Interfacing BIST with NAND Flash Memory using NAND Controller

Varun Patel, Prof. Morris Jones

Department of Electrical Engineering, San Jose State university, San Jose, California 95192.

Introduction

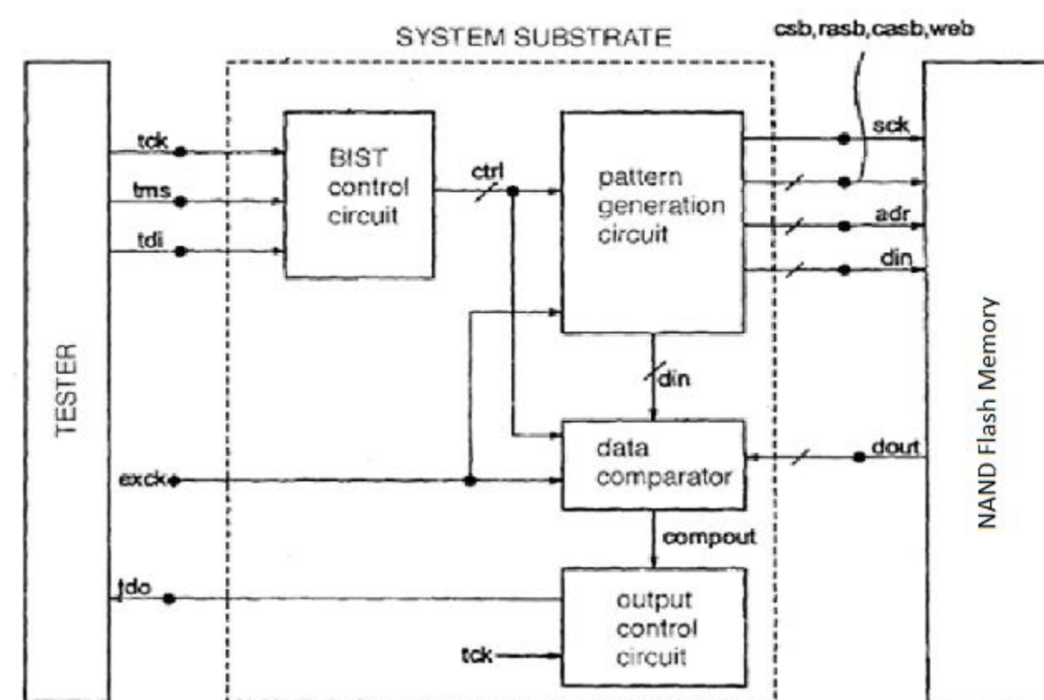
Sub-micron technology, complex array structures and more complex features calls for exhaustive self-testing for NAND memory.

BIST provides benefits of self-testing on-chip mechanism, hence reduces test time. Challenges includes test time, reliability and area overhead[1]. Knowing the limitations for NAND many faults occur in it. Many faults like disturbance, circuit level, conventional were researched and categorized along with test algorithms, however Diagonal or March algorithms provides less fault coverage[2]. Furthermore, March-like tests distinguishes just disturb faults .

Because March-FT algorithm aims to get full coverage for transient and conventional faults with least number of program/erase cycles[3].
Three unique alternatives are performed for testing which includes quick test, full test and random test.
In addition marking of defective addresses are performed

Methodology

BIST Architecture



- BIST Controller- Utilized to control and operate general mechanism for BIST
- Test Pattern Generator- Produce address sequence for CUT(memory)
- Test Collar- Utilized for separating BIST hardware and Flash controller
- Data Analyzer- Analyze the test as pass or fail.

Pin Sequencer block was designed as NAND memory requires user command for operation. This block communicates with NAND memory.

It decodes the user operation command, converts the address specified into 5 address cycle which is required by the memory model.
Verilog HDL with ModelSim simulator was used to achieve the functionality of the above blocks.

