International Journal of Engineering Sciences & Emerging Technologies, Oct. 2023.
ISSN: 22316604 Volume 11, Issue 2, pp: 337-349 ©IJESET

A COMPREHENSIVE STUDY OF NETWORK ON CHIP AND SYSTEM ON CHIP IN PERSPECTIVE OF INTERNET OF THINGS

Ashutosh Dhar Dwivedi, Shreya Chandola

Assistant Professor, Institute of Hospitality,
Management & Sciences Kotdwar Uttrakhand India
ashutosh.dwivedi@ihms.ac.in
shreya165014@gmail.com

ABSTRACT

Network on Chip (NoC) and System on Chip (SoC) are both important technologies used in various fields, including embedded systems, high-performance computing, and Internet of Things (IoT). NoC is used to provide a communication infrastructure for systems that integrate multiple components, such as processing units, memory, and peripherals, onto a single chip or across multiple chips. NoC can improve the scalability, performance, and energy efficiency of such systems by providing a high-bandwidth, low-latency communication channel between the components. SoC is used to integrate multiple components, such as processing units, memory, and peripherals, onto a single chip. SoC can be used in a wide range of applications, including smartphones, IoT devices, automotive systems, and aerospace systems. SoC can improve the performance, power efficiency, and cost of such systems by providing a compact and integrated solution. Some examples of the use of NoC and SoC are-High-performance computing: NoC can be used to provide a communication infrastructure for large-scale parallel computing systems, such as clusters and supercomputers. SoC can be used to integrate multiple processing units and memory modules onto a single chip, providing a high-performance computing platform. IoT devices: NoC can be used to provide a communication infrastructure for IoT devices, allowing them to communicate with each other and with the cloud. SoC can be used to integrate multiple components, such as sensors, processing units, and wireless communication interfaces, onto a single chip, providing a compact and energy-efficient solution for IoT devices. Automotive systems: NoC can be used to provide a communication infrastructure for automotive systems, allowing different components, such as engine control units, infotainment systems, and safety systems, to communicate with each other. SoC can be used to integrate multiple components, such as processing units, memory, and communication interfaces, onto a single chip, providing a compact and integrated solution for automotive systems. Overall, NoC and SoC are both important technologies that can be used to improve the performance, power efficiency, and cost of various systems in different fields. The choice of which technology to use depends on the specific requirements of the system and the application.

KEYWORDS: *Network on Chip, System on Chip, IoT, Hardware Trojan, Security.*

1. INTRODUCTION

A Network on Chip (NoC) is a communication architecture that enables efficient data transfer between multiple processing elements or intellectual property (IP) cores in a System on Chip (SoC) [2,48]. Traditionally, SoCs used a bus-based architecture for communication between different cores. However, as the number of cores in SoCs has increased, the bus-based approach has become a bottleneck due to limited bandwidth and scalability. NoC replaces the bus-based approach with a packet-switched network architecture that can handle high bandwidth data transfer and scale to accommodate a large number of cores. In a NoC, processing elements communicate with each other by sending packets of data through a network of switches and interconnects. The switches route the

packets to their intended destinations based on the network topology and routing algorithms. The network topology can be customized based on the application requirements, and the routing algorithms can be optimized to reduce latency and power consumption. NoCs provide several benefits over bus-based architectures, such as higher bandwidth, lower latency, improved scalability, and better power efficiency. They also offer greater flexibility in designing SoCs and can be easily adapted to new applications. NoCs are commonly used in applications such as video processing, machine learning, and high-performance computing [3]. They are also used in mobile devices, gaming consoles, and automotive systems, where multiple processing elements need to communicate efficiently and reliably. Silicon on Chip (SoC) refers to the integration of all the components of a system, including the processor, memory, peripherals, and interfaces, onto a single silicon chip. SoC is a complete computer system on a chip and can be found in various devices such as smartphones, tablets, wearable devices, and Internet of Things (IoT) devices. SoC is the result of advancements in semiconductor technology that have enabled the integration of millions of transistors onto a single chip. This integration has led to smaller, more power-efficient devices with higher performance and increased functionality. The components of a SoC are designed to work together seamlessly, which reduces power consumption and increases performance [47]. SoCs are highly optimized for specific applications and can be customized to meet the requirements of a particular device or product. SoCs are also commonly used in embedded systems, where a small, efficient, and reliable system is required. They are also used in automotive systems, medical devices, and industrial equipment. SoCs have revolutionized the design and manufacture of electronic systems, and their use is expected to grow as more devices become connected to the internet and require increased processing power and functionality. The use of Network on Chip (NoC) in Internet of Things (IoT) devices can provide significant benefits in terms of communication efficiency, scalability, and power consumption. I have studies so many research papers [1-56] to find out the comparison, analysis and all things for this paper.

