An Efficient Permanent Fault Detection Method in FIFO Buffers of NoC Routers

Pallavi Mamidala
Assistant Professor, Dept of ECE, HITAM College, India.

K. Anil kumar
Assistant Professor, Dept of ECE, HITAM College, India.

Abstract – The Network-on-Chip (NoC) communication architecture is a packet based network where cores communicate among themselves by sending and receiving packets. High parallelism, smaller latency in data transmission and facility of Intellectual Property (IP) re-use have made NoCs overcome the problem of bandwidth and latency in conventional bus-based interconnects. In this brief proposes an on-line transparent test technique for detection of latent hard faults which develop in first input first output buffers of routers during field operation of NoC. The technique involves repeating tests periodically to prevent accumulation of faults. A prototype implementation of the proposed test algorithm has been integrated into the router-channel interface and on-line test has been performed with synthetic self-similar data traffic. The performance of the NoC after addition of the test circuit has been investigated in terms of throughput while the area overhead has been studied by synthesizing the test hardware. In addition, an on-line test technique for the routing logic has been proposed which considers utilizing the header flits of the data traffic movement in transporting the test patterns. In the proposed technique the bit size is four, and in the extension the size of the bit is eight.

Index Terms – FIFO buffers, in-field test, NoC, permanent fault, transparent test.

1. INTRODUCTION

Over the last decade, network-on-chip (NoC) has emerged as a better communication infrastructure compared with bus-based communication network for complex chip designs overcoming the difficulties related to bandwidth, signal integrity, and power dissipation. However, like all other systems-on-a-chip (SoCs), NoC-based SoCs must also be tested for defects. Testing the elements of the NoC infrastructure involves testing routers and inter router links. Significant amount of area of the NoC data transport medium is occupied by routers, which is predominantly occupied by FIFO buffers and routing logic.

Accordingly, the probabilities of run-time faults or defects occurring in buffers and logic are significantly higher compared with the other components of the NoC. Thus, test process for the NoC infrastructure must begin with test of buffers and routing logic of the routers. In addition, the test must be performed periodically to ensure that no fault gets accumulated. The occasional run-time functional faults have been one of the major concerns during testing of deeply scaled CMOS-based memories.

These faults are a result of physical effects, such as environmental susceptibility, aging, and low supply voltage and hence are intermittent (nonpermanent indicating device damage or malfunction) in nature. However, these intermittent faults usually exhibit a relatively high occurrence rate and eventually tend to become permanent. Moreover, wear-out of memories also cause intermittent faults to become frequent enough to be classified as permanent. Thus, there is a need for online test technique that can detect the run-time faults, which are intermittent in nature but gradually become permanent over time.

Chip integration has reached a stage where a complete system can be placed in a single chip. When we say complete system, we mean all the required ingredients that make up a specialized kind of application on a single silicon substrate. This integration has been made possible because of the rapid developments in the field of VLSI designs. This is primarily used in embedded systems. Thus, in simple terms a SoC can be defined as "an IC, designed by stitching together multiple stand-alone VLSI designs to provide full functionality for an application."

A NoC is perceived as a collection of computational, storage and I/O resources on-chip that are connected with each other via a network of routers or switches instead of being connected with point to point wires. These resources communicate with each other using data packets that are routed through the network in the same manner as is done in traditional networks.

It is clear from the definition that we need to employ highly sophisticated and researched methodologies from traditional computer networks and implement them on chip. We have to explore the motivating factors that are compelling the researchers and designers to move toward the adoption of NoC architectures for future SoCs.

2. FAULTS IN A NOC

A NoC is an on-chip communication infrastructure that implements multi-hop and predominantly packet-switched communication. Through pipelined packet transmission, NoCs permit a more efficient utilization of communication resources than traditional on-chip buses. Regular NoC structures reduce VLSI layout complexity compared to custom routed wires.

In future chip generations, faults will appear with increasing probability due to the susceptibility of shrinking feature sizes to process variability, age-related degradation, crosstalk, and single event upsets. To sustain chip production yield and reliable operation, very large numbers of faults will have to be tolerated Forecasts for future technologies suggest that component failure rates will be much higher than 0.1%.

