Design of Bio-Network on Chip (Bio-NoC) to detect and destroy Trojan and Fault

Dr.K.Balamurugan, Department of Electronics and Communication Engineering, Swarnandhra College of Engineering and Technology, Andhra Pradesh, India.
R.Shijukumar, Senior R&D Engineer, Firmtech, Thiruvanathanpuram, Kerala, India.
Dr. M.Vijayaraj, Department of Electronics and Communication Engineering, Government College of Engineering Srirangam, Tamilnadu, India.

Article Info
Volume 83
Page Number: 3129 - 3137
Publication Issue:
March - April 2020

Article History

***Article Received**: 24 July 2019*
***Revised**: 12 September 2019*
***Accepted**: 15 February 2020*
***Publication**: 21 March 2020*

Abstract:
The communication inside an Integrated Chip (IC) is nowadays done by the technology called Network on Chip (NoC).NoC uses packet transfer methodology instead of signal. NoCs improves the scalability and power efficiency of multicore ICs. Similarly, the chances of failures or faults in on-chip components are also high. Apart from the physical and logical fault occurs in NoC, nowadays Malicious Trojan's (MalT) threats the NoC security. Malicious Trojan's are the hardware virus which causes more trouble in multicore ICs. Bio-inspired NoC is another method to detect and destroy the Trojans and it also provides fault tolerance in multicore ICs. In this work, we have determined the latency of NoC architecture without Trojans or fault. We have applied synthetic Trojans and faults to the NoC architecture and detected it during the simulation. The latency for this case is also noted. This novel Bio-NoC structure eliminates the applied Trojans, as well as faults and the latency for this process, is determined. It is showed that this novel Bio-NoC provides better results compared to the affected NoC. The outputs are compared with already existing technologies such as synaptogenesis and sprouting based NoC technologies.

***Keywords**: NoC Security, Malicious Trojans, Network-on-Chip (NoC), Hardware Security, System-on-Chip (SoC).*

I. INTRODUCTION

The modern world today is moving towards a fast process in Electronics. Integrated Chip (IC) is an important component in most of the Electronic devices. Multicore IC is nowadays popular for its speed and reliability. Multicore IC contains many cores inside a Chip. The communication between the cores is very much important for its speed. The typical bus and crossbar relation of SoC compon ents has been replaced by NoC.NoC is an on-chip communication protocol. NoC gives the measured quality, versatility and proficient reuse of the assets with high data transmission.

Many fault tolerance algorithms provide only fault, failure detection but not recovering the NoC architectures form the faults or failures [3]. Nowadays everybody is working towards miniaturization of electronic devices for many types of applications. Another part of the IC can be minimized drastically but it is difficult to reduce the size of the interconnects in the NoC architecture. If anybody is trying to reduce the interconnects, it leads to permanent and transient or provisional faults. Permanent faults are the faults which cannot be corrected perfectly.

Very efficient Fault-tolerant algorithms are using the concept of the human brain. The brain is an important part of the living being, which can take the necessary steps for every activity[2]. This novel biologically stimulated fault-tolerant algorithms can detect, avoid the faults or damage caused by any malicious program also.

Tolerance of faults occur inside the chip is a way of adjusting both permanent and provisional faults [12]. These faults are because of changes in the quantity of voltage or temperature as well as router congestion etc. These faults can be determined by the aid of fault-tolerant routing algorithms.

Today, the IC fabrication process, assembly and testing are mostly outsourced to the third parties brought the chip confronting new challenge on hardware security issues. There are many threats or virus are available to destroy the NoC system[4]. Malicious Trojans (MalT) are a very dangerous threat to an NoC system. It collapses all its process and reduces the speed of the system drastically. This MalT is designed using activate circuit and payload circuit.

The activate circuit is utilized to inspect the appearance of the activation condition that the attacker specifyed in the MalT inclusion stage. The MalT payload circuit may cause a problem called as denial-of-service (DoS), which adjust a chip's typical activities or give the benefit to the anonymous person with the illegal license to access private memory . It is very challenging to detect the MalT using the algorithms using Register Transfer Level (RTL) of the design.

