then it performs a write 0 operation to SRAM.

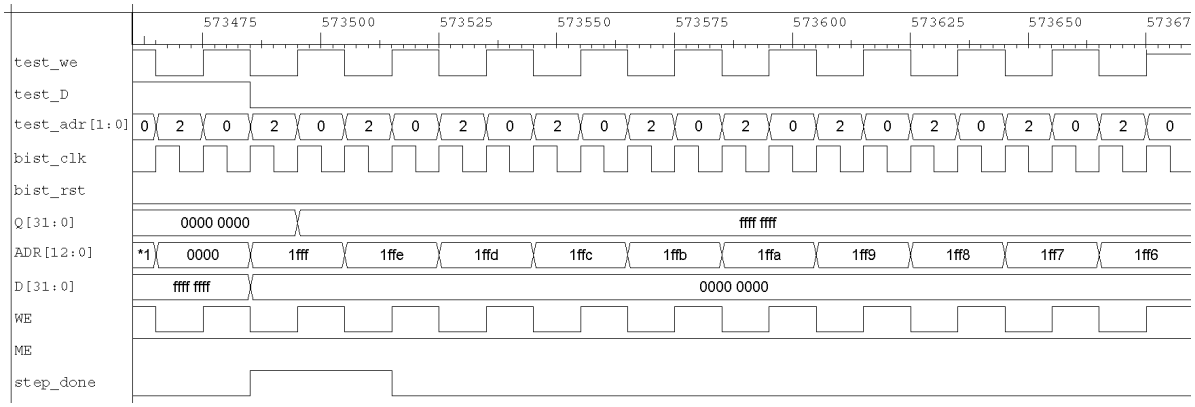
In Fig. **7** (d), D signals always keep at 32'hffffffff, the address signals ADR change descendingly. At every address, the WE signal keeps at 0 for one clock period, and then keeps at 1 for one clock period. It performs a read operation to SRAM at the first positive edge of bist_clk signal, and then it performs a write 1opearation at the second positive edge of bist_clk signal.

Similarly in Fig. **7** (e), address signals ADR changes descendingly. At every address, it firstly performs a read operation to SRAM, and then it performs a write 0 operation to SRAM.

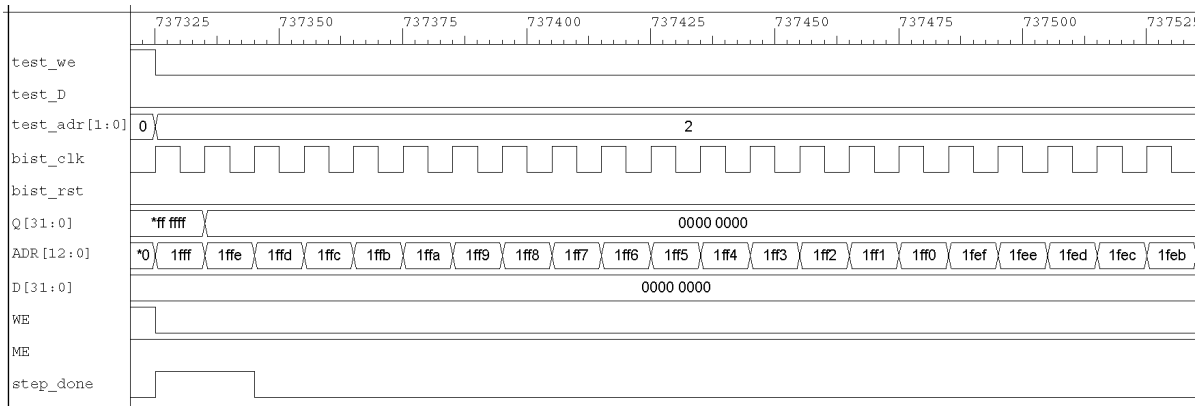
In Fig. **7** (f), WE signal always keeps at 0, and ADR signals change descendingly. At every address, when a positive edge of clock signal bist_clk arrives, it performs a read operation to SRAM.



