Self-checking and Fault Tolerant approaches can help BIST fault coverage: a case study

Fulvio CORNO, Paolo PRINETTO, Matteo SONZA REORDA
Politecnico di Torino — Torino, Italy[1]

This abstract describes the design of a FIFO component with BIST capabilities. The component is now being used in the Italtel standard library and is exploited in several industrial designs. Our main contribution is to show how the effectiveness of complex BIST design can be improved, and brought to acceptable fault coverage levels, through the coupling with more advanced test architectures developed for on-line testing schemes. We adopted fault tolerant schemes and self-checking components to ensure that critical circuitry is correctly working.

1. Design goals

The constraints imposed by the targeted industrial environment strongly limited design freedom and influenced final choices. In particular, the most compelling constraints were compliance to a company-wide BIST protocol through the Boundary-Scan Test Access Port, the independence of the design from the size of the RAM, and no knowledge about the internal structure of the RAM.

The following testability goals were then settled:

- the embedded RAM should be tested for at least stuck-at, transition and coupling faults
- RAM test must be implemented using the same addressing logic of the non-BIST component
- the data path (counters, comparators, multiplexers, etc.) should be tested for stuck-at faults
- the control unit responsible for the "normal mode" behavior is a critical component, therefore 100% fault coverage is required
- in the BIST controller faults that shorten the test sequence have to be covered.

2. Test strategy

Given the above constraints, different design choices concerning the BIST architecture were adopted:

- the dual-port *memory*, inclusive of all decoding logic: the March B– test for SOA memories is selected [VdGZ93], that meets fault coverage goals
- the *data path*: a functional test is applied by the BIST controller while the March test is in progress
- the *normal-mode control unit*, controlling the FIFO behavior of the component while not under test. Several alternatives were examined. The implemented alternative is a fault-tolerant control unit, using triple redundancy. Since the control unit is small (one flip-flop and some tens of gates), the area overhead for triplication is comparable with other solutions.
- the *BIST controller*, i.e., the FSM responsible for generating the correct sequence of operations composing the test algorithm. A self-checking architecture must be implemented to avoid that a fault in the BIST controller could flag a circuit as good without actually performing the test

3. Conclusions

The BIST component has been designed in the Synopsys environment with the SGS-Thomson ISB24000 technology. It is now being used in the Italtel standard library and exploited in industrial designs. The attained fault coverage was quite satisfying.

Part	TFC%	fault model	test strategy
memory	100	[CPSB95]	BIST
data path	83	stuck-at	functional testing
control unit	N/A		fault tolerance by triplication
voters	100	stuck-at	exhaustive BIST
BIST controller	74	stuck-at	self-checking via signature analysis

4. References

[CPSB95] P. Camurati, P. Prinetto, M. Sonza Reorda, S. Barbagallo, A. Burri, D. Medina: "Industrial BIST of Embedded RAMs", IEEE Design and Test of Computers, Fall 1995, pp. 86-95

[VGSZ94] J. van de Goor, I. Schanstra, Y. Zorian: "Fault Models and Test for Ring Address Type FIFOs", VTS'94: IEEE VLSI Test Symposium, 1994

[VdGZ93] J. van de Goor, Y. Zorian: "Effective March Algorithms for Testing Single-Order Addressed Memories", EDAC'93: IEEE European Design Automation Conference, 1993, pp.499-505

[1] Contact address: Paolo Prinetto, Politecnico di Torino, Dip. Automatica e Informatica, c.so Duca degli Abruzzi, 24, I-10129 Torino, e-mail **Paolo.Prinetto@polito.it**

ED&TC '96
0-89791-821/96 $5.00