1.1 Use of NoC & SoC in IoT-

IoT devices often require the integration of multiple sensors, actuators, and processors, which need to communicate with each other and with the cloud. NoC can facilitate the communication between these components and provide a reliable and efficient communication infrastructure. One of the key advantages of NoC in IoT is its scalability. NoC can be designed to support a large number of devices and enable seamless communication between them. This is particularly important in IoT, where the number of devices is expected to grow significantly in the coming years.Another advantage of NoC in IoT is its power efficiency. NoC can optimize the communication between devices and reduce the energy consumption required for communication. This is crucial in IoT, where devices often have limited battery life and need to operate for long periods without recharging. NoC can also improve the security of IoT devices. With NoC, communication can be encrypted and authenticated to ensure that only authorized devices can communicate with each other. This can prevent unauthorized access and ensure that sensitive data is protected. Overall, the use of NoC in IoT devices can provide significant benefits in terms of communication efficiency, scalability, power consumption, and security. As IoT continues to grow and become more prevalent, the use of NoC is likely to become increasingly important for designing and implementing efficient and reliable IoT systems.

System on Chip (SoC) is a common architecture used in Internet of Things (IoT) devices due to its compactness, low power consumption, and ability to integrate multiple components onto a single chip. IoT devices often require the integration of multiple components, such as sensors, processors, memory, wireless communication modules, and power management units. SoC enables the integration of these components onto a single chip, which reduces the size, cost, and power consumption of the

device. SoC architecture can also optimize the performance of IoT devices by integrating hardware components that are specifically designed for a particular application. This enables efficient data processing, storage, and communication, which can improve the functionality and reliability of the device. Another advantage of SoC in IoT devices is its flexibility. SoC can be customized to meet the specific requirements of an application, such as power consumption, performance, and connectivity. This enables the creation of IoT devices with specific functionality and features, which can be tailored to the needs of different applications and use cases. SoC architecture can also enhance the security of IoT devices. By integrating security components such as hardware-based encryption and authentication, SoC can provide a secure platform for IoT devices, which can prevent unauthorized access and protect sensitive data. Overall, SoC architecture is an effective solution for IoT devices due to its compactness, low power consumption, performance optimization, flexibility, and security. With the growth of IoT, the use of SoC is likely to become more prevalent as it enables the creation of efficient, reliable, and secure IoT systems.

1.2 Difference between SoC and NoC-

System on Chip (SoC) and Network on Chip (NoC) are both architectural solutions that enable the integration of multiple components onto a single chip [1,48]. However, they have some key differences:

1. Integration of components: SoC integrates all components of a system, such as the processor, memory, and peripherals onto a single chip, while NoC integrates the communication infrastructure of the system onto a single chip.
2. Communication: SoC uses a bus-based architecture for communication between different components, while NoC uses a packet-switched network architecture for communication between processing elements or intellectual property (IP) cores.
3. Scalability: SoC has limitations in scalability due to the bus-based architecture, while NoC can scale easily to accommodate a large number of processing elements.
4. Customization: SoC is highly optimized for specific applications and can be customized to meet the requirements of a particular device or product, while NoC can be customized based on the application requirements for communication efficiency, latency, and power consumption.
5. Power consumption: SoC architecture can be optimized to reduce power consumption, while NoC can optimize communication for low power consumption.

In summary, SoC integrates all components of a system onto a single chip and uses a bus-based architecture for communication, while NoC integrates the communication infrastructure of the system onto a single chip and uses a packet-switched network architecture for communication. SoC is highly optimized for specific applications and can be customized to meet the requirements of a particular device or product, while NoC can be customized based on the application requirements for communication efficiency, latency, and power consumption.

2. ARCHITECTURE OF NOC

The architecture of a Network on Chip (NoC) typically consists of a set of processing elements (PEs), interconnect fabric, and network interface units (NIUs)[2].

1. Processing elements (PEs): These are the individual IP cores or processing units that are connected to the NoC. PEs can include processors, memories, digital signal processors (DSPs), and other specialized hardware modules.

2. Interconnect fabric: This is the network infrastructure that connects the PEs and enables communication between them. The interconnect fabric consists of a set of network switches, routers, and links that are organized in a mesh, torus, or tree topology.
3. Network interface units (NIUs): These are the units that connect the PEs to the interconnect fabric. The NIUs handle the communication between the PEs and the interconnect fabric, and they can also perform protocol conversion and error checking.

The interconnect fabric can be designed using different routing algorithms, such as wormhole, virtual channel, and source routing. These routing algorithms can be optimized for different application requirements, such as low latency, high throughput, and low power consumption.