The detection of this dangerous MalT can be classified as non-ruinous and damage identification strategies. The non-ruinous strategy incorporate testing, quickened Trojan actuation pursued with testing, and examination on side-channel signals[7]. This non-ruinous also determines the power, delay, temperature. The damaging technique classification is predominantly ascribed to, picture remaking and investigation, and fingerprint examination. As MalT plans are getting progressively unpredictable and the MalT target framework coordinates more transistors, MalT identification in an exceptionally huge scale incorporated framework is equivalent to finding a needle in a bundle.

When this MalT is embedded in NoCs, it will prompt data spillage, unauthorized memory access get right of entry to, and DoS attacks, which includes wrong path dirrection, deadlock, and livelock. Previous techniques and countermeasures have been intended for the general Chips.

Existing methodologies to detect the MALT and countermeasures have been designed for the entire Chip. For multicore chips the firmware countermeasures are to be had to come across the occurrence of inactive NoCs.

II. BACKGROUND

Amir Hosseini et.al [1] centred to keep up the performance of NoC in the presence of faults due to virus or Trojans in their work. They proposed a completely versatile steering calculation which utilizes one and two VCs along with the X and Y measurements. They presented the High-Performance Fault-tolerant Routing calculation to endure the shortcomings without influencing the presentation of NoC. The parcels were moved along the west, east, north and south headings to keep away from deadlock[10].

Chaochao Feng et.al [5] introduced another dynamic routing algorithm for NoC application which deals with both the dynamic and static undeviating failures. This routing circulates the heap over the entire system by considering the stress factors. The dynamic routing algorithms in the router is working to push the data packets to the destination.The dynamic routing method of every router checks the destination location of the packet and then it chooses the required route to reach the destination and in a similar time to maintain a strategic distance from defective or clogged connections.

S. Narasimhan et. al [9] proposed an undercover threat model for model for ICs designed by the help of third parties. The proposed work likewise outlines that the virus affected NoC can disturb the accessibility of components inside the chip, consequently rooting enormous performance bottleneck on NoC platform. The creator proposed a realtime latency reviewer that empowers an IC integrator to screen the dependability of deployed NoC all through the chip lifetime. Realtime Latency assessor for NoCs is the runtime technique

used which is non-invasive and does not yield any support from IP provider of NoC. RLAN infuses painstakingly chosen packets, these packets detects bizarre deferrals in its transit.

R. S. Chakra borty, and S. Bhunia[5] in their work aimed for expanding the accessibility of an NoC by methods for the execution of hardware-based mechanisms that filter malicious packets infused into the network system by an assaulting centre core and dispose of packets that influence the system accessibility, and direct the infusion rate. The core and network system interfaces use dynamic routing tables. The improvement of security components is done in a module put among the core and router. They designed a module which works as a controller that filters the pernicious packets.

G. K. Contreras et.al [8] introduced a protected NoC system made out of a lot of Data Guard Units (DGU) executed inside Network Interfaces card (NIC) of the NoC. The setup of the executable part of the DGUs is overseen by a focal point and the NSM (Network Security Manager). An adaptable execution of the protected system design that is appropriate for frameworks described by different functions and a reconfigurable situation is proposed. The DGU ensures protection for recollections and memory-mapped components. The DGU projected in this work signifies empowering rights to the system memory just if the originator of the solicitation is approved. Accessing the memory is performed by considering the address of the system memory as well as the kind of activity mentioned and the status of the initiator. The utilization of the DPU offers the plausibility of effectively stacking/putting away basic information and guidelines while shielding them from illicit gets to by vindictive program running on the chip, without requiring tedious encryption/decoding.

III. INSERTION OF MALT IN NOCs

MALT s are intended to convey attacks, for example, DoS, data spillage, information control, and framework corruption. MALTs can be installed at the RTL representation, or directly into the door point net rundown, prompting coherent attacks on the framework.