If computers are to benefit from future advances in technology then there lie major challenges ahead, involving understand how to build reliable systems on increasingly unreliable technology and how to exploit parallelism increasingly effectively, not only to improve performance, but also to mask the consequences of component failure.

Rapid development in silicon technology is enabling the chips to accommodate billions of transistors. It has been observed however, that the current on-chip interconnects buses are becoming a bottleneck as they are unable to cope with growing number of participating cores on a chip. Shrinking silicon die size will lead to enhanced levels of cross talks, high field effects and critical leakage currents which, in turn, will lead to more temporary and permanent errors on chip. Crash or permanent failures can occur due to electro migration of a conductor or a connection failure permanently halting the operation of some modules.

On the other hand, faults like Gaussian noise on a channel and alpha particles strikes on memory and logic can cause one or more bits to be in error but do not cause permanent failures. Firstly, transient faults can corrupt individual packets causing them to be mis-routed or invalid, in which case a retransmission is required. Secondly, due to electro migration, cracks, or dielectric breakdowns, links and/or routers become permanently unavailable causing them to stop functioning.

The area of NoC is still in its infancy, which is one of the reasons why there are various names for the same thing; some call it on-chip networks, some networks on silicon, but the majority agrees upon "Networks on Chips" (NoCs). However, we will be using these terminologies interchangeably throughout our tutorial.NOC is Integrating various processors and on chip memories into a single chip .Faults occur in NOC.

- Permanent faults
- Transient fault

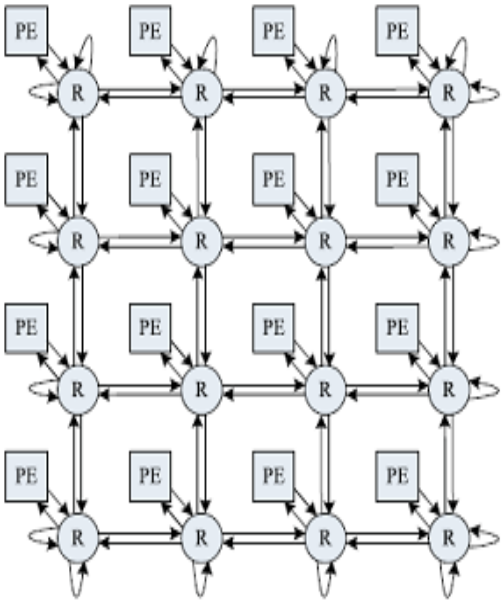


Fig 1 NoC Architecture

3. EXISTING SYSTEM

Transparent Test Generation

The faults considered in this brief, if applied for SRAMs or DRAMs, can be detected using standard March tests. However, if the same set of faults are considered for SRAM -type FIFOs, March test cannot be used directly due to the address restriction in SRAM-type FIFOs mentioned in and thus we were motivated to choose single-order address MATS++ test (SOA-MATS++) for the detection of faults considered in this brief. The word oriented SOA-MATS++ test is represented as {_ (wa); ↑ (ra,wb); ↓ (rb,wa)}; _ (ra)} where, a is the data background and b is the complement of the data background. ↑ and ↓ are increasing and decreasing addressing order of memory, respectively. _ means memory addressing can be increasing or decreasing. Application of SOA-MATS++ test to the FIFO involves writing patterns into the FIFO memory and reading them back. As a result, the memory contents are destroyed. However, online memory test techniques require the restoration of the memory contents after test. Thus, researchers ha ve modified the March tests to transparent arch test so that tests can be performed without the requirement of external data background and the memory contents can be restored after test. We have thus transformed the SOA-MATS++ test to transparent SOA MATS++ (TSOAMATS++) test that can be applied for online test of FIFO buffers. The transparent SOA -MATS++ test generated is represented as {↑ (rx,w¯ x, r ¯ x,wx, rx)}.