The NoC architecture can also include additional features, such as quality of service (QoS) mechanisms, flow control, and error correction. QoS mechanisms can ensure that critical traffic is prioritized over non-critical traffic, while flow control can prevent congestion and ensure efficient use of network resources. Error correction can detect and correct errors that occur during data transmission.

Overall, the architecture of a NoC is designed to provide a scalable, efficient, and reliable communication infrastructure for a large number of PEs in a system-on-chip (SoC) design. The design can be customized based on the application requirements for communication efficiency, latency, and power consumption.

2.1 Architecture of SoC-

The architecture of a System on Chip (SoC) typically consists of multiple components integrated onto a single chip[46,48]. These components include:

1. Processor(s): The processor is the core of the SoC, and it can be a general-purpose processor, a specialized processor, or a combination of both. The processor can also include caches, memory management units (MMUs), and other peripherals.
2. Memory: The memory can be integrated onto the same chip as the processor, and it can include both volatile (e.g. DRAM) and non-volatile (e.g. flash) memory.
3. Input/Output (I/O) interfaces: The I/O interfaces enable communication between the SoC and external devices, and they can include interfaces for wired and wireless communication, audio and video, storage, and other peripherals.
4. Power management: The power management unit (PMU) is responsible for managing the power consumption of the SoC by controlling the voltage and frequency of the various components.
5. Security: The SoC can also include security features, such as hardware-based encryption, authentication, and secure boot, to protect the device and the data it processes.

The architecture of a SoC can also include additional features, such as on-chip debugging, interrupt controllers, and analog-to-digital converters (ADCs). The design of the SoC architecture can be optimized for specific applications and use cases, such as low power consumption, high performance, or a balance between the two. The SoC can also be customized to include only the necessary components, which can reduce the cost and complexity of the device. Overall, the architecture of a SoC enables the integration of multiple components onto a single chip, which reduces the size, cost, and power consumption of the device. The design of the SoC can be optimized for specific applications, and it can include additional features for power management, security, and other purposes.

3. APPLICATION OF NOC AND SOC

3.1 Network on Chip (NoC) has many applications in various fields, including [1,46]:

1. System on Chip (SoC) design: NoC is widely used in the design of SoCs, which are integrated circuits that contain multiple components such as processors, memory, and input/output interfaces. NoC can provide a high-speed and low-latency interconnect fabric that connects these components and enables efficient communication between them.
2. High-performance computing: NoC can be used in high-performance computing systems, such as supercomputers and data centers, to provide a scalable and efficient interconnect fabric that connects the different nodes and components of the system.
3. Embedded systems: NoC can be used in embedded systems, such as automotive and aerospace systems, to provide a reliable and efficient communication infrastructure that connects the different components of the system.
4. Internet of Things (IoT): NoC can be used in IoT systems to provide a scalable and efficient interconnect fabric that connects the different devices and sensors in the system.
5. Multimedia systems: NoC can be used in multimedia systems, such as video and audio processing systems, to provide a high-speed and low-latency communication infrastructure that connects the different components of the system.
6. Robotics and automation: NoC can be used in robotics and automation systems to provide a reliable and efficient communication infrastructure that connects the different sensors, actuators, and controllers in the system.
7. Artificial intelligence and machine learning: NoC can be used in artificial intelligence and machine learning systems to provide a scalable and efficient interconnect fabric that connects the different processing units, such as CPUs and GPUs, in the system.

Overall, the application of NoC is broad and diverse, and it can provide many benefits such as scalability, efficiency, reliability, and customization in various fields.

3.2 System on Chip (SoC) has numerous applications in various fields, including [46,48]:

1. Mobile devices: SoC is widely used in mobile devices, such as smartphones and tablets, to integrate multiple components, including the CPU, GPU, memory, and wireless communication modules, onto a single chip. This integration reduces power consumption and increases performance and efficiency.
2. Internet of Things (IoT): SoC is used in IoT devices to integrate multiple components, including sensors, microcontrollers, and wireless communication modules, onto a single chip. This integration reduces the size and power consumption of the devices, making them suitable for deployment in various IoT applications.
3. Automotive systems: SoC is used in automotive systems to integrate multiple components, including the microcontroller, power management, and communication modules, onto a single chip. This integration reduces the size and complexity of the systems and increases their reliability and efficiency.
4. Medical devices: SoC is used in medical devices, such as pacemakers and insulin pumps, to integrate multiple components, including sensors, microcontrollers, and wireless communication modules, onto a single chip. This integration reduces the size and power consumption of the devices and increases their reliability and efficiency.
5. Aerospace and defense: SoC is used in aerospace and defense systems to integrate multiple components, including sensors, microcontrollers, and communication modules, onto a single

chip. This integration reduces the size, weight, and power consumption of the systems and increases their reliability and efficiency.