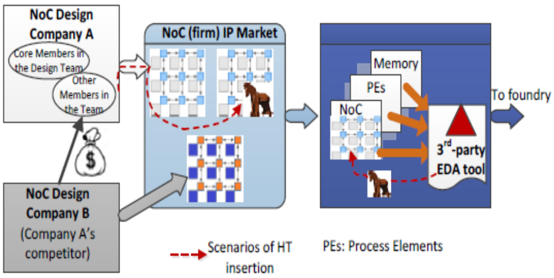
NoC Design Company A
Core Members in the Design Team
Other Members in the Team
$
NoC Design Company B (Company A's competitor)
NoC (firm) IP Market
Memory
PEs
NoC
3rd-party EDA tool
To foundry
Scenarios of HT insertion
PEs: Process Elements

Fig.1.Hardware Trojan insertion scenario[4]

At the manufacturing side[7], structure formats can be adjusted to incorporate a MalT by changing inner circuit qualities. This work address the MALTs, that is introduced by the unfaithful associate in the NoC arrangement, as showed up in Fig.1

A few partners obtained by the contender of the NoC design association (Example A) can put the MALT in NoC structure. Notwithstanding the way that not being a middle part in the structure gathering, the traitorous laborer may have the passage to the netlist of NoC and unkindly changes a couple of switch dispatches by the association A. In spite of whether the principal NoC IP doesn't contain MALTs, the untrusted outcast EDA mechanical assembly may implant MALT s into the IP of NoC to design an unbound ICs or SoCs[7]. The resulting circumstance causes disastrous results for the undermined ICs used in military and government applications. Despite rely upon the rule game plans, an NoC fashioner ought to realize countermeasures to restrict potential MALT increases from the generation network.

In NoCs, the bounce is the essential stream power entity that is moved on the NoC. Every NoC packet has 1 header bob, 1 extremity vacillates and a couple of payload moves. The header value in bit is the fundamental piece, where elevated basis regard shows the proximity of header skip. The consequent piece is the extremity bit, where elevated rationale esteem indicates the appearance of extremity Flit.

The other bits of the move bits consists of information, for instance, a starting place identifier, objective location and coordinating show, or data bits[10]. Bob source/objective address and move type information guide NoC change to moves over the diverse hop coordinating ways. Pernicious changes on the header skip or move type may annihilate the vacillate's information respectability, realizing a wrong routed pack or a move misfortune. The wrong routed group possibly harms the gridlock liberated coordinating guidelines, making ends in the framework. Stop, livelock, and bounce misfortune will incite data transfer capacity consumption[13].

With regards to NoC-based ICs or multicore SoCs, MALT attacks that concentrate mystery data in the NoC switches. As the fundamental point of convergence in this concept is to set the NoC structure, we address the MALT s that perform DoS ambushes in the NoC switches.

The results of MalT DoS assaults incorporates data transfer capacity consumption, mistaken way steering, stop and livelock. Even more unequivocally, MALTs in switches can toxically modify the move source/objective address or information category of a packet from the transmitter NIC. In case a MalT payload alters the objective location of a package, that group could be facilitated to an unapproved IP centre.

IV. CONTRIBUTIONS

(1) We put-forwarded countermeasures to set the NoC structure, instead of complete contingent upon programming or firmware answers for recognizing an undermined NoC used in the ICs or SoCs. Our NoC solidifying technique keeps an eye on the MALTs put in the gate level netlist of NoC by the double-crossing specialist in the structure domicile or untrusted outsider framework Industry.

(2) Our system is expected for the NoC switches, instead of the NICs centred in the present works, to quickly direct the MALT impacts before the data transfer capacity exhaustion happens to the complete NoC. In addition, our switch level MALT help instrument increases present expectations for an enemy to all the while control different directing bounces to make a pernicious correspondence way between two IP centres in the NoC-based multicore SoC. In this manner, a pack affected by MALTs can't successfully transverse different switch hops and land at the goal to complete the malignant endeavour.

(3) The proposed synergistic powerful change and move respectability check strategy is fit for taking the factors of NoC quickly end the recognized MALTs.The proposed periodic vector age method abuses stand-out system assortments and the subjective substance from each change in the registers to assure each switch uninhibitedly contradict potential MALT attacks.

(4) Many explicit MALT attacks in NoCs are considered in this proposed system than the current scholarly works. To the extent we might know, this is the principal work that diminish the MALTs that malignantly change the bounce content, for instance, essential move work bits and controlling way bits. Appropriately, our recreation results show the critical move bits being chosen the NoC throughput, interface accessibility, traffic hotspot development, information transmission utilization, region cost and ordinary packet inactivity.