The transparent SOA-MATS++ algorithm is intended for test of stuck-at fault, transient fault, and read stuck-at fault, transition fault, and read disturb fault tests developed during

field operation of FIFO memories. The fault coverage of the algorithm is shown in Fig. 2. In both the figures, the word size of FIFO memory is assumed to be of 4 bits. As shown in Fig. 2,assume the data word present in lut be 1010.The test cycles begin with the invert phase (memory address pointer j with 0 value) during which the content of location addressed is read into temp and then backed up in the original.

The data written back to SOA-MATS++ test.lut is the complement of content of temp. Thus, at the end of the cycle, the data present in temp and original is 1010, while lut contains 0101. Assume a stuck-at-1 fault at the most significant bit (MSB) position of the word stored in lut. Thus, instead of storing 0101, it actually stores 1101 and as a result, the stuck-at-fault at the MSB gets excited. During the second iteration of j, when lut is readdressed, the data read into temp is 1101. At this point, the data present in temp and\original are compared (bitwise XOR ed). An all 1's pattern is expected as result. Any 0 within the pattern would mean a stuck-at fault at that bit position.

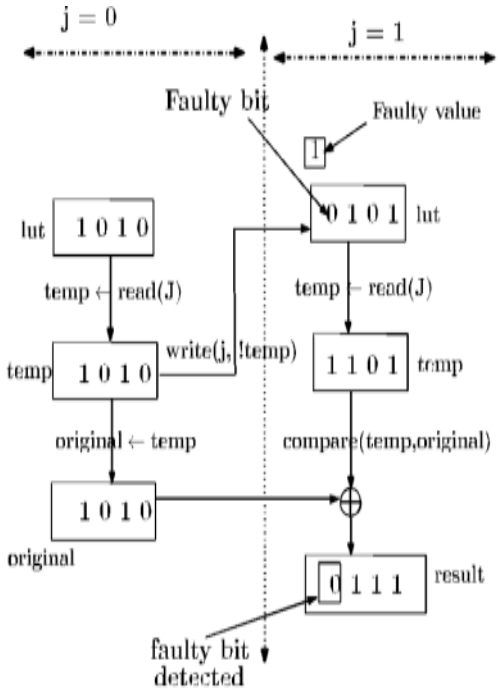


Fig 2 Fault detection during invert phase and restore phase of the transparent SOA-MATS++ test.

Where the XOR of 1010 and 1101 yields a 0 at the MSB position of the result indicating a stuck-at-fault at the MSB position. However, for cases where the initial data for a bit position is different from the faulty bit value, the stuck-at-fault cannot be detected for the bit position after the restore phase of the test. It thus requires one more test cycle to excite such faults.

4. PROPOSED SYSTEM

IMPLEMENTATION OF THE TEST ON FIFO BUFFERS OF NOC ROUTERS.

We present the technique used for implementing the proposed transparent SOA-MATS++test on a mesh-type NoC. Data packets are divided into flow control units (flits) and are transmitted in pipeline fashion. The flit movement in a mesh-type NoC infrastructure considered for this work is assumed to require buffering only at the input channels of routers. Thus, for a data traffic movement from one core to another, the online test is performed only on the input channel FIFO buffers, which lie along the path. The buffers operate in two modes, the normal mode and the test mode. The normal mode and test mode of operation of a FIFO buffer are synchronized with two different clocks. The clock used for test purpose (referred as test_clk in this brief) is a faster clock compared with the clock required for normal mode (router clock).