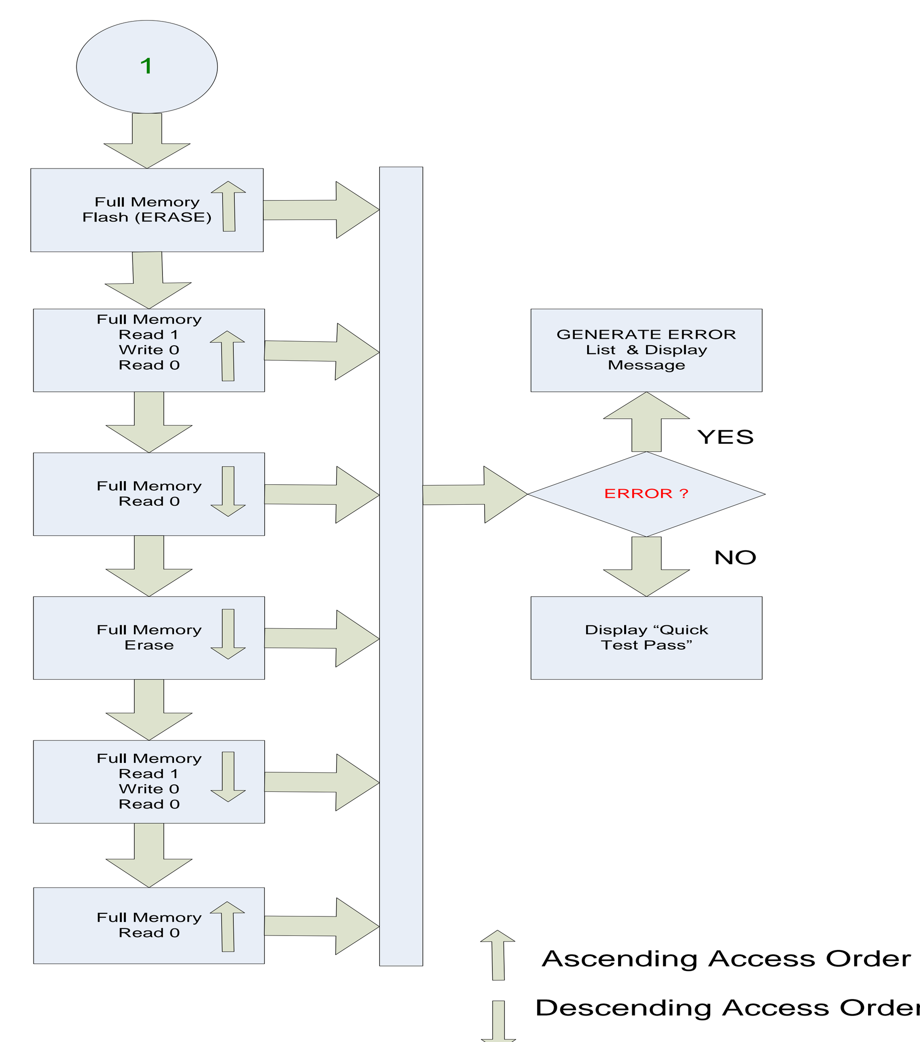
Methodology

March-FT Algorithm

March-FT Algorithm: {F↑; ↑ (R1, W0, R0); ↓R0; F↓; ↓ (R1, W0, R0); ↑R0}[4]
Test Length = 2F + 2NP +2 NR where, N is number of addresses
The main difference is, in conventional algorithms we use write-1 operation as reset condition, but in March-FT this operation is replaced by Flash Erase(F) operation.

- Verilog HDL with ModelSim simulator was used.
- NAND Flash memory obtained from Micron Technologies.
- LFSR was designed to generate random data and random addresses for testing according to algorithm specified.
- Testbench was designed to give user test command as well as case to perform quick test, full test or random test.
- As the various faults occur at circuit level, these faults were injected from external system to NAND Flash.
- Ultimate result is 'Test Pass' or 'Test Fail', and if required detected the bad block.

Below is described the 'Full Memory Test' case.



Results

Coverage for conventional and transient faults have almost full fault coverage.
March-FT algorithm is compared with the Flash March and Diagonal results for fault coverage.

Parameters	March-FT	Flash March [13]	Diagonal –FT [10]
WPD	**100%**	100%	100%
BPD	**100%**	100%	100%
WED	**100%**	100%	100%
BED	**100%**	100%	100%
OE	**100%**	100%	100%
RD	**100%**	100%	100%
SAF	**100%**	100%	100%
TF	**100%**	100%	100%
SOF	**100%**	50%	100%
AF	**100%**	100%	81.6%
CFst	**100%**	75%	89.15%
Fault coverage	**100%**	93.18%	97.34%
Test length	**2F + 2NP + 6NR**	2F + 2NP + 4NR	$2F + (1N+2\sqrt{N})P + (2N+\sqrt{N})R$

In spite of the fact that March FT calculation requires additional testing time, the upside of it incorporates, many more for fault scope, easy in era of examples, are general also symmetrical during testing.
To verify the test time of a block for different algorithms a random block was generated from random generator in case to test random case.
Looking at the severity between fault coverage and test time, fault coverage proves to be critical due to vendor's creditability.

Parameters | Test Time	March-FT	Flash March	Diagonal-FT
Block Size	512KB	512KB	512KB
Test Time	309.28ms	252ms	276ms

To detect a defective block, a fault was generated by during the execution of a algorithm in order to read incorrect value, and thus this detected the wrong page value.
The result was 'Test Fail' and detected the faulty block.
Also as an addition, area overhead of the BIST circuitry should ideally be 1.20%

Conclusions

Two algorithms presented in report i.e. Diagonal and March-FT are effective, but March-FT have the lead over diagonal in terms of coverage, though time required is little more as compared to test time required for diagonal algorithm. Thus, the objective to obtain 100% fault coverage for conventional faults and disturb faults is achieved using the March-FT algorithm.
Thus, the feasibility to use March-FT based BIST for NAND Flash memory is possible and facilitates regularity for getting good test generation mechanism. The future possibility can be to make it versatile so that more applications can be tested. Also diagnostic and repairing mechanism can be applied

References

[1] Pod slurping - an easy technique for stealing data GFI white paper.
[2] Hamdioui, S., Krishnaswami, V., Irobi, I.S., Al-Ars, Z.A, "New Test Paradigm for Semiconductor Memories in the Nano-Era", in Test Symposium (ATS), 2011 20th Asian. Pages - 347 – 352.
[3] Xiaogang Du, Mukherjee N, Wu-Tung Cheng, Reddy S. M - Full-Speed Field-Programmable Memory BIST Architecture.. Test Conference, 2005. IEEE International. Austin, Texas. Pages 9 pp. – 1173.
[4] Adaptive Endurance Coding for NAND-Flash. Ashish Jagmohan, Michele Franceschini, Luis A. Lastras-Montaño, John Karidis. GLOBECOM Workshops (GC Wkshps), 2010 IEEE. Pages: 1841-1845.
[5] Jen-Chien Yeh, Yan-Ting Lai, Yuan-Yuan Shih and Cheng-Wen Wu. Dept. of Electrical Engineering, National Tsing Hua University - Flash Memory Built-In Self-Diagnosis with Test Mode Control. IEEE 2005.

Acknowledgements

I thank Prof. Morris Jones for providing invaluable expertise, required guidance from time to time.

www.PosterPresentations.com