V. BIO- NoC ARCHITECTURE

The bio- NoC algorithms are implemented using different system of NoC architectures and it is assumed to be a Two Dimensional 4*4 (16 node) mesh NoC. There is a space table in each scheduler to dispense a period cut to a specific association. Presently, the ideal four openings are saved depends on the writing. The four spaces allude to the virtual channels (4 number of VCs) which can at the same time happen per port. Fig. 2 and 3 shows the architecture of scheduler and inPort respectively.

The scheduler shown in Fig.2 has four slot and each slot is allocated to a specific VC. This method holds the timeslot and data transmission for a specific connection. Subsequently, 4 channels (namely A, B, C, D) are conceivable utilizing 4 spaces. Time-Division Multiplexing (TDM) clock peruses these 4 slots. The clock time frame is set as 2ns. The capacity of the clock segment is to peruse the scheduler openings after each 2ns.

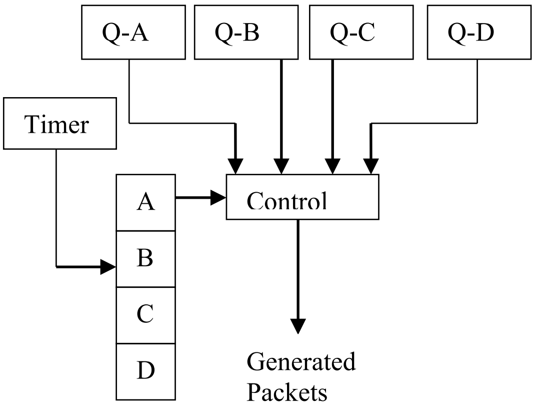

Fig.2.Scheduler architecture

The slots aid to productively use and gap the data transmission of the NoC among 4 associations. So as to use all the clock cycles, switches increase the opening number by one. The slots are increased utilizing a round-robin service. This procedure serves to productively use the data transmission and one flit is steered from four switches in the continuous four clock cycles. The scheduler additionally avoids those openings which are not allotted to any of the virtual channels. This encourages the association with productively use the data transfer capacity of interconnect and keep up the elevated throughput for that specific association. The proficient use and perusing of the slots likewise increment the presentation of the specific port as it will devote the accessible transfer speed to as of now dispensed dynamic connections.

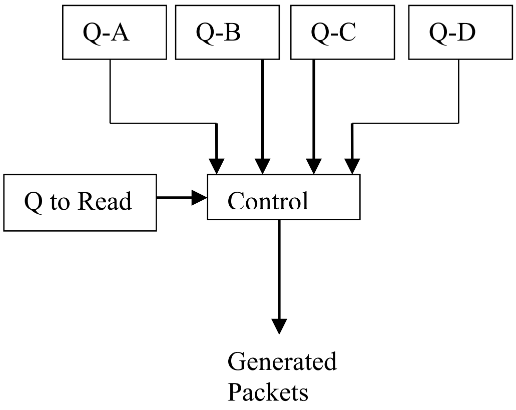

Fig.3. InPort architecture

Thus, at the inPort, 4 Qs are assigned to each VC. The specific queues are perused by the VC esteem availablet in the Q to Read variable as shown in Fig.3. The Queue to Reading esteem is given by the scheduler, that is, the particular Q is assigned to a specific VC. In HOL, when the header of the bundle is hindered because of blockage or defective link then the entire packet is obstructed. The portion of openings to each VC additionally maintains a strategic distance from the dispute between numerous connections. While perusing the slots, the scheduler avoids those openings which are not dispensed to any VC.

It proficiently uses the transmission capacity of the link and furthermore keeps up the throughput for that association.

V. BIO-INSPIRED NOC FRAMEWORK

In our work, the Bio-NoC is executed on a GUI based MATLAB test system. The scheduler of the port continually screens the links utilizing the bio-NoC algorithm. At whatever point a link gets defective or affected by the MALT, the specific ports related with that interconnects are additionally blocked. At the point when a flit is gotten by the scheduler to send it on the (Out) connection of the stopped port, the neighbour hub scheduler recognizes a fault. The scheduler then establishes a fresh link from the present node to the destination node.