(d) S4: ⇓ r0w1



(e) S5: ⇓ r1w0



(f) S6: ⇓ r0

Fig. (7). March1 algorithm implemented by the controllable BIST circuit.

When force the input ports Q sticking to 32'h00000001, the controllable BIST circuit detected the faults at every read operation. Once the controllable BIST detects faults, the bist_fail signal will jump to 1. After completing the test, bist_done signal jumps to 1, and the total number of faults will be 1 * 8192 * 3+31 * 8192 * 2 = 532480. As can be seen in Fig. **8**, bist_fail signal jumps to 1 at first read operation, and bist_done signal jumps to 1 at the end of the test. The total faults number is 532480 as illustrated in (Fig. **8**).

When the address ports of SRAM stick to 13'b00000000000, the controllable BIST circuit only performs read and write operation at ADR=13'b0000000000000,as shown in (Fig. **9**). After completing the test, bist_done signal jumps to 1, and the total number of faults will be 31 * 8192 * 4= 1048448. Thesignal bist_fail jumps to 1 at first read operation, and bist_done signal jumps to 1 at the end of the test. The total faults number is 1048448 as illustrated in (Fig. **9**).

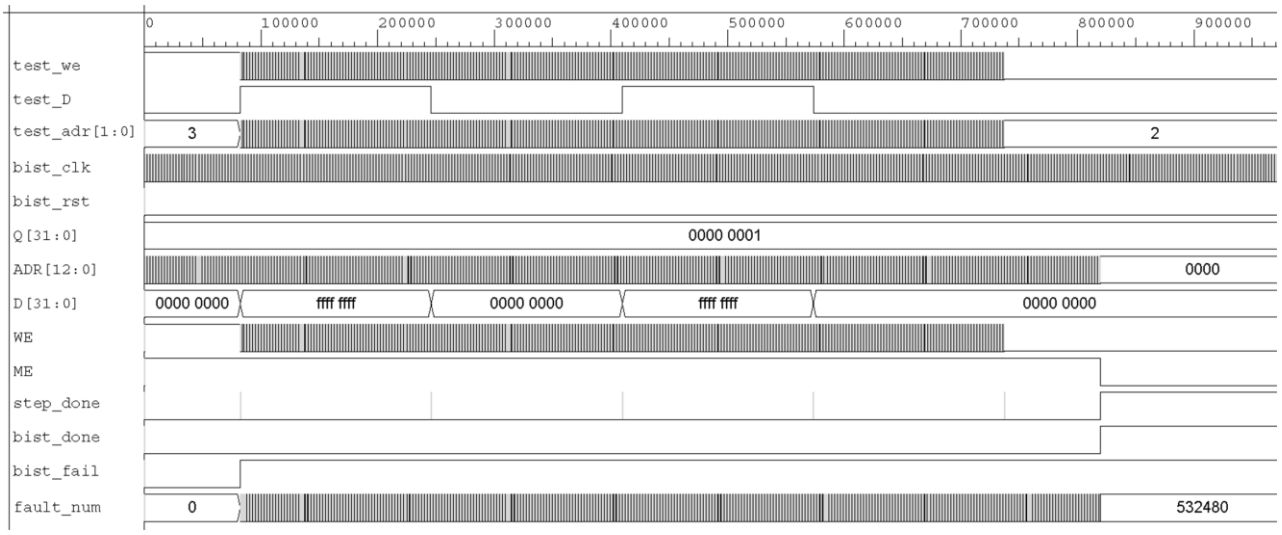


Fig. (8). The result of the controllable BIST with incorrect data of Q ports.

