International Journal of Engineering Inventions
ISSN: 2278-7461, www.ijeijournal.com
Volume 1, Issue 8 (October2012) PP: 76-80

AN OPTIMAL APPROACH FOR TESTING EMBEDDED MEMORIES IN SOCS

B.Prathap Reddy[1]   P.RameshYadav[2]
[1]M.Tech (pursuing), VITS, Proddatur
[2]M.Tech, (Ph D), Asst. Professor, Department of ECE, VITS, Proddatur

Abstract:—In general, RAM's are playing very important roles in design of SOC's, by improving the yield of RAM's we can improves the yield of SOC. yield of the chip is improves by repairing the RAM's in SOC. so many techniques are came to improve the yield of the SOC. the current paper presents the efficient technique for improves the yield of the SOC, which is called Reconfigurable Built-in self Repair scheme for RAM's in SOC(Re-BISR).in the technique we are using the spare cells (Spare rows/Spare columns) for eliminating the faults in the RAM's. The proposed repair scheme is increasing the repair rate and reconfigurable for less area, used to repair multiple RAM's with different sizes and redundancy. The experimental results show that proposed Re-BISR technique reduces the area and increases the yield of the RAM's.

Keywords:—Built-in self-test, Built-in self-repair, Built-in redundancy-analysis, memory testing, semiconductor memory.

I. INTRODUCTION

The VLSI manufacturing technology advances has made possible to put millions of transistors on a single die. A complex IC that integrates the major functional elements of a complete end product into a single chip is known as System on Chip (SOC). It enables the designers to move everything from board to chip. Reduction in size, lower power consumption, higher performance, higher reliability, reuse capability and lower cost are the benefits of using SOC. Redundancy increases the silicon area and thus has a negative impact on yield. To maximize the yield, redundancy analysis is necessary. embedded memories are harder to deal with Automatic Test Equipment (ATE). The BISR (Built in Self Repair) technique is a promising and popular solution for enhancing the yield of memories with the redundancy logic.

RAMs in an SOC usually have various sizes, different number of redundancies, and even different types of redundancy organizations. If each repairable RAM uses one self contained BISR circuit, then the area cost of BISR circuits in an SOC becomes high. To reduce the area cost, several processor based BISR schemes are also proposed. Therefore, a time efficient and area efficient BISR scheme is needed to improve the yield of RAMs in SOCs economically. The solution to the above problem is reconfigurable BISR (Re BISR) scheme, which is implemented in this thesis. The Built-in self diagnosis method presented for repairable SRAMs uses a reduced-instruction-set processor to determine a repair solution. The Re-BISR can be shared by multiple RAMs with different sizes and redundancy organizations. This can reduce the area cost of the BISR circuits in an SOC. Also, an efficient reconfigurable BIRA (Re BIRA) scheme is used to allocate 2D redundancies of multiple RAMs.

II. OVERVIEW OF BIST SCHEME

The various components of BIST hardware shown in fig 2.1.They are test pattern generator (TPG), test controller; circuit under test (CUT), input isolation circuitry and the output response analyzer (ORA).The memory BISR (MBISR) concept contains an interface between memory BIST (MBIST) logic and redundancy wrapper for storing faulty addresses.

Test Pattern Generator (TPG): Responsible for generating the test vectors according to the desired technique (i.e. depending upon the desired fault coverage and the specific faults to be tested for) for the CUT. Linear feedback shift register (LFSR) and pseudo random pattern generator (PRPG) are the most widely used TPGs.

Test Controller: Responsible for controlling the other components to perform the self test. The test controller places the CUT in test mode and allows the TPG to drive the circuit's inputs directly. During the test sequence, the controller interacts with the ORA to ensure that the proper signals are being compared. The test controller asserts its single output signal to indicate that testing has completed, and that the ORA has determined whether the circuit is faulty or fault-free.

Output Response Analyzer (ORA): Responsible for validating the output responses i.e. the response of the system to the applied test vectors needs to be analyzed. Also, a decision is made about the system being faulty or fault-free. LFSR and multiple input signature register (MISR) are the most widely used ORAs.