During the creation of sprout (temporary link), the need is consistently to interface with the more established principle link to avoid needless router traversal. This makes the work exceptionally powerful and diminishes the latency between source and destination.

During the revival of a fault the throughput is somewhat corrupted for shorter timeframes and it attempts to recover the first throughput after avoiding the defective interconnect[11]. The flits are as yet crossing on more seasoned primary connections, as those flits have just circumvented the defective link. This permits the bio-enlivened algorithm to productively use the data transfer capacity of NoC.

At first, the link is started from the source towards the destination. The destination consequently propels the respond signal. In the wake of accepting the respond signal at source, the flits are sent from source to destination.

In the event that no issue is distinguished during the correspondence, at that point are come to at the destination, otherwise, the neighbour router recognizes the deficiency and starts the provisional link between the present hub and the previous link or to the destination hub. In the event that there is no more established neural connection (synapse) [8] to associate with, the temporary connection is legitimately associated with the destination .

The routers in NoC demand and respond the control signals. In the wake of distributing the control signals, routers think about the standing of their neighbour routers. The bio-NoC algorithm just needs network detection toward the beginning of the simulation. This effectively uses the data transmission of the NoC. After the identification of network, the source creates the link signal "synap" inorder to connect with the destination.

This proposed work uses the advantages of existing synaptogenesis and sprouting algorithm (SSA) to find the static faults during the connection establishment and to the runtime faults respectively. This work also mitigates the MalT or viruses using the Bio-NoC algorithm.

VI. RESULT AND DISCUSSION

The simulation results are assessed in MATLAB using the GUI tool. The latency, throughput, bandwidth are acquired using MATLAB.

a) Simulation result for Bio- NoC fault tolerance

The path in Fig 4 indicates the multiple faults during flit transfer from source to destination. In this case, bio-NoC fault-tolerant techniques are utilized to identify the fault and bypass the packets in the older connection.

b) Simulation results for MALT mitigation methods

The simulation result shows the bandwidth utilisation of bio- NoC is better than the existing TDM approaches. The bandwidth utilisation is increased to 125.73% than the average TDM connections. The usage of bandwidth (average) per fault of proposed Bio-NoC algorithm shows 40 Mbps.This is the best bandwidth compared to the existing Sprouting and synaptogenesis algorithm (SSA) as illustrated in Fig.5.

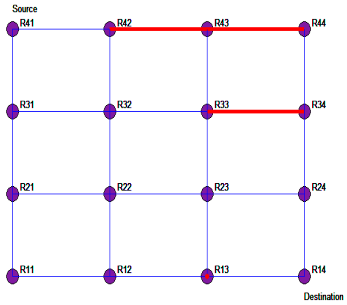

Fig. 4 Multiple path detection in Bio- NoC

The latency of Bio-enlivened techniques increases as the number of fault in the NoC increases.Fig.6 shows the average latency of the proposed algorithm is 4ns, compared to SSA.

Table 1No.of Packets received for different MalTS

		3 D MalT			3 H MalT			3 E MalT		
		I	II	III	I	II	III	I	II	III
No.of packets Received	0.07	3851	2010	989	3001	2080	1020	3800	200	54
	0.08	3980	2100	990	3180	2200	1100	4100	2900	120
	0.09	4101	2220	995	3352	2415	1150	4325	2870	150
	0.1	5011	2350	998	3800	2600	1220	4500	2800	230
	0.12	5200	2690	1050	4100	2980	1109	5000	1200	240
	0.15	6900	3250	1209	5350	3150	1330	6800	870	420
	2	9820	4850	1424	7690	5019	1850	9590	550	530

Table 1 denotes the number of packets received for different kinds of MalTs injected such as 3DMalTS, 3HMalTs, 3EMalTS.