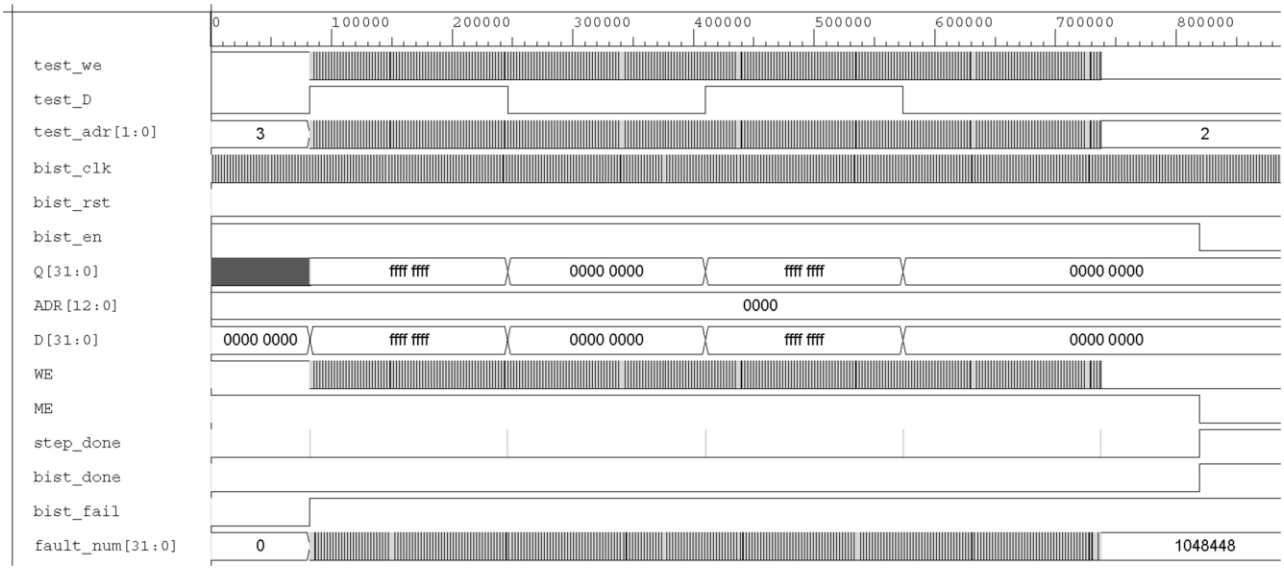


Fig. (9). The result of the controllable BIST with incorrect address.

4. DISCUSSION AND CONCLUSION

Most of the existing BIST circuits only test memory according to several algorithms. Sometimes they may not be able to meet the memory test requirements.We have developed a flexible and controllable BIST circuit in this paper. In the proposed controllable BIST circuit, we use three external inputs and five internal modules to implementthe process of test pattern generation and response analysis. The three external inputs, *i.e.*test we, test_D and test_adr controls the generation of WE, D and ADR signals respectively. The five internal modules, *i.e.*Control Generator, Data Generator, Address Generator, Data Comparator and Fail Accumulator perform the generation of WE, D and ADR, response analysis and faults statistics. The test pattern generation of the proposed controllable BIST circuit is controlled by external signal, not fixed by hardware, so the controllable BIST circuit can not only achieve correct test function, supports a variety of March algorithms, but also greatly simplifies the internal test pattern generation circuit, sharply reduces the hardware overhead of BIST circuit, and improves the timing characteristics and the test flexibility.Therefore, it is an easy-to-use and area saving BIST controller for testing embedded SRAMs.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests regarding the publication of this paper.

ACKNOWLEDGMENTS

This work is supported by the Natural Science Fund for Colleges and Universities in Anhui Province Grant No.KJ2013A006, the Anhui Provincial Natural Science Foundation Grant No.1308085QF112, the National Natural Science Foundation of China Grant No.61001081, and 2014 Talents Project for Promoting Higher Education inAnhui Provincial.