6. Industrial control systems: SoC is used in industrial control systems to integrate multiple components, including microcontrollers, sensors, and communication modules, onto a single chip. This integration reduces the size and complexity of the systems and increases their reliability and efficiency.
7. Wearable devices: SoC is used in wearable devices, such as smartwatches and fitness trackers, to integrate multiple components, including sensors, microcontrollers, and wireless communication modules, onto a single chip. This integration reduces the size and power consumption of the devices and increases their reliability and efficiency.

Overall, the application of SoC is diverse and varied, and it can provide many benefits, such as integration, power efficiency, reliability, and performance, in various fields.

3.3 Security Issue in NoC:-

Security is an important consideration in Network on Chip (NoC) design, as the communication infrastructure of the system is vulnerable to various types of attacks, including:

1. Confidentiality attacks: These attacks aim to extract sensitive information from the NoC, such as communication content or system configuration. They can be performed by eavesdropping on the communication channels or by accessing the memory of the system.
2. Integrity attacks: These attacks aim to modify the communication content or system configuration, either by injecting false data or by altering legitimate data. They can be performed by intercepting and modifying the communication channels or by accessing the memory of the system.
3. Availability attacks: These attacks aim to disrupt the communication infrastructure of the system, either by causing congestion or by blocking the communication channels. They can be performed by flooding the network with traffic or by jamming the communication channels.

3.4 Security Issue in SoC:-

Security is an important consideration in System on Chip (SoC) design, as it can be vulnerable to various types of attacks, such as:

1. Side-channel attacks: These attacks exploit physical characteristics of the system, such as power consumption, electromagnetic radiation, or timing, to extract sensitive information, such as encryption keys or other secret data.
2. Malware attacks: These attacks aim to infect the system with malicious software, which can then compromise the system or extract sensitive information. Malware can be introduced through various means, such as infected software, compromised firmware, or physical access to the system.
3. Hardware attacks: These attacks aim to modify the hardware components of the system, such as the CPU or memory, to extract sensitive information or compromise the system. These attacks can be difficult to detect and mitigate, as they often require physical access to the system.
4. Denial of Service (DoS) attacks: These attacks aim to disrupt the operation of the system, either by flooding the system with traffic or by exploiting vulnerabilities in the system's software or hardware components.

3.5 Comparison between NoC and SoC in Security point of View-

Both Network on Chip (NoC) and System on Chip (SoC) have their own security concerns and considerations[1,46].

NoC primarily deals with communication security, as it provides the communication infrastructure for the system. NoC security concerns include confidentiality, integrity, and availability of communication, and it can be vulnerable to attacks such as eavesdropping, interception, injection, and denial of service. Security techniques such as authentication, encryption, access control, traffic analysis, redundancy, and fault tolerance can be employed to address these concerns.

SoC, on the other hand, deals with both communication security and overall system security. SoC security concerns include confidentiality, integrity, and availability of data and resources, as well as protection against malware, side-channel attacks, and hardware attacks. Security techniques such as hardware security modules, secure communication protocols, secure booting and firmware updates, access control, and authentication can be employed to address these concerns.

In terms of security comparison, NoC and SoC both share similar concerns regarding confidentiality, integrity, and availability, but SoC has additional security concerns due to the nature of the system being a complete computing device. SoC requires protection against a wider range of threats, including those that target the system's software and hardware components, while NoC focuses primarily on communication security.

Overall, both NoC and SoC require a comprehensive security approach that addresses all aspects of the system and its communication infrastructure. While the specific security techniques employed may vary, the goal is to ensure the confidentiality, integrity, and availability of the system and its components.

3.6 Method to detect HT in NoC-

Hardware trojans (HTs) are malicious modifications made to a chip's design or fabrication process that can cause the chip to perform unintended and often malicious functions. HTs can be inserted into a chip's design or fabrication process by an adversary to create a backdoor, leak sensitive information, or cause denial of service [1,48].

Network on Chip (NoC) can be vulnerable to HTs, as it provides the communication infrastructure for the system. HTs in NoC can disrupt the communication flow or inject malicious data into the system. HTs can also compromise the confidentiality and integrity of communication by eavesdropping, intercepting, or modifying communication packets.

To detect and prevent HTs in NoC, several techniques can be employed, such as:

1. Trustworthy design and fabrication: Designers and fabricators can follow best practices to ensure that the NoC is designed and fabricated in a trustworthy environment, using trusted tools and processes.
2. HT detection: Various techniques can be employed to detect HTs in NoC, such as side-channel analysis, fault injection, and formal verification. These techniques can detect any malicious modifications to the NoC and identify their location.
3. HT prevention: Various techniques can be employed to prevent HTs in NoC, such as secure design techniques, insertion of redundancy and diversity, and monitoring of system behavior.

These techniques can make it difficult for an adversary to insert HTs into the NoC and ensure that the NoC operates as intended.

