

# Charka: A Reliability-Aware Test Scheme for Diagnosis of Channel Shorts Beyond Mesh NoCs

Biswajit Bhowmik, Jantindra Kumar Deka, Santosh Biswas  
 Department of Computer Science and Engineering  
 Indian Institute of Technology Guwahati  
 Assam, India, 781039  
 Email: {b.bhowmik, jatin, santosh\_biswas}@iitg.ernet.in

**Abstract**—This paper presents a fast and low cost on-line scheme named *Charka* that analyzes short faults in channels of octagon NoCs. Experimental results demonstrate that the proposed scheme achieves 100% coverage metrics and its on-line evaluation reveals compelling effect of these faults on system performance. We observe that the proposed scheme is upto 9X faster while packet latency is improved by 13.79–21.17% and energy consumption is reduced by 17.57–24.97%. Further, the test area overhead is reduced by 13–26% that shows 52–57.77% improvement.

**Keywords**—channel short faults; design for manufacturability, yield, reliability, and defect tolerance; on-line fault detection and diagnosis; model effectiveness; on-line performance evaluation.

## I. INTRODUCTION

With the advancements in VLSI technology, SoC designs are seamlessly integrated with IP blocks. But, a bus-based SoC system has failed to provide expected performance requirements by several complex applications. A NoC in the place has emerged as a new methodology that encompasses networking paradigms and allows effective addition of storage and functional blocks [1]. However, a NoC channel outlined with a group of metallic wires is often susceptible to logic level manufacturing faults e.g. shorts. A short fault whether intra- or inter-channel [2], [3] when experienced in interswitch and local channels, the issue of establishing a reliable connection between sender and receiver in a network (NoC) becomes a major concern. The network performance as a result is influenced noticeably. Additionally, test time as a driving factor may influence the performance when the channels are in test mode.

Many researchers have shown their interests in testing short faults in communication channels of a NoC-based architecture. Cota *et al.* [4] have addressed pairwise shorts on channel-wires in a  $2 \times 2$  mesh neighborhood. Multiple applications of the neighborhood in larger mesh NoC cover the fault. Though, the model has achieved good fault coverage metrics for meshes but fails for other topologies e.g. octagon. Then, a test model discussed in [5] must be taken. The model consequently leads to high test time. In on-line mode, Kakoe *et al.* [6] have analyzed stuck-at faults (including pairwise shorts) in interswitch channels of general NoCs. Though, the model is scalable irrespective of NoCs but offers high test time. Because, every router is selected sequentially to test its channels. Bhowmik *et al.* [2] have proposed a time efficient

on-line test mechanism for channel-shorts. The model well behaves in large-scale general NoCs but fails for smaller network. Because, the model is based on selection of sixteen nodes at a time and brings the test application on a smaller NoC equivalent to an off-line mode application. Above all, these modules are discussed with mesh NoCs. Therefore, an efficient methodology must be devised that have a capability of detecting and diagnosing channel-shorts in other NoCs e.g. octagon [7] and is mandatory in design for manufacturability, yield, reliability, and defect tolerance improvement.

This paper presents a cost effective and fast on-line scheme named *Charka* that tests intra- and inter-channel short faults in an octagon NoC (Figure 1). We consider the faults on data, control, and handshake wires of interswitch and local channels. The proposed scheme schedules the nodes (routers and their dedicated cores) so that the channels in the NoC can be tested in just few clocks. The NoC during testing is kept functional except the channels of scheduled nodes under test.

We see that the experimental synthesis needs few logic gates as hardware area overhead to implement the test algorithm and fault simulation achieves 100% coverage metrics. The on-line evaluation of the proposed scheme reveals deep insights on the effect of these channel short faults on system performance. We observe that the proposed scheme reduces the hardware area overhead by 13–26% that shows 52–57.77% improvement and is up to 9X faster. Simultaneously, average packet latency is improved by 13.79–21.17% and energy consumed by a packet flit is reduced by 17.57–24.97%.