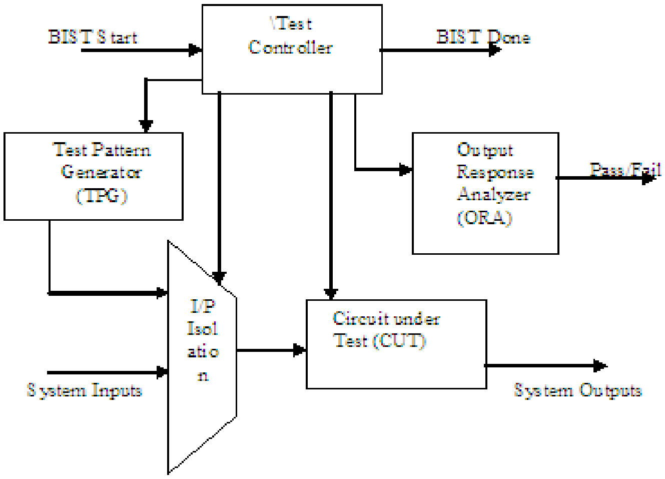

Fig 2.1: Basic BIST Architecture

2.1 Redundancy organization

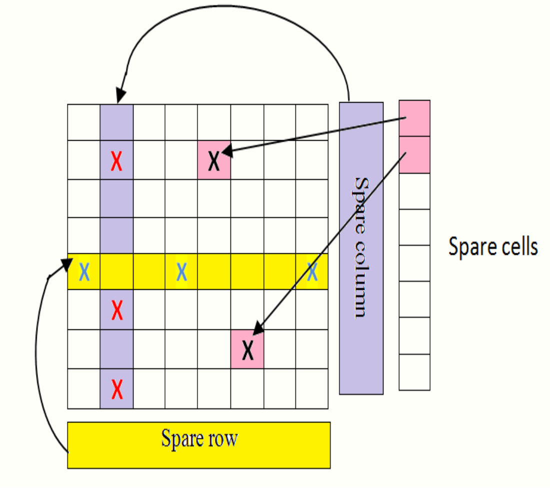

Fig 2.2: Redundancy Organization

The fig 2.2 shows an example of 8x8 main memory modules along with 1 spare row, 1 spare column and spare cells. Since most of the memory faults are single cell defects, here spare cells are used for better utilization of spare elements. The row/column having multiple defects is remapped with corresponding spare row/column. The single defects in the main memory are re mapped with spare cells. By this redundancy organization the area of spare is efficiently utilized.

1.2 Repairable RAM

A RAM with redundancies and reconfiguration circuit is called as a repairable RAM. Fig 2.2 depicts an example of an 8*8 bit-oriented RAM with 1 spare row and 1 spare column. If a spare row is allocated to replace a defective row, then the row address of the defective row is called row repair address (RRA). Then a decoder decodes the RRA into control signals for switching row multiplexers to skip the defective row if the row address enable (RAE) signal is asserted. The reconfiguration of the defective column and the spare column is performed in a similar way, i.e., give a column repair addresses (CRA) and assert the column address enable signal to repair the defective column using the spare column.