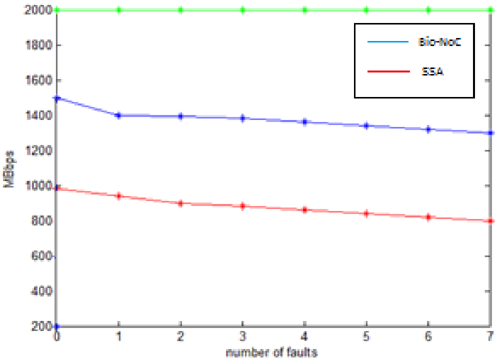

Fig.5 Bandwidth vs. number of faults

It also illustrates the number of packets received per cycle per node. The packets received is almost proportional to the strength of the MalTS except for the II- 3EMalTs.The value of II 3EMalTs increases, constant and then decreases.

The total number of applicable packets received by the NoC NIC in the known simulation time is defined as throughput. In this work, the different MalTS are injected to check the throughput of the system. Usually, three kinds of MalTs can be applied. First is Destination MalT (D MalT), second is Header MalT and the third is EXTREMITY MALTs (E MalT). The strength of each MalT is designated from I to IV. The MalT 'I' represents less strength and the MalT 'IV' represents high strength.

The proposed mitigation method receives the same amount of packet for the same traffic injection rates for any number of MALTs injected illustrated in Fig.7 compared to SSA.In the proposed strategy, the average packet latency does not increase with an increase in the number of D MALTs. D MalTs are the malicious virus injected or exist at the destination node. At the point when the traffic infusion rate increases, the average latency is within only 2 %.

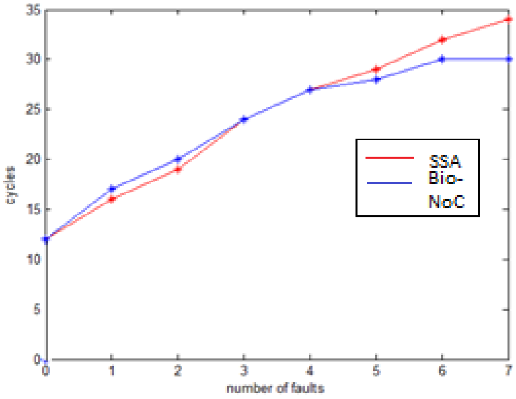

Fig.6 Latency vs. number of faults

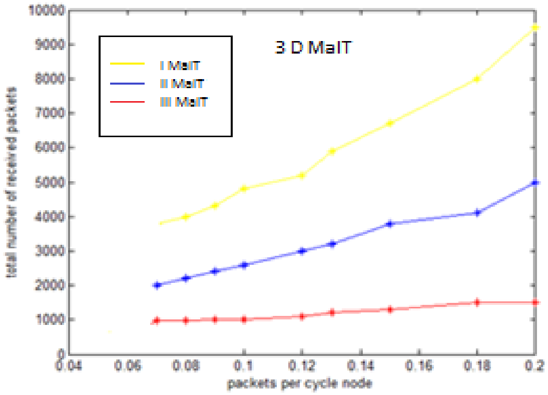

Fig.7 Number of valid packets received for 3D MalTs

Probability of attack success (P_{as}) is defined as the ratio of the number of times that the Malicious Trojan (MalT) effectively chooses the bits having a place with the flit field of interest over the total number of test cases. P_{as} value is less than 1 for this proposed mitigation method. This attack is carried out for 32 bits, 64 bits and 128 bits.