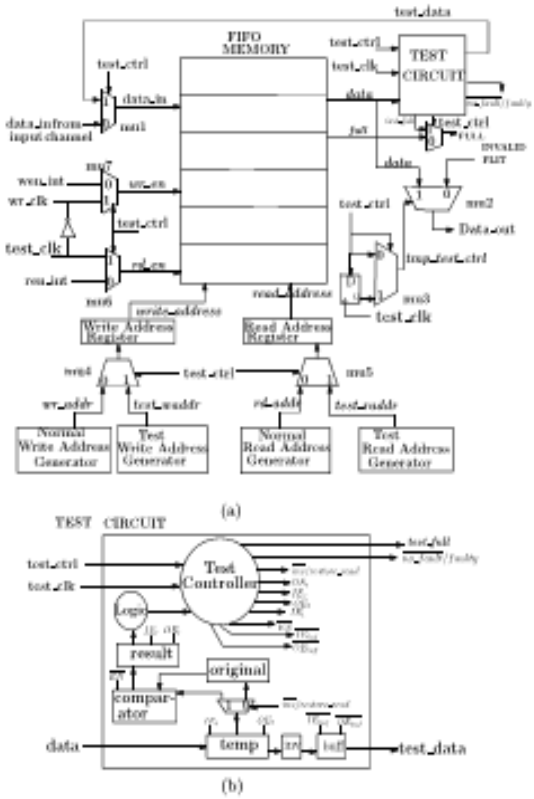


Fig.3. (a) Hardware implementation of the test process for the FIFO buffers.

(b)Implementation of test circuit.

The FIFO buffers are allowed to be operative in normal mode for sufficient amount of time before initiating their test process. This delay in test initiation provides sufficient time for run-time

intermittent faults developed in FIFO buffers to transform into permanent faults. The test process of a targeted FIFO buffer is initiated by a counter, which switches the FIFO buffer from normal mode to test mode. The switching of FIFO buffers from normal mode to test mode occurs after a certain period of time without caring about the present state of the FIFO buffer. It may be argued that at the instant of switching, the buffer may not be full, and as a result not all locations would be tested during the test cycle. However, test initiation after the buffer gets full would cause the following problems. First, wait for the buffer to get full would unnecessarily delay the test initiation process and would allow faults to get accumulated. Second, test of the entire buffer would prolong the test time and would negatively affect the normal mode of operation.

Test Architecture

The FIFO buffer present in each input channel of an NoC router consists of a SRAM-based FIFO memory of certain depth. During normal operation, data flits arrive through a data_in line of the buffer and are subsequently stored in different locations of the FIFO memory. On request by the neighboring router, the data flits stored are passed on to the output port through the data_outline. Fig. 3(a) shows the FIFO memory with data_in and data_out line. To perform the transparent SOA-MATS++test on the FIFO buffer, we added a test circuit, few multiplexers and logic gates to the existing hardware, as shown in Fig. 3(a). The read and write operations on the FIFO buffer are controlled by the read enable and write enable lines, respectively. The multiplexersmu6andmu7 select the read and write enable during the normal and test process. During normal operation when the test_ctrl is asserted low, the internal write and read enable lines, wen_intandren_int, synchronized with the router clock, provide the write and the read enable, respectively. However, during test process, the write enable and read enable are synchronized with the test clock. As mentioned earlier, the read and write operations during test are performed at alternate edges of a test clock. The read operations are synchronized with the positive edges, while the write_clk is obtained by inverting the test clock. In test mode (test_ctrl high), the test read and write addresses are generated by test address generators implemented using gray code counters similar to the normal address generation. Muxesm4andm5 are used to select between normal addresses and test addresses. Consider the situation when the FIFO buffer is in normal mode with flits being transferred from the memory to the data_out line. After a few normal cycles, the test_ctrl is asserted high, switching the buffer to test mode. As long as the buffer is in test mode, no external data is allowed to be written to the buffer, or in other words, the buffer is locked for the test period. As a result, the input data line for the FIFO memory is switched from the external data_inline to test_data line available from the test circuit. At the switching instant, the flit which was in the process of being transferred to the data_outline is simultaneously read into the Test Circuit. However, a one clock cycle delay is created for the flit to move to thedata_outline. This delay ensures that the flit is not lost during the switching instant and is properly received by the router, which requests for it. The single cycle delay in the path of the traveling flit is created by the D-type flip-flop and the multiplexerm3, as shown in Fig. 3(a). The flit, which is read in the test circuit, is stored in a temporary register temp and the test process begins with this flit.

Extension.