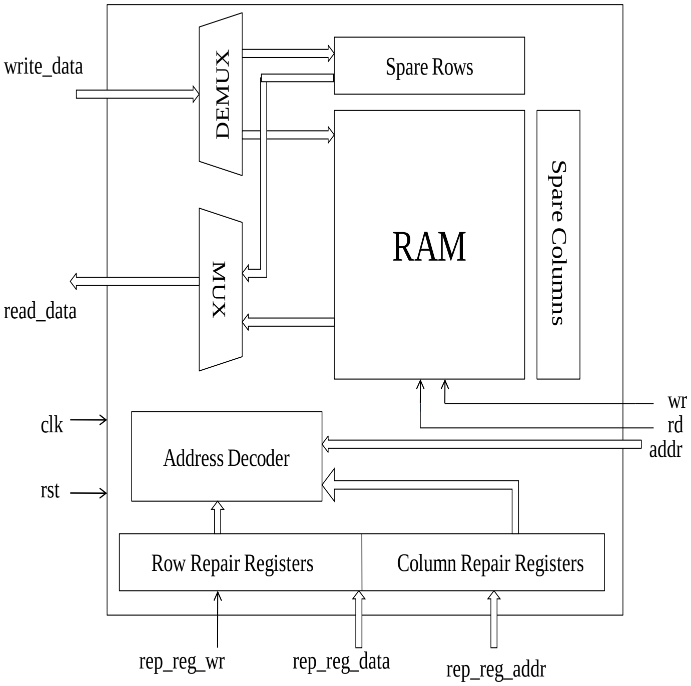

Fig 2.3: Repairable RAM

III. THE PROPOSED RE-BISR SCHEME

The block RAM details table is used for storing the configurations of RAMs which includes the memory data width, memory depth, number of spare rows and number of spare columns. RAM details table is of size 4*16. FSM is the main block that acts as a controller for generating the control signals during testing and repairing processes.

3.1Architecture of Re-BISR

The overall RAM Re-BISR flow is described as follows. Before the BIST circuit and the Re-BIRA circuit start testing and repairing the RAMs, the RAM configurations (details) should be known by these two circuits. This is done by the FSM, where it generates the necessary control signals that are required for sending the RAM configurations from RAM details table to BIST and Re-BIRA circuits. Because, the memory depth and memory data width are required by the BIST circuit for testing the RAM and the number of spare rows and number of spare columns are required by the Re-BIRA circuit to perform the analysis before repairing. The process of entering the RAM configurations into the RAM details table is described as follows.

When reset is high, all the locations in the RAM details table are filled with zeroes. Else, if the signal program_ram_details is high, the RAM details are entered into RAM details table through the pin ram_details using the write pointer. As the details are entered one by one, the write pointer is incremented by 1.Once the RAM details table is full, the write pointer stops incrementing and holds the value. Once the RAM details table is full, the BIST and Re-BIRA circuits start the testing and repairing processes of RAMs one by one. If the BIST circuit detects a fault, then the fault information is exported to the Re-BIRA circuitry, and then the Re-BIRA performs redundancy allocation on the fly using the rules of the implemented redundancy algorithm. The redundancy algorithm implemented in our scheme is Range Checking First Algorithm (RCFA). The Re-BIRA allocating redundancy on the fly means that the redundancy allocation process and the BIST process are performed concurrently. The proposed Re-BIRA scheme uses a local bitmap (i.e., a small bitmap) to store fault information of the faults detected by the BIST circuit. The bitmap or the fault table present in the Re-BIRA circuitry is of size 4*64 in our proposed Re-BISR scheme. Once the local bitmap is full, the BIST is paused and the Re-BIRA allocates redundancies according to the faulty information. After the Re-BIRA allocates a redundancy to repair a corresponding faulty row or column, the local bitmap is updated and the BIST is resumed. This process is iterated until the test and repair process is completed. The repair signatures from the Re-BIRA circuit are then sent to the repair registers that are present in the repairable RAMs. Repair signatures include repair register data (defective row/column address), repair register address and repair register write signal. The repairing procedure involves the entering of the repair register data in the repair registers. When the repair register write signal is high, then the repair register data is written in the repair registers at the address location specified by the repair register address. The BIST tests the RAMs once again (after the testing and repairing processes) to ensure that there are no faults present.