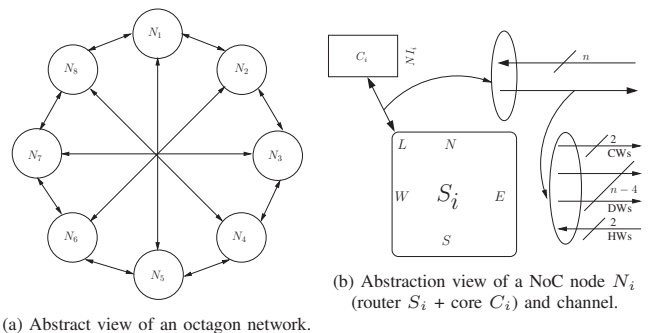


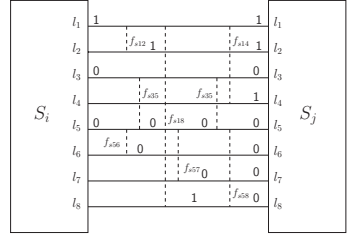
Fig. 1: General representation of an octagon network with nodes and  $n$ -bit channels. The DW, CW, HW are data, control, and handshake wire sets respectively.

Rest of the paper is organized as follows. Proposed test model is discussed in Section II. Section III provides sim-

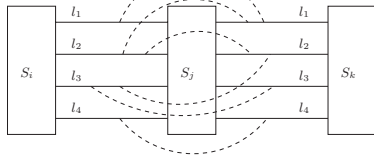
ulation results. Countering the prior models is shown in Section IV. Section V concludes the paper.

## II. PROPOSED WORK

In this section, we present our test model for intra- and inter-channel short faults experienced in an octagon NoC. First, we define the short fault model followed by test infrastructures. Next, we discuss the test mechanism that exploits the walking-one (W1) sequence to testify our channel-shorts. Finally, we propose a node selection method to lower the test time.



(a) Intra-shorts in the channel ( $S_i, S_j$ ).



(b) Inter-shorts between channels ( $S_i, S_j$ ) and ( $S_j, S_k$ ).

Fig. 2: Representation of intra- and inter-channel shorts with single/multiple occurrences.

### A. Short Fault Model

In this work, we consider shorts, one kind of logic level manufacturing faults in channels of an octagon network. A short is a state that is experienced by two or more channel-wires because of their closeness. A short occurs in a group  $fg$  of channel-wires. We term the  $fg$  as fault group. Short faults are intra- and inter-channel types [2], [3] shown in Figures 2a and 2b [8]. The symbol  $f_{spq}$  represents a short between wires  $l_p, l_q$  with  $fg$  instances in a channel. E.g. the wires  $l_1, l_2, l_4, l_8$  (Figure 2a) are shorted together with  $fg = 4$ . As the  $fg$  increases, occurrences of *intra-channel* shorts decrease. It is observed that this short beyond  $fg > 4$  is negligible. Further, two channels in an NoC architecture are sufficiently separated. The probability of *inter-channel* shorts is also negligible. We thus realistically consider  $fg = 2, \dots, 5$  for intra-channel shorts and pairwise ( $fg = 2$ ) inter-channel shorts at a node for the analysis.

$$\#S1 = \sum_{fg=2, \dots, 5} \frac{n!}{fg!(n-fg)!} \quad (1)$$

$$\#S2 = \frac{(8n)!}{2!(8n-2)!} \quad (2)$$

$$\#S = \begin{cases} 40 \times \#S1 \\ + \\ 8 \times \#S2 \end{cases} \quad (3)$$

Each node in an octagon NoC shares 8 unidirectional channels, each has  $n$ -bit width. So, a node shares  $8n$  channel-wires. Equations 1 and 2 define  $\#S1$  and  $\#S2$  as the size of intra- and inter-channel shorts at an octagon node respectively. An octagon network consists of 8 nodes and 40 unidirectional channels. Then, total shorts  $\#S$  modeled in the network is equated in Equation 3. Our analysis of the shorts is with 16-bit channels. Hence,  $\#S1=274720$ ,  $\#S2=65024$  i.e.  $\#S=339744$  short faults are modeled in our octagon NoC for testing.

### B. Test Infrastructures for Channel Shorts

The proposed test algorithm implementation involves test sequence generation, transmission, and response analysis. We use the W1 sequences as test pattern set. The derived test set is sufficient to detect all possible shorts in channels. A test pattern generator (TPG) block generate raw W1 set including header and trailer. These data set is segmented into small packets. Each test packet is shifted in time by the TPG block. The TPG at a core unicasts test packets. We add extra circuits in the logic of router's TPG. This integrated TPG (I-TPG) multicasts test packets. The received test vectors as test responses are analyzed by another block- test response analyzer (TRA). A TRA block has two sub-blocks. One block is called fault diagnosis module (FDM) that analyzes test responses to diagnose shorts on channel-wires. Another block is called TRA signal generator (TSG) that declares a channel error- payload, misrouting, and dropping depending upon diagnosis result from the FDM. Towards lowering the test clocks and performance overhead these two blocks as a test module (TM) like our early works including [8], [9] are placed side by side in both router and core of every node in the underlying octagon network. Figure 3 gives an abstract view of a TM in a node.