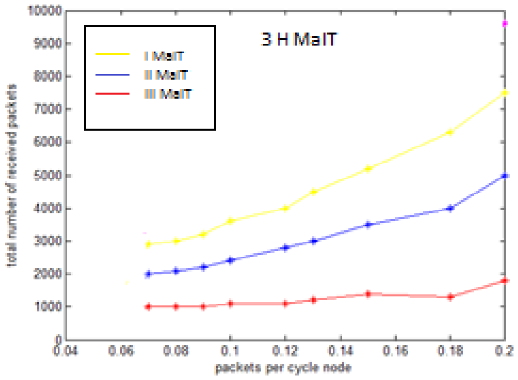

Fig.8 Number of valid packets received 3H MALTs

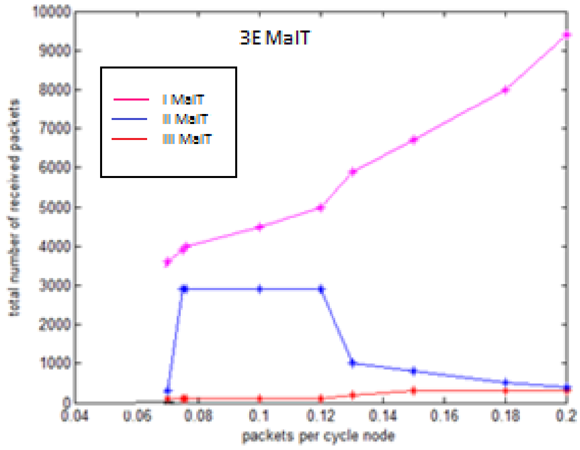

Fig.9 Number of valid packets received for 3E MALTs

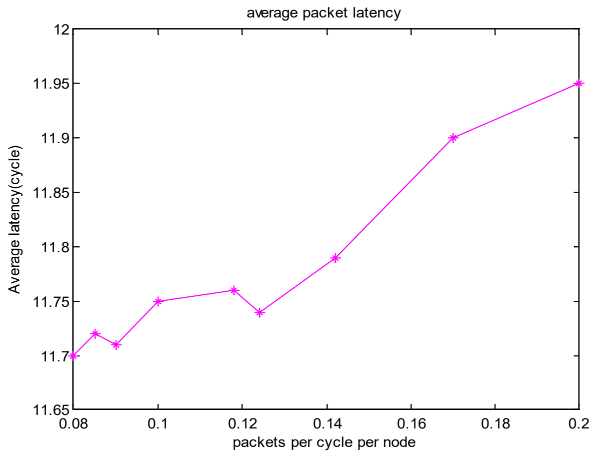

Fig.10 Average packet latency for different number of MALTs

The number of valid packet received during the different strength of the Header MalTs is injected are shown in Fig.8.H MalTs are the viruses injected in the header of the data packets.The total number of received packets vary according to the strength.As the proposed Bio-NoC algorithm is used,the number of packet received is higher than the existing SSA methods.

Fig.9 shows the number of valid packets received for 3 EXTREMITY MalTs(E MalT).E MalT is the virus available in the end flit of the packet. The harmfulness of the MalT depends on the strength of the Malt.

Fig.10 shows the average packet latency for different number and different types of MALTs, which harms the NoC system. The Bio-NoC system improves the average latency of the NoC compared to the existing method called as SSA.

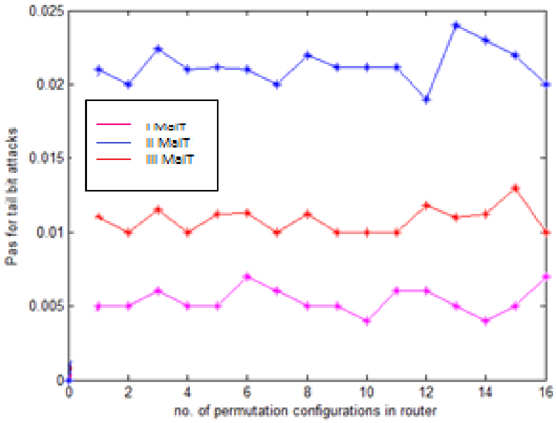

Fig.11. The success rate of extremity bit attacks

The probability of success rate (P_{as}) considerably diminishes with the increasing number of permutation configuration increases. If the number of permutation is equal to 16, this proposed method achieves a probability of attack success of $1.04*10^{-2}$.

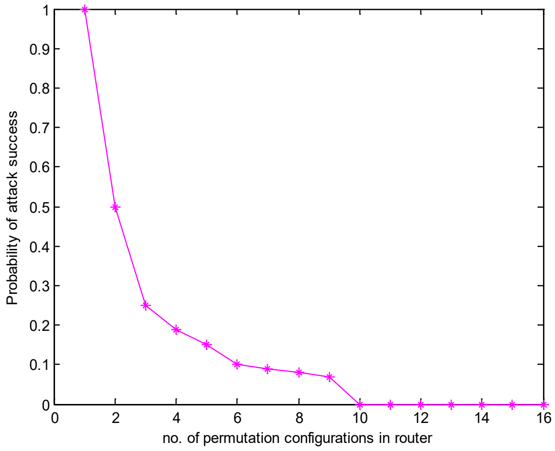

Fig.12. Probability of attack success rate (P_{as})

The Success rate of extremity bit attacks for the various number of permutations configurations is shown in Fig.11 and Fig.12 exemplify the probability of attack success rate (P_{as}). The P_{as} is high for zero permutation configuration in the router and diminishes drastically when the number of permutation configuration in the router increases.

