In-Field Test for Permanent Faults in FIFO Buffers of NoC Routers

DHANAVATH KIRAN KUMAR NAIK
kiran00418@gmail.com

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 concise proposes an on-line straightforward test method for recognition of idle hard blames which create in first info initially yield cradles of switches amid field operation of NoC. The strategy includes rehashing tests intermittently to anticipate aggregation of deficiencies. A model execution of the proposed test calculation has been incorporated into the switch channel interface and on-line test has been performed with manufactured self-comparative information movement. The execution of the NoC after expansion of the test circuit has been researched as far as throughput while the region overhead has been examined by incorporating the test equipment. What's more, an on-line test system for the directing rationale has been proposed which considers using the header bounces of the information activity development in transporting the test designs.

Key words: *FIFO buffers, in-field test, NoC, permanent fault, transparent test.*

I.INTRODUCTION

Over the most recent decade, arrange on-chip (NoC) has developed as a superior correspondence foundation contrasted and transport based correspondence organize for complex chip plans beating the troubles identified with data transfer capacity, flag uprightness, and power scattering. Be that as it may, similar to every other framework on-a-chip (SoCs), NoC-based SoCs should likewise be tried for surrenders. Testing the components of the NoC foundation includes testing switches and bury switch joins. Critical measure of zone of the NoC information transport medium is possessed by switches, which is transcendently involved by FIFO supports and steering rationale. In like manner, the probabilities of run-time blames or imperfections happening in cushions and rationale are fundamentally higher contrasted and alternate segments of the NoC. Therefore, test process for the NoC framework must start with trial of cushions and directing rationale of the switches. What's more, the test must be performed intermittently to guarantee that no blame gets gathered. The infrequent run-time utilitarian shortcomings have been one of the real worries amid testing of profoundly scaled CMOS-based recollections.

These flaws are a consequence of physical impacts, for example, ecological weakness, maturing, and low supply voltage and henceforth are discontinuous (nonpermanent showing gadget harm or breakdown) in nature. In any case, these irregular blames for the most part display a generally high event rate and in the end have a tendency to end up noticeably changeless. Besides, destroy of recollections likewise make discontinuous deficiencies wind up plainly visit enough to be named perpetual. In this manner, there is a requirement for online test strategy that can identify the run-time flaws, which are irregular in nature yet steadily end up noticeably lasting after some 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 researchers and designers to move toward adoption of NoC architectures for future SoCs.

II.FAULTS IN A NOC

A NoC is an on-chip correspondence framework that executes multi-jump and prevalently parcel exchanged correspondence. Through pipelined bundle transmission, NoCs allow a more productive usage of correspondence assets than conventional on-chip transports. General NoC structures diminish VLSI design unpredictability contrasted with uniquely steered wires.

In future chip ages, deficiencies will show up with expanding likelihood because of the helplessness of contracting highlight sizes to process fluctuation, age-related debasement, crosstalk, and single occasion upsets. To maintain chip creation yield and solid operation, huge quantities of shortcomings should be endured Forecasts for future innovations recommend that segment disappointment rates will be considerably higher than 0.1%. On the off chance that PCs are to profit by future advances in innovation at that point there lie significant difficulties ahead, including see how to manufacture solid frameworks on progressively untrustworthy innovation and how to misuse parallelism progressively viably, to enhance execution, as well as to cover the results of segment disappointment. Quick improvement in silicon innovation is empowering the chips to oblige billions of transistors. It has been watched nonetheless, that the current on-chip interconnects transports are turning into a bottleneck as they can't adapt to developing number of taking an interest centers on a chip. Contracting silicon pass on size will prompt improved levels of cross talks, high field impacts and basic spillage streams which, thus, will prompt more brief and lasting mistakes on chip. Crash or lasting disappointments can happen due to electro relocation of a conductor or an association disappointment for all time stopping the operation of a few modules. Then again, deficiencies like Gaussian commotion on a channel and alpha particles strikes on memory and rationale can make at least one bits be in mistake however don't cause perpetual disappointments. Right off the bat, transient issues can degenerate individual bundles making them be mis-steered or invalid, in which case a retransmission is required. Besides, due to electro relocation, splits, or dielectric breakdowns, joins or potentially switches turn out to be for all time inaccessible making them quit working.

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

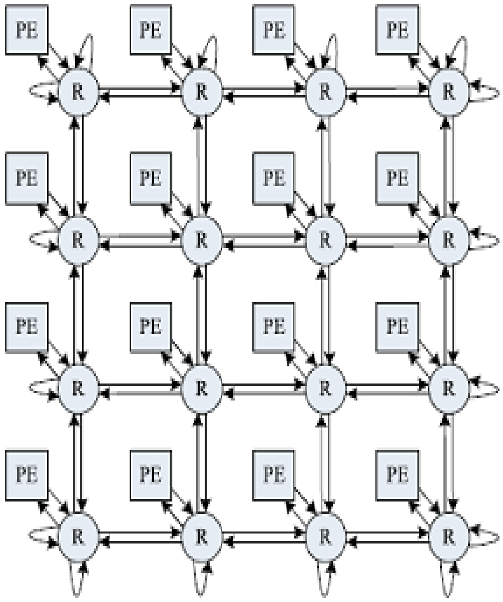

Fig 1: NoC Architecture

III.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.