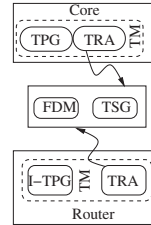


Fig. 3: Basic test blocks embedded in both core and router of a node.

TABLE I: Test packet organization for a channel with data-width=2.

	Payload		T r a i l e r
	$dw_2$	$dw_1$	
H	0	1	
e	1	0	
a	0	0	
d	0	0	
e	0	0	
r	0	0	
	0	0	

An octagon NoC-based communication system is a packet switched network where two communication parties exchange data in the form of packets. Every packet has header, payload, and trailer flits. The header and trailer flits contain routing information such as beginning of packet (bop), destination address, end of packet (eop) and so on. We place W1 sequences in these fields. The payload flits contain W1 sequences for DWs. The test set must be accommodated with control information carried with header and trailer flits at the moment to test CWs. Also, required W1 set is placed in handshake information while testing HWs. Size of a test as well as normal packet varies with the  $n$ , routing scheme, and so on. A packet

with larger size is believed to be disadvantageous to NoC performance. Each packet size contains four flits (2 payload, 1 header, 1 trailer flits) to evaluate our fault model. A typical format of a test packet for data wires is provided in Table I. The  $dw_1, dw_2$  are first and second data-wires in a channel. Since, every test flit is dedicated for a wire in a channel. For an  $n = 16$ -bit channel, we need 16 test flits of W1 type. Then, we need 8 test packets to address shorts in channels from a node.

### C. Testing Short Faults in Channels

Now, we discuss the test mechanism that involves detecting and locating both intra- and inter-shorts in channels from a node. The mechanism exercises test packets, each consists of partial W1 set embedding with header and trailer flits like as the wormhole switching does. The header and trailer are necessarily applied. The payload follows the header and the trailer follows the payload.

Similar to authors early works including [8], [9], the proposed test mechanism for channels from a node is based on shifting of test packets from the node by its TPG module and analyzing the received test packets at the neighbors by their TRA modules. The wires of a channel are tested in three phases: DWs, CWs, and HWs. The first phase is accomplished by loading sufficient W1 set into payload fields of the packets. The short faults affecting CWs and HWs are handled one after another by accommodating sufficient W1 sets in the control ("bop" and "eop") and handshake ("ack" and "val") information dedicated for these wire sets. Note that the TPG at core of the node shifts test packets to its neighbor router over the local channel. At the same time, I-TPG of router of the node multicasts test packets on the channels from the router to its connected core and neighbor routers. A test packet is shifted from a node in a such a way that a underlying channel at any time remains filled. To serve the purpose, logic of a wire is set to "1" while remaining wires in the channel are set to "0". If a wire for example  $l_p$  is shorted with another wire  $l_q$  in the channel, then the test bit "1" sent on the  $l_p$  is also available on the  $l_q$ . The TRA's input port at a receiver over the wires detects this short fault between these wires.

Alongside the detection of short faults among wires in channels considered in a test mode, the location of the faults on channel-wires that indicate the set of faulty wires in these channels must be identified. The FDM of the TRA at a receiver analyzes the incoming test flits i.e. received W1 sequences (responses) with predefined local test sets. The FDM essentially verifies whether these responses correspond to the local test flits, W1 set generated by the TPG/I-TPG at the receiver. To diagnose the shorts in a channel i.e. identify the faulty wires in the channel, the FDM performs the following logical operations: *XORing*, *ORing* separately between a response and its corresponding test flit. For instance consider Figure 2a where the test bit "1" is sent from the router  $S_i$  on the  $l_1$  in the form of test flit "1000 0000". Then, the TRA's input port at the receiver  $S_j$  has responses in the form "1000 0000", "0100 0000", "0001 0000", and "0000 0001" over the

$$\begin{matrix} \oplus & \left\{ \begin{array}{l} 10000000 \\ 01000000 \\ 00010000 \\ 00000001 \\ 10000000 \\ \text{---} \\ 01010001 \end{array} \right. & (4)+ & \left\{ \begin{array}{l} 10000000 \\ 01000000 \\ 00010000 \\ 00000001 \\ 10000000 \\ \text{---} \\ 11010001 \end{array} \right. \end{matrix}$$

Fig. 4: Meaning of TRA's 2-bit signal.