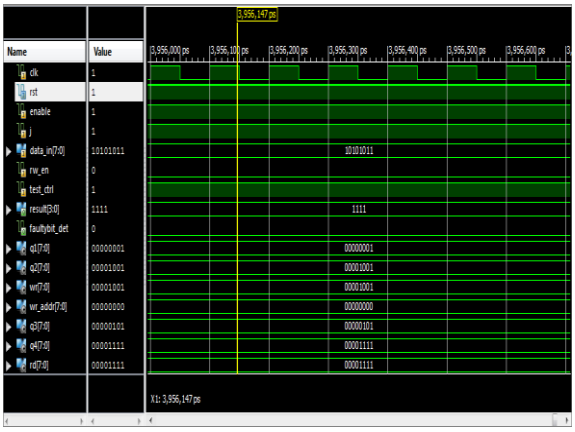
The technique used for implementing the proposed transparent SOA-MATS++test on a mesh-type NoC. The number of data bits used in the NoC is 4 bit. In the extension, the same technique is done by using 8 bits.

5. RESULTS

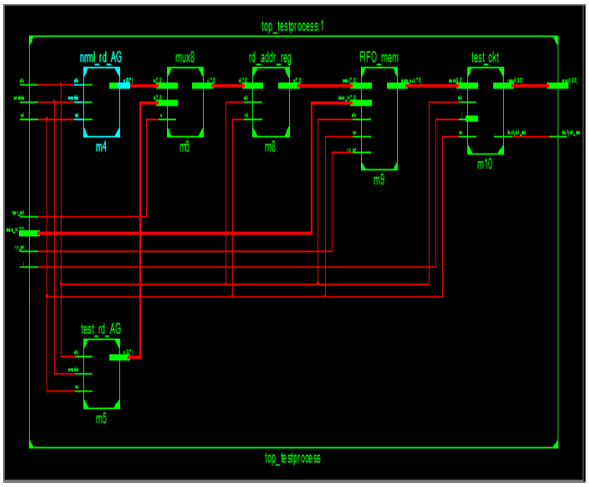
The Verilog HDL Modules have successfully simulated and synthesized using Xilinx ise13.2.

SIMULATION RESULT

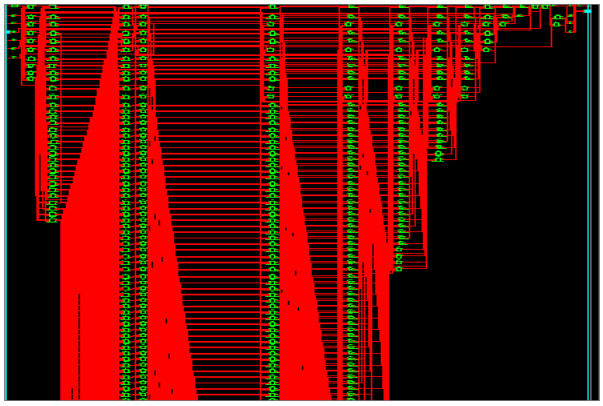
Proposed .



RTL SCHEMATIC:



TECHNOLOGY SCHEMATIC



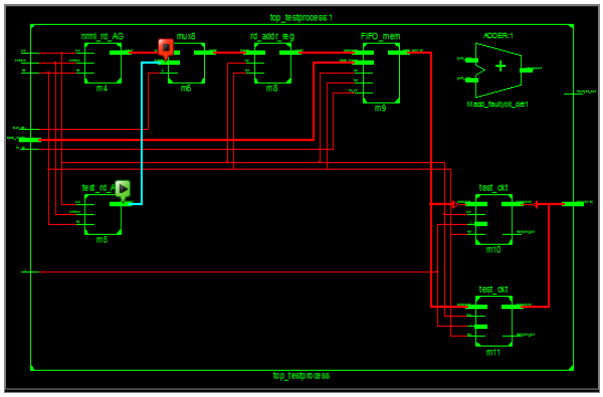
DESIGN SUMMARY

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	265	4656	5%
Number of Slice Flip Flops	291	9312	3%
Number of 4 input LUTs	434	9312	4%
Number of bonded IOBs	12	232	5%
Number of GCLKs	2	24	8%