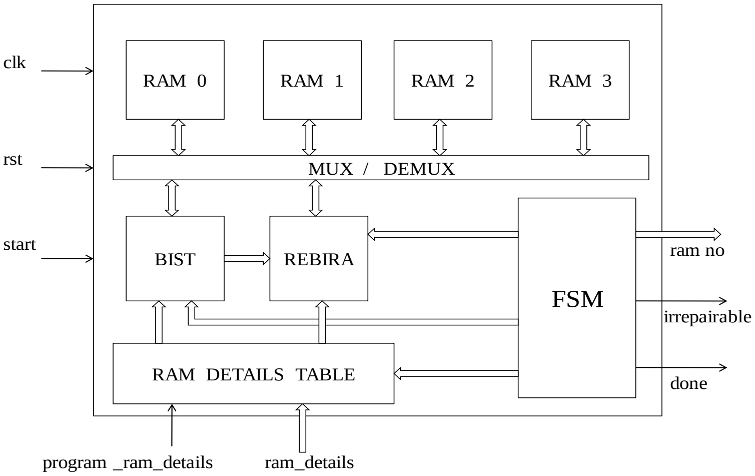

Fig 3.1: Block diagram of the proposed Re-BISR scheme for repairing multiple RAMs

Whenever the memory is accessed later i.e. after repairing, when the defective address is arrived, then the address decoder decodes the row/column repair address to control signals for switching row/ column multiplexers to skip the defective row/column. And, the control is immediately transferred to the relevant location either in the spare row/spare column since there is one to one correspondence between repair registers and spare elements. This is nothing but the address mapping procedure. The Re-BISR FSM is shown in the below figure and the Re-BISR flow using states is roughly described as follows. The Fig 3.1 shows the simplified block diagram of the proposed Re-BISR scheme for repairing multiple RAMs in an SOC. Four repairable RAMs with various sizes having different number of redundancies are considered in our proposed Re BISR scheme. All these RAMs are word oriented memories and their configurations are as listed in the table 3.1.

Table 3.1: Configurations of RAMs

RAM No	Data Width * Memory Depth	No. of Spare Rows	No. of Spare Columns
RAM0	16 * 32	2	2
RAM 1	32 * 64	2	3
RAM 2	128 * 64	3	2
RAM 3	256 * 64	0	0

The table 3.2 shows the stuck-at-faults in the four repairable RAMs at their respective fault row and fault column locations.

Table 3.2: Location of stuck-at-faults

RAM No	Fault Row	Fault Column	Stuck-at-Fault
RAM 0	6,7 14	18 28, 29	0 1,1
RAM 1	15, 16 28	43 61	1,1 0
RAM 2	-	-	-
RAM 3	243	31	0

3.2 RCFA: A Range-Checking First Algorithm for allocating 2D redundancy

One major feature of the RCFA is that the algorithm first checks the number of row entries and the number of column entries with fault information in the local bitmap. Then if the number of row entries with fault information is larger than the number of column entries with fault information, the algorithm allocates a spare column to replace the corresponding faulty column with the largest number of faulty bits. Otherwise, the algorithm allocates a spare row to replace the corresponding faulty row with the largest number of faulty bits. As figure shown above shows, the number of row entries (N_{FRE}) with fault information is 4. Also, the number of column entries with fault information in the left sub-bitmap and in the right sub-bitmap is 2 and 4, i.e., N_{FCEL}=2and N_{FCER}=4, where N_{FECL} and N_{FCER} denote the number of column entries with fault information in the left sub-bitmap and the right sub-bitmap, respectively.

Major steps of the redundancy allocation procedure of RCFA are shown in Algorithm.

1.Run BIST; pause and jump to Step 2 when it detects a fault.

2.Check whether the detected fault has been repaired. If so, go tostep1.Otherwise,go to Step3.

3.Check whether the local bitmap is full. If so, go to the next step. Otherwise,go to Step 1.

4.If $N_{FRE} > N_{FCEL}(N_{FCER})$, allocate an available spare column to replace the C_{MF}.

(If available spare column is exhausted, allocate an available spare row to replace the R_{MF}); else if $N_{FRE} < N_{FCEL}(N_{FCER})$, allocate an available spare row to replace the R_{MF} (if available spare row is exhausted, allocate an available spare column to replace the C_{MF}); else $N_{FRE} = N_{FCEL(}N_{FCER)}$, replace the faulty element with the largest number of faulty bits with a corresponding available spare element. If spare elements are exhausted, the RAM is irreparable.

5.Check whether the BIST is done. If so, go to the next step. Otherwise, go to Step 1 when the local bitmap is not full; go to Step 4 when the local bitmap is still full.