Bits	Error Type
00	No Error
01	Payload Error
10	Misrouting Error
11	Timeout Error

$l_1, l_2, l_4, l_8$ . It shows that the transmitted bit is duplicated. Now, the FDM of the TRA does the logical operations provided in Equations 4 and 5. The former operation ensures that the wires-  $l_2, l_4, l_8$  are shorted with the  $l_1$ . The later operation ensures shorted wires including the  $l_1$ . Either of these results is fed to the TRA's TSG that generates a 2-bit signal to ensure the type of error occurred in the channel. The meaning of the signal bits is provided in Table 4. If the  $l_1$  is data- and handshake-wire experienced a short, a payload error occurs which later is manifested as packet duplication at system level. Diagnosing shorts in control-wires results in packet misrouting and dropping errors. Since a test packet is guided by header and trailer flits, then shorts on CWs affect these flits. Thus, packet header and trailer information get modified and are analyzed by the TRA over the channel. During application, this modified header will cause to transmission of the packet to a unintended receiver resulting packet misrouting. Similarly, the intended receiver may not get the "eop" signal due to modified trailer that results to packet dropping error.

### D. Scheduling Nodes

We apply the proposed scheme in on-line mode where a subset of channels is made under test and keep rest part of underlying octagon network functional that may transmit application packets. However, an application packet must wait at a router which is at the moment engaged in testing a channel and needs forwarding of the packet of course on the channel. Major challenge is the selection of a subnet (a set of channels connected by nodes) in test mode. A subnet selection must involve in lowering the test cost (time) and performance overhead. This task is referred as test scheduling. A test setup of addressing shorts in channels of a subnet is treated as a test configuration/round.

In the proposed test scheduling, we select four nodes from the underlying octagon network that constructs a closed X-shaped subnet. Such subnet is rotated in the clockwise or anticlockwise direction that completes testing of channels of the octagon network in just four test rounds (TRs). A test round is accomplished in two test iterations (Tits). In first iteration (Tit=1), all odd nodes initiate shifting of test packets while the rest even nodes analyze the test responses. Similarly, the even nodes send test packets and the neighbor odd nodes analyze the received packets in the next iteration i.e. second iteration (Tit=2). An odd/even node is found in [2].

The test algorithm is executed at a node and multiple nodes are concurrently allowed to do so in a Tit. The test time  $T_{it}$  needed in an iteration is therefore equal to the test time  $T_{node}$  at a node. Now, the  $T_{node}$  as found in authors earlier works,

is the sum of time required for test set including header and trailer generation labeled as  $T_{tpg}$ , test set organization into packets labeled as  $T_{tpo}$ , test packet transmission labeled as  $T_{tpt}$ , and test response analysis labeled as  $T_{tra}$ . Equation 6 defines the  $T_{node}$ . Thus,  $T_{subnet}, T_{n/w}$ , the time needed for channels of a subnet and an octagon network can be determined in Equations 7, 8 respectively [8], [9].

$$T_{node} = T_{tpg} + T_{tpo} + T_{tpt} + T_{tra} \quad (6)$$

$$T_{subnet} = 2T_{node} \quad (7)$$

$$T_{n/w} = 4T_{subnet} \quad (8)$$

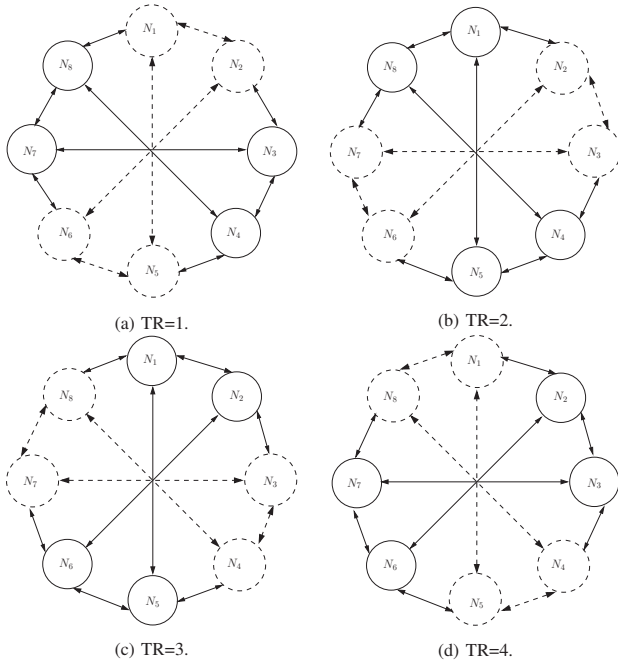


Fig. 5: Application of the proposed test mechanism at various test rounds in an octagon network.

