High secured and area optimized Online Memory Testing for efficient Fault Diagnostic Systems

Takkellapati Venu Gopi[1], Nelarapu Yelli Mahesh[2], Ippala Yasodhara Reddy[3], Mallireddy Leela Krishna Sankara Renuka Reddy[4], ***N.veeraiah*** [5]

[1,2,3,4]***B.TechStudent,***

[5]*Assistant professor in E.C.E,*

V.K.R,V.N.B & A.G.K. ENGINEERING COLLEGE, Gudivada ,A P

ABSTRACT:

The main intention of this project is to recommend a fault diagnoses structure for revealing of any software or hardware or permanent failures in the embedded read only memories. BIST controller, along with row selector and column selector is designed to meet necessities of at speed test thus enabling detection of timing defects. The projected approach offers a simple test flow and does not require intensive communications between a BIST controller and a tester. The system rests on partitioning of rows and columns of the memory array by employing low cost test logic. It is intended to meet requirements of at-speed test thus enabling detection of timing defects.

INTRODUCTION:

Non-volatile memories are among the oldest programmable devices, but continue to have many critical uses. ROM, PROM, EPROM, EEPROM, and flash memories have proved to be very useful in a variety of applications. Traditionally, they were primarily used for long-term data storage, such as look-up tables in multimedia processors or permanent code storage in microprocessors. Due to the high area density and new sub micrometer technologies involving multiple metal layers, ROMs have also gained popularity as a storage solution for low voltage/ low-power designs. Moreover, different methods such as selective pre-charging, minimization of non-zero items, row(s) inversion, sign magnitude encoding, and difference encoding are being employed to reduce the capacitance and/or the switching activity of bit and word lines.

Most large application-specific integrated circuits (ASICs) use scan as a fundamental design for test (DFT) methodology. It has been observed that the amount of test data required to test one gate in a large design can exceed 1 Kb. This depends on several factors, such as the design style, fault models used, and capabilities of the automatic test pattern generation (ATPG) tool used. However, even using state-of-the-art ATPG tools, several gigabits of test data may be required for a multi-million gate design.

Testers have a limited number of channels designed to drive scan chains, typically around 8. The speed of loading is also limited by the maximum scan frequency, usually around 10 to 50 MHz. The large volume of test data creates two problems for testers: capacity and test application time. Very often, testers do not have enough memory to store the entire test set to cover stuck-at, transition, and path delay faults. In some cases, the available memory is not even large enough to store a complete test set for stuck-at faults. In this case, either very time consuming reloads are required, or only a subset of the test vectors is applied with the corresponding reduction of fault coverage.

In BIST, pseudorandom patterns are generated on chip, the responses are compacted on chip, and the control signals are driven by an on-chip controller. The amount of test data exchanged with the tester is therefore drastically reduced. In addition, the scan cells are configured into a large number of relatively short scan chains, thus reducing the time required to apply a single test pattern. The low memory and performance requirements on the tester allow the usage of very low cost testers for manufacturing test of designs with logic BIST. Logic BIST is based on pseudorandom patterns and involves compaction of test responses. Those two characteristics impose more stringent design rules on the BIST logic than scan with stored patterns. Logic BIST requires that bus conflicts are eliminated, sources of X states are properly bounded to prevent corruption of the signatures, the circuit is random-pattern testable, etc. In many cases, the original design does not satisfy many of these requirements, thus posing barriers to BIST. In those cases, and in general, the only practical way to implement BIST is through automation of the design tasks and their integration in the overall methodology and design flow. The introduction of logic BIST at the Texas Instruments MOS design center is driven by limitations of the currently- used test equipment and a number of specific goals. In particular, the testers currently used already limit the ability to run available tests in the following ways:

1. Scan operates at a maximum frequency of 50 MHz.

2. Tester scan memory is usually filled.

3. Tester has a maximum of 8 scan chains, resulting in a long test application time for large designs.

4. Tester functional test memory is also filled, leading to utilization of as little as 10% of the available functional tests.

