Optimized Built-In Self-Repair for Multiple Memories

Abstract:
A new built-in self-repair (BISR) scheme is proposed for multiple embedded memories to find optimum point of the performance of BISR for multiple embedded memories. All memories are concurrently tested by the small dedicated built-in self-test to figure out the faulty memories, the number of faults, and irreparability. After all memories are tested, only faulty memories are serially tested and repaired by the shared built-in redundancy analysis according to the sizes of memories in descending order. Thus, the fast test and repair are performed with low area overhead. To accomplish an optimal repair rate and a fast analysis speed, an exhaustive search for all combinations of spare rows and columns is proposed based on the optimized fault collection. Experimental results show that the proposed BISR has the optimal repair rate because of the exhaustive search. The performance of the proposed BISR is located in the optimum point between the test and repair time, and the area overhead. For example, the proposed BISR requires 49.6% of the area and 1.3 times of the test and repair time in comparison with parallel BISR scheme for four memories (one 128 K, two 256 K, and one 512 K memories). Furthermore, the more there are memories, the more superior performance in terms of the test and repair time, and the area overhead is shown. The proposed architecture of this paper analysis the logic size, area and power consumption using Xilinx 14.2.

Enhancement of the project:

Existing System:
There are various BISR schemes for multiple memories. Because routing, timing, and power consumption are realistically considered, the memories are partitioned into memory groups. Most BISR schemes perform test and repair for the memory groups. Processor-based BISR schemes using infrastructure intelligent property have been proposed. A BISR scheme based on a self-test and repair processor can perform multiple-time memory repair. To perform the RA algorithm, the BISR scheme uses specific instructions. However, because the processor-based BISR schemes use specific instructions to construct the RA algorithm and the RA algorithm may need multiple instructions to be implemented, the test and repair time can be long.
Reconfiguration BISR schemes based on a serial test and repair are proposed to reduce the area overhead. Although they can reduce area overhead through hardware sharing, the test and repair time can be long because of the memory repairs are completed serially. To reduce the test and repair time, a BISR scheme based on a parallel procedure is proposed. However, the area overhead required to implement these BISR schemes is very high, because they use many BIST and BIRA modules. Parallel test and serial repair BISR schemes are proposed. These BISR repair memories in serial after all memories have simultaneously been tested. The area overhead, and test and repair time are placed in the middle of the parallel test and repair and the serial test and repair.

Disadvantages:

- Repair rate is less

Proposed System:

Because there are many memories in the SoC, the memories are partitioned into groups considering the routing, timing, and power consumption. Each group has the proposed BISR circuit. In other words, the number of required BISR circuits is equal to the number of memory groups in the SoC. Then, the proposed BISR targets multiple memories in each group. All the memories are tested and classified as faulty memory or fault-free memory by parallel test procedure. The fault free memories are excluded from the test procedure, because it is not necessary to repair the fault-free memories. Only faulty memories are selected and serially tested and repaired according to the size of memories in descending order.

Fig. 1 shows a block diagram of the proposed BISR architecture. It mainly consists of BIST and BIRA modules. The main purpose is to classify the memories as faulty or not, and the number of faults in each memory is stored in the dedicated wrapper.
Fig. 1. Block diagram of the proposed BISR architecture.

Test and Repair Method of the Proposed BISR:

Fig. 2 shows a conceptual block diagram of the proposed BIST and wrapper modules. Assume that the number of memories is n. The BIST module consists of a test pattern generator (TPG), a test address generator (TAG), and a controller (CTR). The wrapper, which is dedicated to the memory, consists of a comparator (CMP) and a fault number register (FNR).
Fig. 2. Block diagram of the proposed BIST and wrapper modules.

PROPOSED BIRA APPROACH:

Fig. 3 shows a conceptual block diagram of the proposed BIRA architecture. It consists of a multifault detector, a counter, fault storing CAMs, repair registers, and a CTR. If a fault is detected, it is sent to the multifault detector through the port Fault_info. There can be multiple faults in a word within a word-oriented memory. If there are multiple faults in a word, the faults are stored serially in the fault storing CAMs.

Fault Collection:

The notations are defined as follows. M is the number of rows of memory, and N is the number of columns of memory. The fault storing CAMs are classified into two CAMs. The enable flag is used to recognize the valid information. The row and column addresses store the row and column address fields.

Maximizing the Parent Faults:
For the same faulty memory, there can be different fault collection results depending on the fault detection order.

Redundancy Analysis:

The RA algorithms are based on a binary search tree, such as IS, ISF, or SFCC. Because each branch of the binary search tree can be a repair solution, and these RA algorithms compare a faulty address with every node in the branch, a considerable amount of time is required to find a repair solution.

Advantages:

- Optimal repair rate

Software implementation:

- Modelsim
- Xilinx ISE