Figure 5 illustrates test application in a subnet denoted by dotted region. The subnet is rotated here in the clockwise direction. And, the application of the proposed test algorithm by the odd/even nodes in first subnet i.e. at TR=1 is shown in Figure 6. Figures 6a and 6b illustrate the test application at Tit=1 and Tit=2 respectively. In the TR=1, Tit=1, the  $N_1 \leftarrow \langle S_1, C_1 \rangle, N_6 \leftarrow \langle S_6, C_6 \rangle$  (Figure 6a) are treated as odd nodes that transmit test packets over channels (dotted arrows) to their neighbors even nodes where responses are analyzed to detect and diagnose shorts. Similarly, the  $N_2 \leftarrow \langle S_2, C_2 \rangle, N_5 \leftarrow \langle S_5, C_5 \rangle$  (Figure 6b) treated as even nodes transmit test packets over channels (dotted arrows) to their neighbor odd nodes in the next iteration of the current round i.e. TR=1, Tit=2.

### III. RESULTS

The proposed test strategy is implemented on an octagon NoC. The network consists of 8 routers, 8 cores, and 40

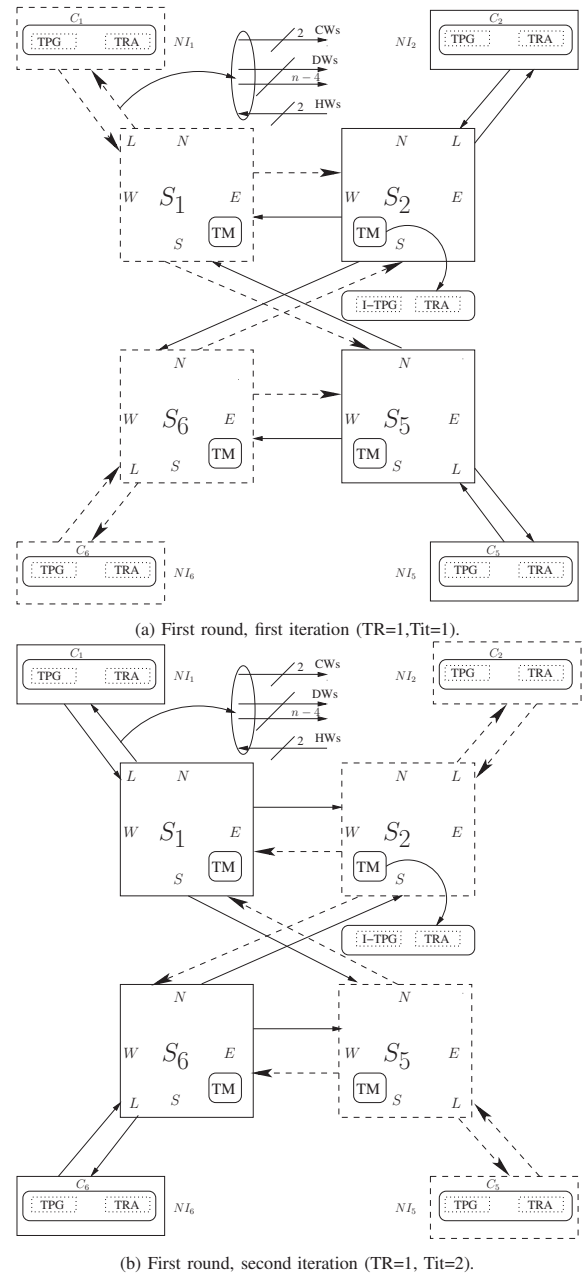


Fig. 6: Scheduling test executions at odd/even nodes in an iteration.

channels. Each router is connected to a core via local channel and three neighbor routers via interswitch channels. We consider in this paper that each channel has width  $n = 16$ -bit. Also, the  $n$  can be determined on the basis of application type, traffic size in the network, and so on. The channels with injected faults in a subnet (or test round) are placed under test. The TPGs at one end of the channels generate, organize, and transmit test packets. The TRAs at other end of the channels analyze responses to detect and identify faulty channel-wires. Experimental setup is built with Xilinx version 10.3 simulator and the Verilog is used to synthesize these TPGs and TRAs. The size of these units are measured in terms

TABLE II: Synthesis Results for the TM.