REFERENCES

[1] J.C. Yeh, S.F. Kuo, C.W. Wu, C.T. Huang, and C.H. Chen, "A Systematic approach to reducing semiconductor memory test time in mass production", *IEEE International Workshop on Memory Technology, Design, and Testing (MTDT),* 2005, pp. 97-102

[2] M. Zhang, D. Tao, and B. Wei, "A programmable BIST for embedded SDRAM", *International Symposium on VLSI Technology, Systems, and Applications,* 2001, pp. 244-248
[http://dx.doi.org/10.1109/VTSA.2001.934530]

[3] Y. Fukazawa, Y. Yoshikawa, H. Ichihara, and T. Inoue, "A design of response analyzers with self- distinguishability in built-in self-test", *International Symposium on Communications and Information Technologies (ISCIT),* 2010, pp. 732-737
[http://dx.doi.org/10.1109/ISCIT.2010.5665086]

[4] D.K. Pradhan, and L.C. Liu, "EBIST: a novel test generator with built-in fault detection capability", IEEE Trans. Comp.-", *Aided Design of Integrated Circuits and Systems,* vol. 24, no. 9, pp. 1457-1466, 2005.
[http://dx.doi.org/10.1109/TCAD.2005.850815]

[5] M.I. Masnita, W. Zuha, R.M. Sidek, and I.A. Halin, "The data and read/write controller for March-based SRAM diagnostic algorithm MBIST", *IEEE Student Conference on Research and Development (SCOReD),* 2009, pp. 96-299
[http://dx.doi.org/10.1109/SCORED.2009.5443018]

[6] W.L. Wang, and K.J. Lee, "A complete memory address generator for scan based March algorithms", *IEEE International Workshop on Memory Technology, Design, and Testing,* 2005, pp. 83-88

[7] M. Inoue, T. Yamada, and A. Fujiwara, "A new testing acceleration chip for low-cost memory tests", *IEEE Des. Test Comput.,* vol. 10, no. 1, pp. 15-19, 1993.
[http://dx.doi.org/10.1109/54.199800]

[8] M. Linder, A. Eder, U. Schlichtmann, and K. Oberlander, "An analysis of industrial SRAM test results-A comprehensive study on effectiveness and classification of March test algorithms", *Design & Test,* vol. 31, no. 3, pp. 42-53, 2014.
[http://dx.doi.org/10.1109/MDAT.2013.2279752]

[9] M. Saha, and B.K. Sikdar, "A self testable hardware for memory", *IEEE International Conference on Circuits and Systems (ICCAS),* 2013, pp. 45-50

[10] C.V. Gayathri, N. Kayalvizhi, and M. Mallikadevi, "Generation of New March Tests with Low Test Power and High Fault Coverage by Test Sequence Reordering Using Genetic Algorithm", *IEEE International Conference on Advances in Recent Technologies in Communication and Computing,* 2009, pp. 699-703

[11] C. Chen, H. Shih, C. Wu, C. Lin, P. Chiu, S. Sheu, and F. Chen, "RRAM defect modeling and failure analysis based on march test and a novel squeeze-search scheme", *IEEE Trans. Comput.,* vol. 64, no. 1, pp. 180-190, 2015.
[http://dx.doi.org/10.1109/TC.2014.12]

[12] G. Harutunyan, V.A. Vardanian, and Y. Zorian, "A New Method for March Test Algorithm Generation and Its Application for Fault Detection in RAMs", IEEE Trans. Comp.-", *Aided Design of Integrated Circuits and Systems,* vol. 31, no. 6, pp. 941-949, 2012.
[http://dx.doi.org/10.1109/TCAD.2012.2184107]

[13] G. Harutunyan, V.A. Vardanian, and Y. Zorian, "Minimal March tests for detection of dynamic faults in random access memories", *Eleventh IEEE European Test Symposium,* 2006, pp. 43-48
[http://dx.doi.org/10.1109/ETS.2006.32]

[14] L. Yu, J. Hung, B. Sheu, B. Huynh, L. Nguyen, S.L. Wu, L.T. Wang, and X.Q. Wen, "Hybrid Built-In Self-Test Architecture for Multi-port Static RAMs", *IEEE 25th International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT),* 2010pp. 331-339
[http://dx.doi.org/10.1109/DFT.2010.47]

[15] Y. Nakamura, J. Savir, and H. Fujiwara, "Defect level *vs*. yield and fault coverage in the presense of an unreliable BIST", *IEICE Transaction on INF. & SYST., E88-D(6),* pp. 1210-1216, 2005.

Received: March 28, 2015 Revised: June 17, 2015 Accepted: July 08, 2015