6.Check whether the local bitmap is empty. If so, export the repaired addresses and then stop.Otherwise,go to Step 4.

IV. SIMULATION RESULTS

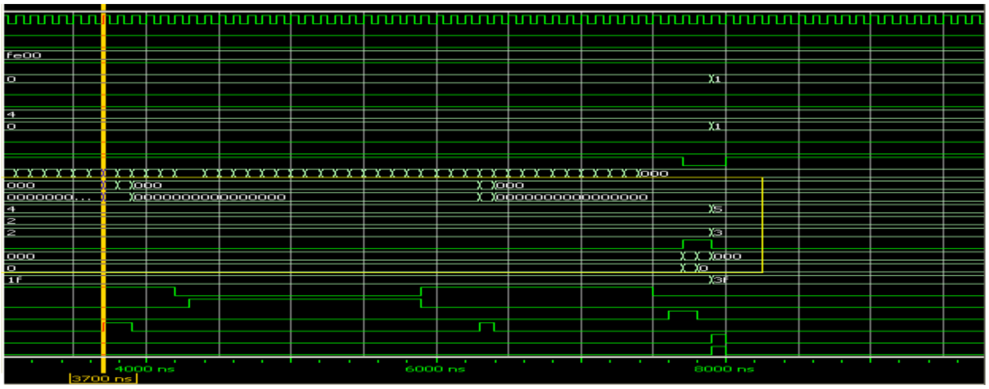

Fig 4.1: Reconfigurable BISR waveform

The RAM details are entered one by one and the RAM details for the RAM3 are 16’hfe00. The RAM details table write pointer is 4 as shown in the figure. When ram number is 0 and when run_mbist is 1, the faults are detected and given as the fault row in RAM0 is 10’h007 and fault column is 64’h0000000000040000 as shown in the figure. The repair signatures are repair_reg_wr is 1’b1, repair_reg_addr is 4’h2 and repair_reg_addr is 11’h412 for repairing fault at fault column 18.

V. CONCLUSION

A reconfigurable built in self repair circuit is efficiently implemented in this paper. The redundancy algorithm for 2D redundancy allocation is introduced. The BIRA is reconfigurable to support for multiple memories with different redundancies. So the area of the repair circuit is effectively reduced, i.e. nearly 60% of repair circuit area is reduced compare to individual repair circuits for 16x16, 32x32 and64x64 bits memory modules. Since the yield of memory plays major role in SOC designs, the proposed Re-BISR effectively increases compared to traditional yield.

REFERENCES

1. S. E. Schuster, Multiple word/bit line redundancy for semiconductor memories IEEE Journal of Solid- State Circuits, vol. 13, no. 5, pp.698–703.
2. M.Horiguchi,J.Etoh, M. Masakazu, K. Itoh, and T. Matsumoto, A flexible redundancy technique for high-density DRAM’s ,IEEE Journal of Solid-State Circuits, vol. 26, no. 1, pp. 12–17.
3. T. Yamagata, H.Sato, K. Fujita, Y.Nishimura and K. Anami, A distributed globally replaceable redundancy scheme for sub-half-micron ULSI memories, IEEE Journal of Solid-State Circuits,vol,31, no. 2, pp. 195–201, Feb. 1996.
4. Kim, Y. Zorian,G. Komoriya, H. Pham, F. P. Higgins, and J. L. Lweandowski, “Built in self repair for embedded high density SRAM,” in Proc. Int. Test Conf. (ITC), Oct. 1998, pp. 1112–1119.
5. M. Nicolaidis, N. Achouri, and S.Boutobza, “Optimal reconfiguration functions for column or data-bit built-in self-repair,” in Proc. Conf. Des., Autom., Test Eur. (DATE), Munich, Germany, Mar. 2003, pp. 590–595.
6. M. Nicolaidis, N. Achouri, and S. Boutobza, “Dynamic data-bit memory built-in self-repair,” in Proc. IEEE/ACM Int. Conf. Comput-Aided Des. (ICCAD), San Jose, CA, Nov. 2003, pp. 588–594.