5. Transition fault or path delay scan ATPG patterns are not used due to lack of tester memory.

FAULT DIAGNOSIS

So far the outputs from the LFSR are directly given to the circuit under test (CUT) through scan chain directly. LFSR is mainly used to generate patterns. Scan chain is a group of registers. Here the scan chain is mainly used to store the patterns generated
fromLFSR.

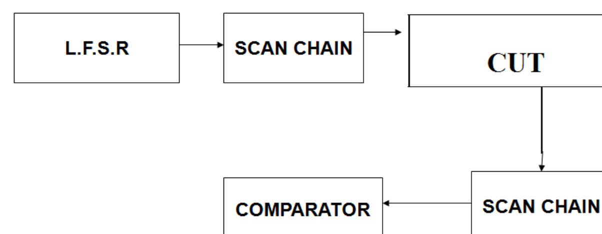


Fig 2.1 Existing method for fault diagnosis

These patterns are applied to CUT and the outputs generated from CUT are again stored in scan chain. For every input response stored in scan chain the corresponding output patterns are stored in the scan chain i.e., located at the output of the CUT. Here in comparator, it compares the input to CUT and corresponding output of CUT.

Errors

There are three types of errors that often occur:

- Soft errors
- Hard errors
- Unhandled errors

Soft errors

Soft errors usually happen due to handling mistakes. Coppermine will display an error message, but we can continue to browse the gallery. The soft errors are fully localized so the programmers of Coppermine have (more or less) anticipated that such an error may happen under certain circumstances.

Hard errors

Hard errors are messages that won't go away, usually because something is broken and needs fixing. This is usually the case if something is wrong with the database. As a result, you will see the reared "Fatal error" message that usually doesn't mean much too inexperienced users. Hard errors usually indicate that something is wrong which the programmers of Copper mine could not foresee, that's why only a generic error message is being displayed.

Unhandled Errors

Unhandled errors are the ones that usually are most tricky to solve, as they make the application Coppermine crash in mid-air, without a meaning full error message that could tell users what is wrong. This is usually the case if you get a blank page or just a template error.

Fatal error:

The message "Fatal Error" is a generic error message that just says that there is something wrong. It usually is a hard error we won't be able to continue using Copper mine unless we fix the reason for the error message. The reasons for such a generic error message are manifold. For security reasons the "real" error message is not being displayed by default, but only the generic "Fatal Error" message.

Stuck-at faults

Types of stuck-at-faults are

- Single stuck-at-faults
- Multiple stuck-at-fault

Single stuck-at faults

Three properties define a single stuck-at fault

- Only one line is faulty
- The faulty line is permanently set to 0 or 1
- The fault can be at an input or output of a gate

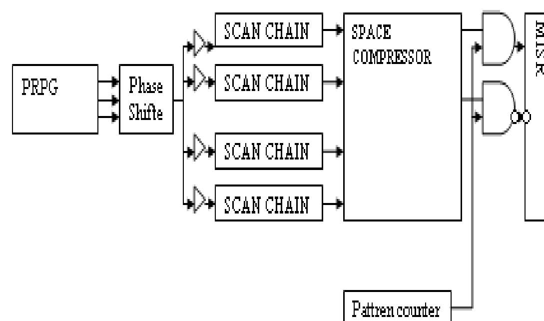
Following classes of single stuck-at faults are identified by fault simulators:

- *Potentially-detectable fault* -- Test produces an unknown (X) state at *primary output* (PO), detection is probabilistic, usually with 50% probability.
- *Initialization fault* -- Fault prevents initialization of the faulty circuit; can be detected as a potentially-detectable fault.
- *Hyperactive fault* -- Fault induces much internal signal activity without reaching PO.
- *Redundant fault* -- No test exists for the fault.
- *UN testable fault* -- Test generator is unable to find a test.