$n$	TM Unit	#LGs	RASoC [10]	Xpipe [11]	Core
16	TPG-Core	118	7%	8%	2.5%
	TRA-Core	84	5%	6%	1.5%
	I-TPG	152	9%	11%	3%
	TRA-Router	124	7%	8%	2%
	Total	202-276	12-16%	14-19%	4-5%

of logic gates (LGs) that vary with channel width. Since few W1 test sequences are used, these units take very little space in routers and cores. Table II provides the synthesis results of TPG and TRA (TM) units of a node for a 16-bit channel. The space needed by a TM is measured with reference to RASoC [10] and Xpipe [11] routers. Since, the size of a core is much more than its router, the TM insists very little area overhead on the core than a router.

#### A. Model Effectiveness

Considering the test time evaluation method discussed in [2], [8], [9], each iteration needs 40 clocks ( $T_{tpg} = 5, T_{tpo} = 1, T_{tpt} = 32, T_{tra} = 2$ ) to detect shorts in a channel of  $n = 16$ -bit. Therefore, the  $T_{n/w}$  equals to 320 clocks. Our fault injection followed by simulation campaigns are done in the sequence of DWs, CWs, and HWs in a test round. Let us consider the test application of TR=1 shown in Figure 5a that shows 16 channels under test. In first fault simulation, we assume the shorts exist in DWs. Then, #S1=25168, #S2=10224 i.e. #S=35392 are detected by transmitting sufficient W1 test sequences and analyzing responses. In second fault simulation, shorts are assumed to be on CWs and the fault region is extended to DWs. Additional W1 sequences are accommodated in control information ("bop, eop" signals). The fault simulation campaign detects #S1=55328, #S2=13944 i.e. #S=69272. Finally, we consider shorts on HWs and now the fault region is extended to both DWs and CWs. Therefore, extra W1 set is placed in the handshake communication information ("val, ack" signals). The fault simulation detects #S1=109888, #S2=18240 i.e. #S=128128. In the current test round, we test 16 channels that achieve 40% link (channel) coverage metric (LCM) [2], [3]. All modeled faults in channels in the round are detected resulting 100% fault coverage metric (FCM). However, the FCM is 37.71% at the network level i.e. against total modeled faults. Diagnosis of modeled faults i.e. fault simulation is carried out in ModelSim PE version 10.3c simulator integrated with the Xilinx simulator and is resulted as channel errors. We observe that our fault simulation detects 70.76%, 16.52%, and 12.72% as payload, misrouting, and dropping error respectively which again cumulatively ensure 100% FCM. Similar fault simulations are repeated in other test rounds by shifting the test application shown in Figures 5b to 5d to detect and diagnose shorts in rest of the channels. Thus, the LCM and FCM are achieved to 100%.

#### B. On-Line Performance Evaluation

When channels in the octagon network are non-faulty then one can observe expected performance and application data can reliably be transported over channels. However, the

performance can be visibly affected while one or multiple channels in a routing path suffer from short faults. Subsequently, various system level failures are raised in the network. These failures are manifested as channel errors. The proposed test model (*Charka*) evaluates following three performance metrics- throughput, latency, and power (energy) consumption. A cycle-accurate Noxim [12] simulator with similar simulation setup discussed in [2], [3], [8] is used for performance evaluation. The octagon network is built using a  $2 \times 4$  mesh network where diagonally opposite routers are interconnected too. Figure 7 demonstrates the on-line evaluation of the proposed test model in a test round. In other words, we observe the effect of channel shorts on the above mentioned performance metrics on a test application in a round. The traffic in terms of packet flits are determined in the network at different packet injection rate (PIR). Corresponding received and dropped flits are observed in the network. Note that the received flits sums the original and duplicated flits. A fraction of injected traffic is lost due to packet timeout fault. However, this fault is enhanced when CWs that carry packet trailers are affected by shorts. Generally 4–7% of injected traffic is lost due to timeout. But, with faulty channels, this dropping is enhanced to 10–16% [8], [9]. Consequently, the behavior of performance metrics is changed.