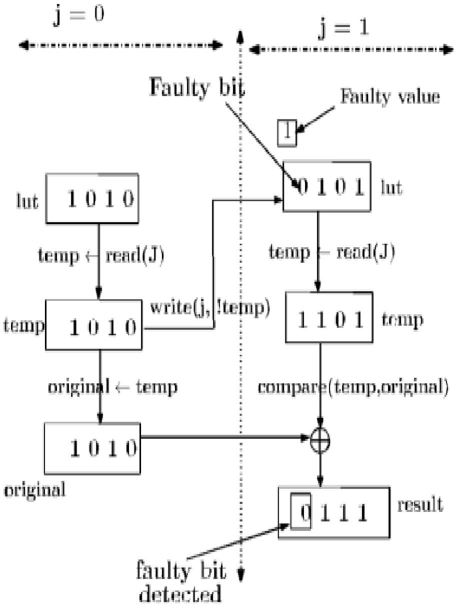

Fig 2 Fault detection during invert phase and restore phase of the transparent SOA-MATS++ test.

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.

IV.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).

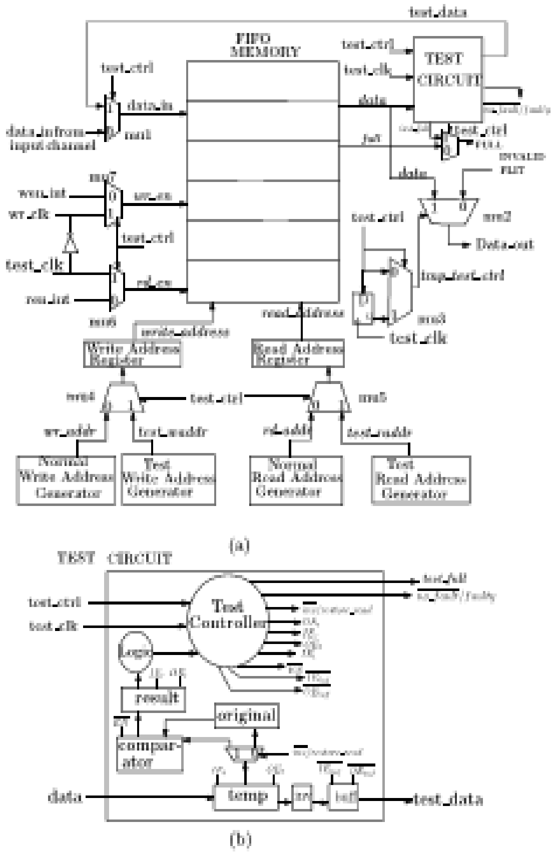

Fig.3: (a) Hardware implementation of the test process for the FIFO buffers. (b)Implementation of test circuit.

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 support introduces in each info channel of a NoC switch comprises of a SRAM-based FIFO memory of certain profundity. Amid typical operation, information flutters touch base through a data_in line of the cushion and are consequently put away in various areas of the FIFO memory. On ask for by the neighboring switch, the information dances put away are passed on to the yield port through the data_outline. Fig. 3(a) demonstrates the FIFO memory with data_in and data_out line. To play out the straightforward SOA-MATS++test on the FIFO cushion, we included a test circuit, couple of multiplexers and rationale doors to the current equipment, as appeared in Fig. 3(a).

The read and compose operations on the FIFO support are controlled by the perused empower and compose empower lines, individually. The multiplexersmu6andmu7 select the read and compose empower amid the ordinary and test process. Amid ordinary operation when the test_ctrl is affirmed low, the interior compose and read empower lines, wen_intandren_int, synchronized with the switch clock, give the compose and the read empower, individually. Be that as it may, amid test process, the compose empower and read empower are synchronized with the test clock. As said before, the read and compose operations amid test are performed at substitute edges of a test clock. The read operations are synchronized with the positive edges, while the write_clk is gotten by modifying the test clock. In test mode (test_ctrl high), the test read and compose addresses are created by test address generators executed utilizing dim code counters like the typical address age. Muxes m4 and m5 are utilized to choose between typical locations and test addresses.

Consider the circumstance when the FIFO support is in ordinary mode with flutters being exchanged from the memory to the data_out line. After a couple of typical cycles, the test_ctrl is affirmed high, changing the support to test mode. For whatever length of time that the cushion is in test mode, no outside information is permitted to be composed to the support, or at the end of the day, the cradle is bolted for the trial.

Therefore, the info information line for the FIFO memory is changed from the outer data_inline to test_data line accessible from the test circuit. At the exchanging moment, the dance which was being exchanged to the data_outline is all the while read into the Test Circuit. Be that as it may, a one clock cycle delay is made for the dance to move to the data_outline. This postponement guarantees that the bounce isn't lost amid the exchanging moment and is appropriately gotten by the switch, which demands for it. The single cycle delay in the way of the voyaging bounce is made by the D-sort flip-flounder and the multiplexerm3, as appeared in Fig. 3(a). The bounce, which is perused in the test circuit, is put away in a brief enroll temp and the test procedure starts with this dance.

V.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.

DHANAVATH KIRAN KUMAR NAIK received his Bachelor’s degrees in Electronics and communication from Nagarjuna Institute of Technology and Sciences. He received his Master’s degree in VLSI system design from Swami Ramananda Tirtha Institute of Science and Technology. He is currently working as a executive engineer in WAPCOS limited(Govt.of India).His current research interests include very large scale integration (VLSI) low power design, test automation and fault-tolerant computing.