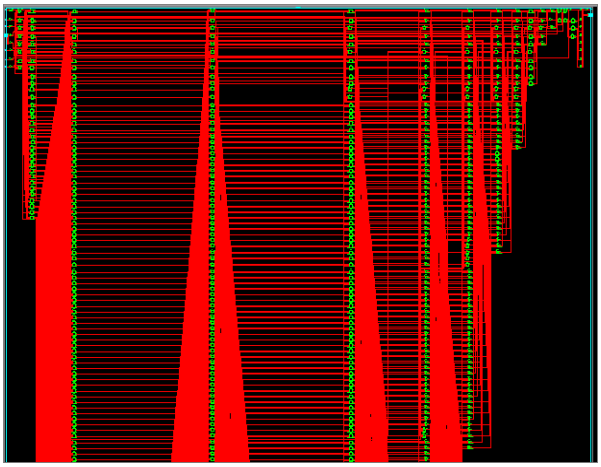
Extension.

Simulation.

RTL SCHEMATIC:



TECHNOLOGY SCHEMATIC



DESIGN SUMMARY

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	477	4656	10%
Number of Slice Flip Flops	547	9312	5%
Number of 4 input LUTs	568	9312	6%
Number of bonded IOBs	17	232	7%
Number of GCLKs	2	24	8%

6. CONCLUSION

In this brief, we have proposed transparent SOA-MATS++ test generation algorithm that can detect run-time permanent faults developed in SRAM-based FIFO memories. The proposed transparent test is utilized to perform online and periodic test of FIFO memory present within the routers of the NoC. Periodic testing of buffers prevents accumulation of faults and also allows test of each location of the buffer. Simulation results show that periodic testing of FIFO buffers do not have much effect on the overall throughput of the NoC except when buffers are tested too frequently. We have also proposed an online test technique for the routing logic that is performed simultaneously with the test of buffers and involves utilization of the unused fields of the header flits of the incoming data packets for test pattern encoding. As future work, we would like to modify the proposed FIFO testing technique that will allow incoming data packets to the router under test without interrupting the test.

REFERENCES

[1] W. J. Dally and B. Towles, "Route packets, not wires: On-chip interconnection networks," inProc. 38th Annu. Design Autom. Conf., 2001, pp. 684–689.

[2] A. Bondavalli, S. Chiaradonna, F. Di Giandomenico, and F. Grandoni, "Threshold-based mechanisms to discriminate transient from intermittent faults," IEEE Trans. Comput., vol. 49, no. 3, pp. 230–245, Mar. 2000.

[3] M. Radetzki, C. Feng, X. Zhao, and A. Jantsch, "Methods for fault tolerance in networks-on-chip," ACM Comput. Surv., vol. 46, no. 1, pp. 1–38, Jul. 2013, Art. ID 8.

[4] S. Ghosh and K. Roy, "Parameter variation tolerance and error resiliency: New design paradigm for the nanoscale era,"Proc. IEEE, vol. 98, no. 10, pp. 1718–1751, Oct. 2010.

[5] S. Borri, M. Hage-Hassan, L. Dilillo, P. Girard, S. Pravossoudovitch, and A. Virazel, "Analysis of dynamic faults in embedded-SRAMs: Implications for memory test," J. Electron. Test., vol. 21, no. 2, pp. 169–179, Apr. 2005.

[6] M. Bushnell and V. Agrawal, Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits(Frontiers in Electronic Testing). New York, NY, USA: Springer-Verlag, 2000.

[7] D. Xiang and Y. Zhang, "Cost-effective power-aware core testing in NoCs based on a new unicast-based multicast scheme," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 30, no. 1, pp. 135–147, Jan. 2011.

[8] K. Petersen and J. Oberg, "Toward a scalable test methodology for 2D-mesh network-on-chips," inProc. Design, Autom., Test Eur. Conf. Exhibit., Apr. 2007, pp. 1–6