4. Secure communication protocols: Secure communication protocols can be used to protect the confidentiality, integrity, and authenticity of communication in the NoC. Encryption, authentication, and access control can be employed to ensure that only authorized parties can access the communication channels.

Overall, HTs are a serious security threat to NoC, and designers and fabricators should take appropriate measures to detect and prevent them. Employing secure design techniques, trustworthy fabrication processes, HT detection and prevention techniques, and secure communication protocols can help ensure the security of the NoC.

3.7 Method to detect HT in SoC-

Hardware Trojans (HTs) are malicious modifications made to a chip's design or fabrication process that can cause the chip to perform unintended and often malicious functions. HTs can be inserted into a chip's design or fabrication process by an adversary to create a backdoor, leak sensitive information, or cause denial of service[46,48].

System on Chip (SoC) can also be vulnerable to HTs, as it integrates multiple components onto a single chip, including the processing unit, memory, and communication interfaces. HTs in SoC can disrupt the normal operation of the system, leak sensitive information, and create backdoors that can be used by attackers to compromise the system.

To detect and prevent HTs in SoC, several techniques can be employed, such as:

1. Trustworthy design and fabrication: Designers and fabricators can follow best practices to ensure that the SoC is designed and fabricated in a trustworthy environment, using trusted tools and processes.
2. HT detection: Various techniques can be employed to detect HTs in SoC, such as side-channel analysis, fault injection, and formal verification. These techniques can detect any malicious modifications to the SoC and identify their location.
3. HT prevention: Various techniques can be employed to prevent HTs in SoC, such as secure design techniques, insertion of redundancy and diversity, and monitoring of system behavior. These techniques can make it difficult for an adversary to insert HTs into the SoC and ensure that the SoC operates as intended.
4. Secure booting and firmware updates: Secure booting and firmware updates can prevent HTs from being inserted into the system through unauthorized access to the firmware. This can be achieved by using secure booting techniques that validate the integrity of the firmware before it is executed.

Overall, HTs are a serious security threat to SoC, and designers and fabricators should take appropriate measures to detect and prevent them. Employing secure design techniques, trustworthy fabrication processes, HT detection and prevention techniques, and secure booting and firmware updates can help ensure the security of the SoC.

4. RESULT & DISCUSSION

4.1 Optimum Solution for Security in NoC-

To address these security issues, various techniques can be employed in NoC design, such as:

1. Authentication and encryption: These techniques can be used to ensure the confidentiality and integrity of the communication content. Authentication ensures that only authorized parties can access the communication channels, while encryption ensures that the communication content is protected from unauthorized access or modification.
2. Access control: Access control can be used to restrict access to the memory of the system, ensuring that only authorized parties can modify the system configuration or access sensitive information.
3. Traffic analysis: Traffic analysis can be used to detect anomalous traffic patterns or behavior, which may indicate the presence of an attacker. This can be done by monitoring the communication channels and analyzing the traffic characteristics.
4. Redundancy and fault tolerance: These techniques can be used to ensure the availability of the communication infrastructure, even in the presence of attacks or failures. Redundancy can be achieved by using multiple communication channels or nodes, while fault tolerance can be achieved by detecting and recovering from errors or failures.

Overall, security is a critical consideration in NoC design, and various techniques can be employed to ensure the confidentiality, integrity, and availability of the communication infrastructure of the system.

4.2 Optimum Solution for Security in SoC-

To address these security issues, various techniques can be employed in SoC design, such as:

1. Hardware security modules: These modules can be used to provide secure storage and processing of sensitive data, such as encryption keys or authentication tokens. They can also be used to perform secure booting and firmware updates, ensuring that only trusted software is running on the system.
2. Secure communication protocols: These protocols can be used to ensure the confidentiality and integrity of communication between different components of the system. Encryption and authentication can be used to protect the communication content and to ensure that only authorized parties can access the communication channels.
3. Secure booting and firmware updates: These techniques can be used to ensure that only trusted software is running on the system. Secure booting ensures that the system only boots from trusted firmware, while firmware updates can be securely delivered and verified before installation.
4. Access control and authentication: These techniques can be used to restrict access to the system's components and to ensure that only authorized parties can access sensitive data or modify system configuration.

Overall, security is a critical consideration in SoC design, and various techniques can be employed to ensure the confidentiality, integrity, and availability of the system and its components.

5. CONCLUSION

The performance of Network on Chip (NoC) and System on Chip (SoC) depends on various factors, such as the size and complexity of the system, the communication patterns, and the workload. Both NoC and SoC can provide high performance in different scenarios, depending on the specific requirements of the system and the application.