#### IV. COUNTERING PRIOR MODELS

The proposed *Charka*-based test application counters a number of existing test modules [2], [3], [4], [5], [6] at least in terms of test clocks. Subsequently, the network performance is improved especially to packet latency and energy consumption by a packet flit. When we execute the test model proposed in [5], we need  $T_{it}=24$  i.e.  $T_{n/w}=2760$  clocks to address the channel shorts. One may consider the  $2 \times 2$  test configuration discussed in [4] is equivalent to our test application in a subnet. Though, the channels can be tested in 4 rounds but the approach takes  $T_{n/w}=1008$  clocks to cover all modeled shorts. The model discussed in [6] has addressed shorts in interswitch channels only and is based on a sequentially router selection method. The model takes  $T_{it}=16$  and is completed in  $T_{n/w}=1840$  clocks. If the faults are to address in local channels, then the  $T_{n/w}$  equals to 2432 clocks. However, if the *Charka* is applied, it needs  $T_{it}=8$  and the  $T_{n/w}=320$  clocks. Therefore, the *Charka* seems to be 9X faster than the above mentioned models. If the model discussed in [2] is taken, the testing can be completed in  $T_{it}=2$  and  $T_{n/w}=80$  clocks but no application data are transmitted till the testing is over. The test mode behaves like an off-line mode where whole network remains busy in a test mode. For some critical systems [13], freezing the whole system in a test mode is not allowed. An application of the 2-hop model [3] offers comparative lower test time over the previous models. The model covers channels short faults by  $T_{it}=6$  i.e.  $T_{n/w}=240$  clocks. However, the test configuration (test setup in a subnet) size varies with an iteration that compel to irregular resource consumption incurring hardware and performance overheads. Thus, the *Charka* successfully replaces the existing models.

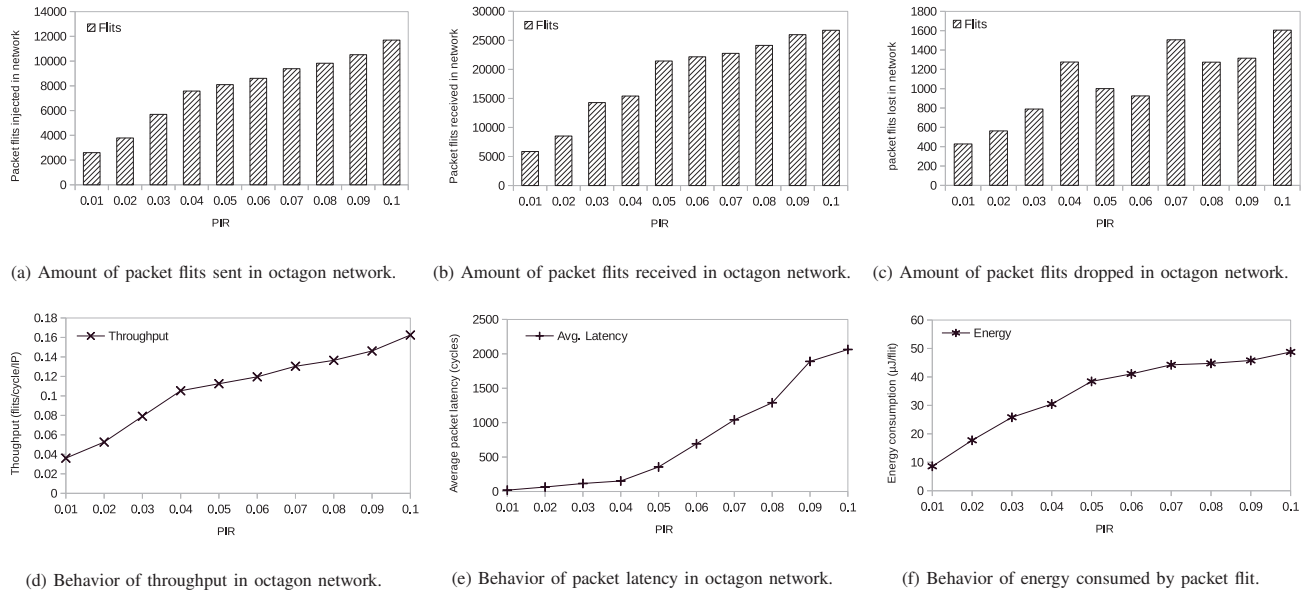


Fig. 7: The on-line evaluation of various performance metrics for the *Charka*-based test application on an octagon network with 16-bit channel width.