LOGIC BIST ARCHITECTURE

A Generic scan based logic BIST architecture A generic single clock logic BIST architecture based on the well-known STUMPS technique is illustrated in Figure 1. The figure depicts the circuit-under-test or core, and the logic BIST controller in the highlighted area. The circuit is composed of combinational logic, and possibly embedded memories, separated by multiple scan chains. Various components of the logic BIST controller are shown in the highlighted area. These components include test pattern generation block - composed of the pseudo-random pattern generator (PRPG) and phase shifter circuit, the output response analysis block - composed of multiple input signature register (MISR), space compactor, and optional AND gates. In addition, there are two counters: the pattern counter and the shift counter which for each pattern keeps track of the number of cycles required to fill the scan chains. The decoder block shown in the figure drives the test points. Finally, the multiplexers between the phase shifter and scan inputs are used to concatenate several shallow BIST-mode scan chains into a few deep ATPG-mode scan chains accessed directly from the chip pins in case top-up ATPG is use to improve the fault coverage obtained by BIST.

The BIST can be initiated either through a boundary scan TAP controller or by appropriately asserting a set of new primary inputs in case a stand-alone mode logic BIST controller is implemented. Prior to running the actual test, the controller components such as PRPG, MISR and the pattern counter need to be initialized. In addition, the internal scan chains can also be optionally initialized.

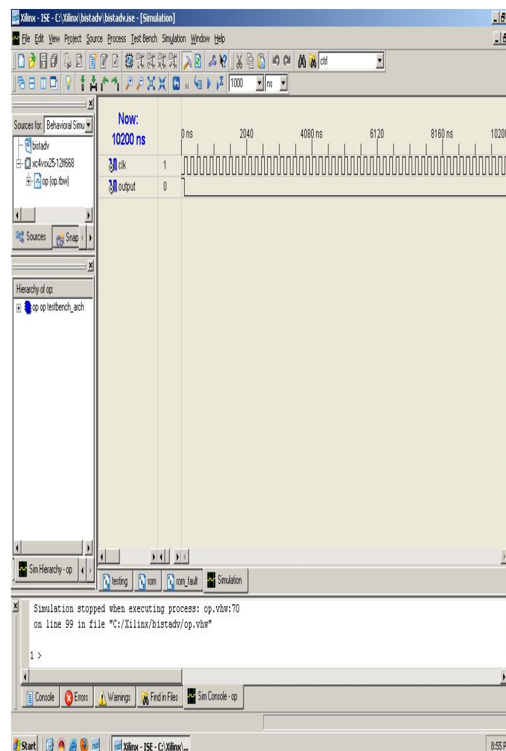


The actual test of the circuit consisting of several patterns then begins. For each pattern the shift counter counts cycles where, the number of cycles in the shift window is equal to the length of the longest scan chain and , the number of cycles in capture window is typically equal to one for a simple capture window. Hence in order to reduce the test application time it is necessary to configure the scan cells into a large number of shallow scan chains. A systematically designed phase shifter circuit is placed between the LFSR and the scan chain inputs to

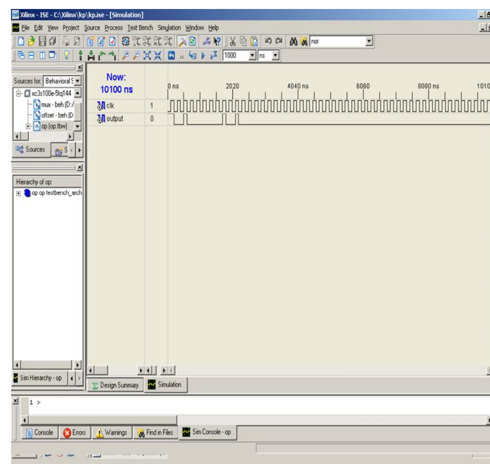
eliminate structural dependencies and allow a large number of scan chains to be driven by a relatively short LFSR.