In general, NoC can provide high performance in large-scale systems that require a high-bandwidth, low-latency communication infrastructure. NoC can improve the scalability, performance, and energy efficiency of such systems by providing a communication channel that can be dynamically reconfigured and optimized for the specific workload. NoC can also support various communication protocols and traffic patterns, such as point-to-point, multicast, and broadcast, and can provide fault-tolerance and quality-of-service guarantees.

On the other hand, SoC can provide high performance in systems that require tight integration of multiple components onto a single chip. SoC can improve the performance, power efficiency, and cost of such systems by providing a compact and integrated solution that minimizes the communication overhead and the power consumption. SoC can also support various processing units, memory modules, and communication interfaces, and can be customized for the specific requirements of the application.

Overall, the best performance between NoC and SoC depends on the specific requirements of the system and the application. NoC is better suited for large-scale systems that require a high-bandwidth, low-latency communication infrastructure, while SoC is better suited for systems that require tight integration of multiple components onto a single chip. In many cases, a combination of NoC and SoC can provide the best performance by leveraging the strengths of both technologies.

REFERENCES

[1]. Asadi, B., Zia, S. M., Al-Khafaji, H. M. R., & Mohamadian, A. (2023). Network-on-chip and photonic network-on-chip basic concepts: a survey. *Journal of Electronic Testing*, *39*(1), 11-25.

[2]. Krishna, N. V., Tripathy, R., Marripudi, J., & Soumya, J. (2023, June). Improving Functional Coverage of Network-On-Chip Using Differential Evolution. In *2023 18th Conference on Ph. D Research in Microelectronics and Electronics (PRIME)* (pp. 369-372). IEEE.

[3]. Trik, M., Akhavan, H., Bidgoli, A. M., Molk, A. M. N. G., Vashani, H., & Mozaffari, S. P. (2023). A new adaptive selection strategy for reducing latency in networks on chip. *Integration*, *89*, 9-24.

[4]. Dhavlle, A., Ahmed, M. M., Mansoor, N., Basu, K., Ganguly, A., & PD, S. M. (2023). Defense against On-Chip Trojans Enabling Traffic Analysis Attacks based on Machine Learning and Data Augmentation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.

[5]. Bagga, S., Gupta, R., & Jose, J. (2023, May). Modelling and Analysis of Confluence Attack by Hardware Trojan in NoC. In *Emerging Electronic Devices, Circuits and Systems: Select Proceedings of EEDCS Workshop Held in Conjunction with ISDCS 2022* (pp. 231-246). Singapore: Springer Nature Singapore.

[6]. Kumar, A. S., & Naresh Kumar Reddy, B. (2023). An Efficient Real-Time Embedded Application Mapping for NoC Based Multiprocessor System on Chip. *Wireless Personal Communications*, *128*(4), 2937-2952.

[7]. Monica, K. M. (2023). Design and study of system on chip design for signal processing applications in terms of energy and area. *Materials Today: Proceedings*, *80*, 3252-3262.

[8]. Huang, Z., Zandberg, K., Schleiser, K., & Baccelli, E. (2023). On-Device Evaluation Toolkit for Machine Learning on Heterogeneous Low-Power System-on-Chip. *arXiv preprint arXiv:2306.14574*.

[9]. Achballah AB, Ben Saoud S (2007) *On network-on-chip comparison*. In: Proc. 10th Euromicro Conf. on Digital System Design Architectures, Methods and Tools. pp 503–510

[10]. Agarwal A, Shankar R (2009) *Survey of Network on Chip (NoC) architectures & contributions*. J Eng Comp Arch 3(1).

[11]. Asadi B, Reshadi M (2016) *Photonic Network-on-Chip: A Survey. International Journal of Computer Science and Information Security* (IJCSIS) 14(11):786–792

[12]. Asadi B, Reshadi M, Khademzadeh A (2017) *A routing algorithm for reducing optical loss in photonic Network-on-Chip.* Photonic Netw Commun 34(1):52–62

[13]. Beausoleil RG, Kuekes PJ, Sinder GS, Wang SY, Stanley R (2008) *Nanoelectronic and Nanophotonic Interconnect*. Proc IEEE 96(2):230–247

[14]. Ben Achballah A, Ben Saoud S (2013*) A Survey of Network-on-Chip Tools*. Int J Adv Comput Sci Appl 4(9):61–67

[15]. Benini L, Micheli GD (2002) *Networks on chips: A new SoC paradigm*. Computer 35(1):70–78

[16]. Bergmen K, Carloni LP, Biberman A, Chan J, Hendry G (2014) Photonic network-on-chip Design. Springer

[17]. Bjerregaard T, Mahadevan S (2006) *A survey of research and practices of network-on-chip. ACM Comp* Surv 38(1):1–51

[18]. Borkar S (2007) *thousand core chips: a technology perspective. In: Proc.* 44th ACM/IEEE Design Automation Conf. pp 746–749

