Memory BIST

Fulvio Corno
Maurizio Rebaudengo
Matteo Sonza Reorda

Politecnico di Torino
Dipartimento di Automatica e Informatica

Introduction

- Embedded RAMs have limited accessibility
- For this reason, BIST a popular test solution for them
- Memory BIST is sometimes used even for stand-alone memory devices.
Pseudo-random BIST

- If BIST is adopted, using a pseudo-random generator is hardly a solution:
  - Because it requires long test times to excite traditional fault models: for example, to test all the single stuck-at faults in the memory cells with a 99.9% confidence requires performing 48,5 accesses per cell
  - Because it does not addresses the fault models which are proper of memories.

Goals

The choice of the algorithm and that of the BIST architecture implementing it should take into account the following goals:
- Minimizing the area overhead
- Minimizing the performance overhead
- Maximizing the achieved fault coverage wrt the set of faults modeling the real defects.
Generating stimuli

Different solutions are possible to stimulate the memory during test:

- Resorting to an embedded microprocessor
- Through an external ATE, via DMA
- Through an external ATE, via scan chains
- Memory BIST.
Resorting to an embedded µP

- Vectors to be applied to the RAM are generated by the processor
- Someone must write the corresponding code (typically in Assembly language)
- Only functional patterns can be applied
- The test can not necessarily be performed after product delivery.

Through an external ATE, via DMA

- This solution
  - requires a DMA controller to be available and accessible from the outside
  - May involve some routing problem if several memory exist in the same circuit
- Some ATEs already support a “Memory Test Function (MTF)”. 
Through an external ATE, via scan chains

- RAMs are often surrounded by flip flops (put in scan for test purposes)
- Using the scan chain(s) it is possible to perform read and write operations on the memories
- Performing the test using the scan chains
  - Requires very long test times
  - Does not address timing faults.

Memory BIST

- It requires designing a BIST circuitry, whose main I/O signals are
  - BIST-INVOKE, BIST-RESET, BIST-DONE, BIST-FAIL
- the BIST circuitry can be shared by several memory modules, or distributed among the modules
- The BIST circuitry can follow two solutions:
  - Hardwired BIST
  - Programmable BIST.
**Hardwired BIST**

- The test algorithm is hardwired.
- **Advantages**
  - High speed
  - Minimum area overhead
- **Disadvantages**
  - The algorithm can not be modified (e.g., for diagnosis purposes, or to support a new technology).

**Programmable BIST**

- The BIST circuitry
  - includes some memory, where the information about the selected algorithm can be uploaded from the outside
  - Executes the selected algorithm.
- **Advantage**
  - Flexibility
- **Disadvantage**
  - Higher area overhead.
Memory BIST: advantages

- The test can be performed at-speed even with low-frequency ATEs
- Managing a memory BIST session does not require long test vectors to be stored in the ATE memory.

On-line test

For safety- or mission-critical applications, it may be required to test devices during their life.

Some of the DfT structures introduced for manufacturing test can be re-used for on-line test.
On-line test types

- **Concurrent BIST**: BIST is performed during the normal activity of the RAM
- **Non-Concurrent BIST**: test is performed during idle times. The original content of the memory is lost.
- **Transparent BIST**: test is performed during idle times. The original content of the memory is restored at the end of the test.

The test algorithm

- Very often the selected algorithm is a March one
- The BIST controller generates the sequence of read/write operations composing the algorithm
- The addresses are generated by proper up and down counters
- Comparisons are done with the expected values, or comparing the outputs of modules tested in parallel.
The BIST controller

- During the test, the BIST controller drives the following signals:
  - DATA IN and DATA OUT, or DATA BUS
  - ADDRESS
  - Control signals (RD, WR, etc.)

Requirements

- The BIST controller
  - is implemented using the same technology used for the rest of the chip
  - It must be able to write different values to the memory
  - It communicates with the chip interface through suitable protocols
- Standards can significantly simplify the design of BIST solutions.
BIST circuitry

Interface (I)

- the interface between the BIST controller and the outside world can be based on
  - Dedicated pins
  - Multiplexed pins
  - A IEEE 1149.1 TAP
  - An upper level test controller.
Interface (II)

- Input signals include
  - Invoke
  - Retention
  - Debug
- Output signals include
  - Done
  - Error
  - Debug data.

Example

```
Algorithm controller  Comparator
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Address generator</td>
<td>Data generator</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>D_{in}</td>
<td>D_{out}</td>
</tr>
<tr>
<td>A_{in}</td>
<td>A</td>
</tr>
<tr>
<td>WR_{EN}</td>
<td>WR</td>
</tr>
<tr>
<td>RD_{EN}</td>
<td>CE</td>
</tr>
</tbody>
</table>
```

Signals: Invoke, Retention, Release, Bitmap, Done, Fail, Hold_out, Bitmap_out.
Design issues

- Trade-off between distributed controller and routing costs
- Trade-off between serial and parallel test, taking into account
  - Test time
  - Power consumption
  - Complexity of the interface with the ATE
- Trade-off between costs and diagnostic properties.

Commercial tools

- Several commercial tools exist for automating the insertion of proper MBIST circuitry:
  - Mentor Graphics: Design-For-Test MBISTArchitect
  - LogicVision: IC Memory BIST
  - Credence HPL: MemBIST
  - Synopsys: DesignWare.