Similarly an XOR structure called space compactor is required to compact the large number of scan outputs before feeding them to a small MISR. During the shift window of a pattern, new pseudo random values from the PRPG are loaded into the scan chains while simultaneously unloading and compacting the circuit's response for the previous pattern into the MISR. In case the internal scan chains are not initialized, for the first pattern, their unknown contents can be blocked as shown in Figure 1 by means of AND gates in front of the MISR. After the scan chains are completely loaded, the multiplexers in the scan cells are placed in system mode for one cycle to capture the circuit's response. This sequence of events continues for each pattern. In addition, if multi-phase test scheme is used, at the beginning of each new test phase, the test point's decoder establishes a pre-determined set of values at the control points that remain fixed for that phase. Once all the test phases are applied, the contents of the MISR, i.e. the signature, can be either scanned out and compared externally or compared with an on-chip reference signature to determine the status of the circuit.

SIMULATION REPORT FOR A FAULT- FREE MEMORY:



SIMULATION REPORT FOR A MEMORY WHICH CONSISTS OF FAULTS:



SYNTHESIS REPORTS:

DEVICE UTILIZATION SUMMARY

Selected Device: 3s100etq144-4

Number of Slices: 180 out of 960 18%

Number of Slice Flip Flops: 222 out of 1920 11%

Number of 4 input LUTs: 266 out of 1920 13%

Number of IOs: 63

Number of bonded IOBs: 63 out of 108 58%

Number of GCLKs: 2 out of 24 8%

TIMING SUMMARY

Speed Grade: -4

Minimum period: 8.741ns (Maximum Frequency : 114.410MHz)

Minimum input arrival time before clock : 4.845ns

Maximum output required time after clock : 4.450ns

CONCLUSION:

In this paper, a new fault diagnosis scheme for embedded read-only memories is proposed. It reduces the diagnostic data that needs to be scanned out during ROM test such that the minimum information to recover the failure data is preserved, and the time to unload the data is minimized. The presented approach allows an uninterrupted collection and processing of test responses at the system speed. This has been achieved by using low-cost on-chip selection mechanisms, which are instrumental in very accurate and time efficient identification of failing rows, columns, and single memory cells. In particular, the scheme employs the original designs of row and column selectors with phase shifters controlling the way the address space is traversed.

Furthermore, the new combined selection logic allows the scheme to collect test results in parallel (leading to shorter test time) without compromising quality of diagnosis. Results of experiments performed on several memory arrays for randomly generated failures clearly confirm high accuracy of diagnosis of the scheme provided the signature registers and the proposed selection logic are properly tuned to guarantee a desired diagnostic resolution.

REFERENCES:

[1] B. T. Murray and J. P. Hayes, "Testing ICs: Getting to the core of the problem", *Computer*, vol 29, pp. 32-38, November 1996.

[2] M. Abramovici, M. A. Breuer and A. D. Friedman, "*Digital System Testing and Testable Design*", IEEE Press, New York, 1990.

[3] J. Rajski and J. Tyszer, "Design of phase shifters for BIST applications", *Proc. IEEE VTS*, pp. 218-224, 1998.

[4] H. Al-Asaad, B. T. Murray and J. P. Hayes, "On-line BIST for embedded systems", *IEEE Design and Test of Computers*, vol 15, pp. 17-24, October-December 1998. [5] P. Bardell, W. McAnney and J. Savir, "*Builtin Self-test for VLSI: Pseudorandom Techniques*", John Wiley and Sons, New York, NY, 1987.

[5] J. Rajski and J. Tyszer, “Diagnosis of scan cells in BIST en-vironment,” IEEE Trans. Comput., vol. 48, no. 7, pp. 724–731, Jul. 1999.

[6] J. Rajski, N. Tamarapalli, and J. Tyszer, “Automated synthesis of phase shifters for built-in self-test applications,” IEEE Trans. Comput.-Aided Design, vol. 19, no. 10, pp. 1175–1188, Oct. 2000.

[7] J. Rajski, N. Mukherjee, J. Tyszer, and A. Pogiel, “Fault diagnosis in memory BIST environment,” U.S. Patent Applicat. 20110055646, Sep. 18, 2008.

[8] C. Selva, R. Zappa, D. Rimondi, C. Torelli, and G. Mastrodomenico, “Built-in self diagnosis device for a random access memory and method of diagnosing