[19]. Campobello G, Castano M, Ciofi C, Mangano D (2006) *GALS networks on chip: A new solution for asynchronous delay-insensitive links.* In: Proc. Design, Automation and Test in Europe Conf. pp 160–165

[20]. Chan J, Hendry G, Bergman K, Carloni LP (2011) Physical-Layer Modeling and System-*Level Design of Chip-Scale Photonic Interconnection Networks. IEEE Transaction on Computer-Aided Design of Integrated Circuit and Systems* 30(10):1507–1520

[21]. Chan JW (2012) Architectural exploration and design methodologies of photonic interconnection networks. PhD Dissertation, Columbia University, New York, USA

[22]. Chen J, Gillard P, Li C (2011) Network-on-Chip (NoC) topologies and performance: A review. https://www.semanticscholar.org/, Corpus ID: 102342317

[23]. Chiu G-M (2000) The odd-even turn model for adaptive routing. IEEE Trans Parallel Distrib Syst 11(7):729–738

[24]. Enright Jerger ND, Peh L-S (2009) On-chip networks. Synthesis lectures on computer architecture. https://doi.org/10.2200/S00209ED1V01Y200907CAC008

[25]. Guerrier P, Greiner A (2000) A generic architecture for on-chip packet-switched interconnections. In: Proc. Design, Automation and Test in Europe Conf. pp 250–256

[26]. Hatamirad M, Reza A, Shabani H, Niazmand B, Reshadi M (2012) Loss-Aware Router Design Approach for Dimension-Ordered Routing Algorithms in photonic Networks-on-Chip. International Journal of Computer Science Issues 9(1):337–345

[27]. Hendry G, Kamil S, Biberman A, Chan J, Lee BG, Mohiyuddin M, Bergman K, Carloni LP, Oliker L, Shalf J (2009) Analysis of photonic networks for a chip multiprocessor using scientific applications. In: Proc. 3rd ACM/IEEE International Symposium on Networks-on-Chip. La Jolla, CA, USA, pp 104–113. https://doi.org/10.1109/NOCS.2009.5071458

[28]. Hendry G, Robinson E, Gleyzer V, Chan J, Carloni L, Bliss N (2010) Circuit-switched memory access in photonic interconnection networks for high-performance embedded computing. In: Proc. ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis. New Orleans, LA, USA, pp 1–12

[29]. Hendry G, Robinsonet E, Gleyzer V, Chan J, Carloni LP, Bliss N, Bergmen K (2011) Time-division-multiplexed arbitration in silicon nanophotonic networks-on-chip for high-performance chip multiprocessors. J Parallel Distrib Comput 71(5):641–650

[30]. Hendry GR (2011) Architectures and Design Automation for Photonic Networks on Chip, Columbia University

[31]. Kachris C, Bergman K, Tomkos I (2012) Optical Interconnects for Future Data Center Networks, Springer New York Heidelberg Dordrecht London

[32]. Kachris C, Tomkos I (2012) A Survey on Optical Interconnects for Data Centers. IEEE Comm Surveys Tutorials 14(4)

[33]. Krasteva YE, de la Torre E, Riesgo T (2010) Reconfigurable networks on chip: DRNoC architecture. J Syst Arch: the EUROMICRO Journal 56(7):293–302

[34]. Micheli GD, Seiculescu C, Murali S, Benini L (2010) Networks on chips: From research to products. In: Proc. Design Automation Conf. pp 300–305

[35]. Min R, Ji R, Chen Q, Zhang L, Yang L (2012) A universal method for constructing N-port nonblocking optical router for photonic networks-on-chip. J Light Technol 30(23):3736–3741

[36]. Mo KH, Ye Y, Wu X, Zhang W, Liu W, Xu J (2010) A hierarchical hybrid optical-electronic network-on-chip. In: Proc. IEEE Computer Society Annual Symposium on VLSI. Lixouri, Greece, pp 327–332

[37]. Moadeli M (2010) Quarc: An Architecture for Effcient On-Chip Communication, PhD Thesis, University of Glasgow

[38]. Nikdast M, Xu J (2007) Crosstalk noise and Loss Analysis Platform (CLAP). Hong Kong Univ Sci Technol 1–17. http://www.ece.ust.hk/~eexu/CLAP.html

[39]. Pan Y, Kumar P, Kim J, Memik G, Zhang Y, Choudhary A (2009) Firefly: Illuminating future network-on-chip with nanophotonics. In: Proc. 36th Annual Symposium on Computer Architecture. Austin, Texas, USA, pp 429–440

[40]. Petracca M, Bergman K, Carloni LP (2008) Photonic network-on-chip: Opportunities and challenges. IEEE Int Symp Circuits Syst pp 2789–2792