VII. CONCLUSION

The proposed bio- NoC provides a mesh network which is free from any type of faults or malicious Trojans. The proposed system accomplished adaptation to non-critical failure by synaptogenesis and growth strategies of the brain. Bio-NoC has an adaptation to non-critical failure and powerful characteristics. The Malicious Trojans are limited by improving the NoC design. The mitigation of MALT is achieved by a collaborative dynamic permutation and flit integrity check strategy. The dynamicity of flit permutation is achieved by random selection vector generation method.

REFERENCES

1. Amir Hosseini, Tamer Ragheb, Yehia Massoud, ‘A fault-aware dynamic routing algorithm for on-chip networks’, in: Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS2008,2008, pp.2653–2656.
2. Saxena, V., Wu, X., Srivastava, I., & Zhu, K., ‘Towards Neuromorphic Learning Machines Using Emerging Memory Devices with Brain-Like Energy Efficiency’, Journal of Low Power Electronics and Applications, vol.8, 2018, pp.34-44.
3. Vijayaraj M, Balamurugan K, ‘Performance oriented docket-NoC (Dt-NoC) scheme for fast communication in NoC’, J Semicond Technol Sci., vol.16, 2016,pp.359–66.
4. M. Banga and M. Hsiao, ‘VITAMIN: voltage inversion technique to ascertain malicious insertions in ICs’, in Proc. HOST’09,2009, pp. 104–107.
5. R. S. Chakra borty, and S. Bhunia, ‘HARPOON: An Obfuscation-Based SoC Design Methodology for Hardware Protection’, IEEE TCAD, vol. 28, no. 10, 2009,pp. 1493–1502.
6. Chaochao Feng, Zhonghai Lu, Axel Jantsch, Minxuan Zhang, Zuocheng Xing, ‘Addressing transient and permanent faults in NoC with efficient fault- tolerant deflection router’, Very Large Scale Integer(VLSI) Systems, IEEETrans. 21,2018, pp.1053–1066.
7. G. K. Contreras, M. T. Rahman and M. Tehranipoor, ‘Secure Split-Test for Preventing IC Piracy by Untrusted Foundry and Assembly’, in Proc. DFT’13, 2013, pp. 196–203, 2013.
8. Falko Dressler, Ozgur B.Akan, ‘Bio-inspired networking: from theory to practice’, Commun. Mag. IEEE, 48, 2010, pp. 176–183.
9. S. Narasimhan, et al., ‘Improving IC Security Against Trojan Attacks Through Integration of Security Monitors’, IEEE Design & Test of Computers, vol.29, no. 5,2012, pp.37–46.
10. A. Prodromou, A. Panteli, C. Nicopoulos, and Y. Sazeides, ‘Nocalert: An on-line and real-time fault detection mechanism for network-on-chip Architectures’,in Proc. MICRO’12,2012, pp. 60-71.
11. R. J. Shridevi, D. M. Ancajas, K. Chakraborty, and S. Roy, Runtime ‘Detection of a Bandwidth Denial Attack from a Rogue Network-on-Chip’,in Proc.NOCS’15,8,2015,pp. 1-8.
12. Teijo Lehtonen, David Wolpert, Pasi Lilje berg, Juha Plosila, Paul Ampadu, ‘Self-adaptive system for addressing permanent errors in on-chip interconnects, Very Large Scale Integr.(VLSI) Syst.IEEETrans.,18, 2010,pp.527–540.
13. Q. Yu and J. Frey, ‘Exploiting Error Control Approaches for Hardware Trojans on Network-on-Chip Link’, in Proc. 16th IEEE Symp. Defect and FaultTolerance in VLSI and Nanotechnology Systems,2013, pp. 266-271.