Further, lowered test time improves packet latency and related resource utilization. We observe that the proposed solution improves packet latency by 13.79-21.17%. Consequently, the energy consumed by a packet flit is reduced by 17.57-24.97% on average as compared to the existing models. Beside the above improvements, the *Charka* is better w.r.t area overhead of the TM. For instance, a TM in the models [4], [5], [6] take 25-45% area in a router while the component as proposed in the *Charka* needs only 12-19% area. Then, the proposed TM reduces 13-26% hardware area resulting 52-57.77% improvement.

## V. CONCLUSION

This paper has presented a low cost and fast on-line test algorithm that addresses short faults in channels of an octagon NoC and has deeply analyzed the effect of these faults on common network performance metrics. The fault simulation has resulted 100% coverage metrics. Also, the proposed model might be selected as an alternative to a set of approaches in analyzing channel shorts in the octagon NoC.

## REFERENCES

- [1] C. Grecu, A. Ivanov, R. Saleh, and P. Pande, "Testing network-on-chip communication fabrics," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 12, pp. 2201-2214, Dec 2007.
- [2] B. Bhowmik, J. K. Deka, and S. Biswas, "An odd-even model for diagnosis of shorts on noc interconnects," in *2015 Annual IEEE India Conference (INDICON)*, Dec 2015, pp. 1-6.
- [3] B. Bhowmik, S. Biswas, and J. K. Deka, "Impact of NoC interconnect shorts on performance metrics," in *Twenty Second National Conference on Communications 2016 (NCC 2016)*, Guwahati, India, Mar. 2016.
- [4] E. Cota, F. Kastensmidt, M. Cassel, P. Meirelles, A. Amory, and M. Lubaszewski, "Redefining and testing interconnect faults in mesh nocs," in *Test Conference, 2007. ITC 2007. IEEE International*, Oct 2007, pp. 1-10.
- [5] C. Concatto, J. a. Almeida, G. Fachini, M. Hervé, F. Kastensmidt, i. Cota, and M. Lubaszewski, "Improving the yield of noc-based systems through fault diagnosis and adaptive routing," *J. Parallel Distrib. Comput.*, vol. 71, no. 5, pp. 664-674, May 2011.
- [6] M. Kakoe, V. Bertacco, and L. Benini, "At-speed distributed functional testing to detect logic and delay faults in nocs," *Computers, IEEE Transactions on*, vol. 63, no. 3, pp. 703-717, March 2014.
- [7] F. Karim, A. Nguyen, and S. Dey, "An interconnect architecture for networking systems on chips," *IEEE Micro*, vol. 22, no. 5, pp. 36-45, Sep. 2002. [Online]. Available: <http://dx.doi.org/10.1109/MM.2002.1044298>
- [8] B. Bhowmik, J. K. Deka, and S. Biswas, "Towards a scalable test solution for the analysis of interconnect shorts in on-chip networks," in *MASCOTS 2016 - 2016 IEEE 24th International Conference on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, Sep 2016.
- [9] B. Biswajit, J. K. Deka, and S. Biswas, "An on-line test solution for addressing interconnect shorts in on-chip networks," in *2016 IEEE 22nd International Symposium on On-Line Testing and Robust System Design (IOLTS)*, July 2016, pp. 9-12.
- [10] C. Zeferino, M. Kreutz, and A. Susin, "Rasoc: a router soft-core for networks-on-chip," in *Design, Automation and Test in Europe Conference and Exhibition, 2004. Proceedings*, vol. 3, Feb 2004, pp. 198-203 Vol.3.
- [11] D. Bertozzi and L. Benini, "Xpipes: a network-on-chip architecture for gigascale systems-on-chip," *Circuits and Systems Magazine, IEEE*, vol. 4, no. 2, pp. 18-31, 2004.
- [12] V. Catania, A. Mineo, S. Monteleone, M. Palesi, and D. Patti, "Noxim: An open, extensible and cycle-accurate network on chip simulator," in *Application-specific Systems, Architectures and Processors (ASAP), 2015 IEEE 26th International Conference on*. IEEE, 2015, pp. 162-163.
- [13] E. A. Rambo, A. Tschien, J. Diemer, L. Ahrendts, and R. Ernst, "Fmea-based analysis of a network-on-chip for mixed-critical systems," in *2014 Eighth IEEE/ACM International Symposium on Networks-on-Chip (NoCS)*, Sept 2014, pp. 33-40.