[41]. Rahimi A, Salehi ME, Mohammadi S, Fakhraie SM, Azarpeyvand A (2010) Energy/throughput trade-off in a fully asynchronous NoC for GALS-based MPSoC architectures. In: Proc. IEEE International Conf. on Design & Technology of Integrated Systems in Nanoscale Era. Hammamet, Tunisia, pp 1–6

[42]. Salminen E, Kulmala A, Hamalainin TD (2008) Survey of network-on-chip proposals. OCP-IP White Paper pp 1–13

[43]. Seiculescu C, Murali S, Benini L, De Micheli G (2009) SunFloor 3D: A tool for networks on chip topology synthesis for 3D systems on chips. In: Proc. Design, Automation and Test in Europe Conf. and Exhibition. pp 9–14

[44]. Shacham A, Bergman K, Carloni LP (2007) On the design of a photonic network-on-chip. In: Proc. First International Symposium on Networks-on-Chip (NOCS'07). Princeton, NJ, USA, pp 53–64

[45]. Shacham A, Bergmen K, Carloni LP (2008) Photonic network-on-chip for future generations of chip multiprocessors. IEEE Trans Comput 57(9):1246–1260

[46]. Shacham A, Lee BG, Chen Q, Carloni LP (2007) Photonic NoC for DMA communications in chip multiprocessors. 15th Annual IEEE Symposium on High-Performance Interconnects (HOTI 2007). Stanford, CA, USA, pp 29–38

[47]. Singh A (2005) Load-balanced routing in interconnection networks. PhD Dissertation, Stanford University, Palo Alto, CA, USA

[48]. Tsai W-C, Lan Y-C, Hu YH, Chen S-J (2012) Networks on chips: Structure and design methodologies. J Electr Comp Eng. https://doi.org/10.1155/2012/509465

[49]. Vaidya AS, Sivasubramaniam A, Das CR (2001) Impact of virtual channels and adaptive routing on application performance. IEEE Trans Parallel Distrib Syst 12(2):223–237

[50]. Vangal SR, Howard J, Ruhl G, Dighe S, Wilson H, Tschanz J (2008) An 80-Tile Sub-100-W TeraFLOPS Processor in 65-nm CMOS. IEEE J Solid-State Circuits 43(1):29–41

[51]. Xie Y, Nikdast M, Xu J, Zhang W, Li Q, Wu X, Ye Y, Wang X, Liu W (2010) Crosstalk noise and bit error rate analysis for optical network-on-chip. In: Proc. Design Automation Conf. (DAC). Anaheim, CA, USA, pp 657–660

[52]. Xie Y, Song T, Zhang Z, He C, Li J, Xu C, Nikdast M, Xu J, Wu X, Zhang W, Ye Y, Wang X, Wang Z, Liu W (2012) Formal worst-case analysis of crosstalk noise in mesh-based optical Networks-on-Chip. IEEE Trans Very Large Scale Integr VLSI Syst 34(15):3550–3562

[53]. Xie Y, Vijaykrishnan N, Das C (2009) Three-dimensional network-on-chip architecture. In: Xie Y, Cong J, Sapatnekar S (eds) Three dimensional integrated circuit design, Springer, pp 189–217

[54]. Zarkesh-Ha P, Beaerra GBP, Forrest S (2010) Hybrid network on chip (HNoC): Local buses with a global mesh architecture. In: Proc. ACM/IEEE International Workshop on System Level Interconnect Prediction. pp 9–14

[55]. Farahmandi, F., Huang, Y., & Mishra, P. (2020). System-on-Chip Security (pp. 173-188). Springer.

[56]. Ray, S., Peeters, E., Tehranipoor, M. M., & Bhunia, S. (2017). System-on-chip platform security assurance: Architecture and validation. Proceedings of the IEEE, 106(1), 21-37.

Authors-

Mr. Ashutosh Dhar Dwivedi is currently working as Assistant Professor in Department of Computer Science & Information Technology, Institute of Hospitality, Management & Sciences Kotdwar Uttrakhand Affiliated to HNB Garhwal Central University. He has eight year experience in Academic filed. He is completed his Master in Technology from Dr APJ Abdul Kalam Technical University in 2017. He is also qualified GATE examination. He has published more than eight research paper in International conference & Scopus Indexed Journal. He has two Indian patents.

Ms. Shreya Chandola is currently working as Assistant Professor in Department CS & IT at Institute of Hospitality Management & Science kotdwar. She has completed her MCA 2019 from GBPIET Ghurdauri, Pauri Garhwal and BSc from PG College kotdwar. Areas of teaching include Data Structure, Software Engineering, Computer based Numerical Techniques and Java Language. She has 2 years' experience in the academic field and 8 months teaching experience as a computer teacher in Lal Bhadur Shastri training institute.