International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

IJISAE

ISSN:2147-6799 www.ijisae.org Original Research Paper

Area-Optimized, Credit-Based, Flow Control Buffered NoC Router

Saylee S. Bidwai[*1], **Dr. Sridhar Iyer**[2], **Dr. Sandeep S. Bidwai**[3]

Submitted: 06/12/2023 **Revised**: 17/01/2024 **Accepted**: 27/01/2024

Abstract*:* The growing popularity of interconnect schemes based on Network-on-Chip (NoC) arises from their exceptional adaptability and scalability. Routers play a significant role in the realm of NoCs, exerting a high impact on performance and cost considerations. To address challenges and enhance the design of NoC routers, the incorporation of numerous innovative techniques becomes essential. We introduce an innovative concept for a NoC router with multiple local ports, developed using Verilog models. Our primary goals encompass the reduction of router size and the enhancement of data transmission speed. The proposed architecture uses XY routing and is further enhanced by optimized buffering, Credit-Based Flow Control, and a Deterministic Clock Approach. The proposed routers are subjected to comprehensive evaluations, scrutinizing their area requirements and operating frequencies. By harnessing distributed control mechanisms, these routers are empowered to operate autonomously, shedding the complexities of intricate handshakes. This, in turn, elevates their overall efficiency and scalability, marking an exceptional breakthrough. The Multi-Local Router design boasts the ability to simultaneously handle multiple independent requests, making it adaptable to high volumes of data traffic in intricate FPSoCs. Its proficiency in meeting essential design criteria such as Power, Performance, and Area (PPA) is truly commendable. To substantiate our claims, we realized and synthesized the proposed router design on a Xilinx Virtex 4 FPGA (4vsx25ff668-12), unequivocally demonstrating its viability and efficacy. This remarkable innovation now opens the door for the implementation of highly efficient NoCs in FPSoCs, particularly for various computationally intensive applications.

Keywords: *FPGA, FPSoc, MLPR, NoC, PPA, SoC*

1. Introduction

Advancements in System-on-Chip (SoC) technology have facilitated the integration of diverse cores, ranging from basic memory to intricate DSPs, onto a solitary chip [1]. Consequently, the complexity of communication between these components has escalated, commensurate with the increasing number of on-chip processing elements. Addressing this challenge, the incorporation of Network-on-Chip (NoC) in SoCs emerges as a viable solution, offering potential resolutions to intricate communication issues in complex SoC architectures [2]. To expedite data transmission across the chip, while minimizing design impact and adhering to critical design criteria like Power, Performance, and Area (PPA), NoC enables physical interconnections among processing components via a network of routers and links, surmounting the limitations of traditional shared bus architectures.

The efficiency of NoC primarily hinges on the prowess of the router, which assumes a vital role in orchestrating data transfers within the network. As the number of network cores grows in proportion, the need for an effective router design becomes paramount. The efficiency of the router is influenced by several factors, including the chosen NoC architecture (Synchronous, Asynchronous, or GALS), arbiter design, buffer size, network topology, and routing technique [3]. By diligently optimizing these aspects, designers can attain the desired system performance and effective communication among the on-chip processing cores.

In the Synchronous NoC architecture, routers operate under a global clock, albeit leading to heightened power consumption. Although Synchronous designs are swift and area-efficient, implementing them at high frequencies poses challenges and may result in Electromagnetic Interference (EMI) issues [4]. In response to the problems arising from global clock distribution in Synchronous NoCs, researchers have explored intermediate solutions such as GALS (Globally Asynchronous and Locally Synchronous) [5]. GALS architecture partitions the system into smaller synchronous regions, eliminating the necessity for a global clock. Some Network-on-Chips (NoCs) have been put forth to cater to both Synchronous (end-to-end path) and Asynchronous communication (NOC-IP) [6], with the primary goal of achieving reduced energy consumption and improved latency. In this context, a highly energy-efficient Synchronous-Asynchronous Circuit switched NoC has been introduced [7], incorporating two sub-routers - one for Synchronous control and the other for Asynchronous data transfer. Conversely, Asynchronous NoC designs operate

1Research Scholar, MIT Academy of Engineering, Alandi, Pune, Pin-412105, India and SGBIT Belagavi
ORCID ID : 0000-0001-5172-5616
2Dept. of CSE(AI), KLE Technological University Dr. MSSCET, Belagavi, Pin-590008, Karnataka, India
ORCID ID : 0000-0002-8466-3316,
3Department of E&Tec, Army Institute of Technology, Alandi-road, Dighi, Pin-411015, Pune, Maharashtra
ORCID ID : 0000-0001-7852-5438
** Corresponding Author Email: sayleebidwai@gmail.com*

without a global clock, endowing them with superior power efficiency, albeit at a relatively slower pace. These Asynchronous designs are particularly well-suited for real-time applications, where the transmission of small data packets while adhering to strict power constraints holds paramount importance. On the other hand, Synchronous designs prove more favorable for scenarios involving large data packets and continuous transmission, such as multimedia applications [8]. Several Asynchronous NoC designs have been presented by diligent researchers, among which is the Bundled data logic approach, offering high throughput with simple hardware requirements but susceptible to timing variations. Furthermore, a 2 Φ click-based Mesh NoC with bonded bundled data has been proposed to improve latency [9]. These diverse approaches cater to specific needs and challenges of different applications, providing flexibility and optimization in Network-on-Chip designs.

Selecting an appropriate routing protocol profoundly influences NoC design. Opting for a complex routing method may render router design more intricate, consequently increasing power consumption and chip area. Conversely, employing a straightforward routing protocol may be advantageous in terms of cost and energy efficiency, but it might not ensure optimal traffic routing throughout the network.

Another critical parameter is the buffer size, playing a crucial role in storing data packets, thereby mitigating packet dropping and misrouting [10]. However, buffers come at a cost of significant power consumption (due to dynamic power during read/write operations) and chip area (due to static power when empty). For example, in [11], input buffers occupy a substantial 75% of the network area.

The proposed routers are thoroughly evaluated based on area and operating frequency. The use of distributed control allows the routers to operate independently without complicated handshakes, which further enhances their efficiency and scalability.

The proposed router's architecture, with its Input Channels, Cross Switch Matrix, and Output Channels, forms a critical component of the Network-on-Chip (NOC) system, enabling effective data routing and communication between different cores and routers within the NOC network.

2. Literature Review

In recent years, significant research efforts have been devoted to improving Network-on-Chip (NOC) router architecture to address the challenges posed by complex System-on-Chip (SOC) designs. Several studies have explored novel router designs to enhance performance, power efficiency, and scalability of on-chip communication. The authors of [12] proposes a highly efficient and scalable NOC router architecture. The authors introduce a novel dynamic routing algorithm that effectively balances network traffic and minimizes congestion. Additionally, the router incorporates a lightweight buffer management system that optimizes buffer utilization while reducing power consumption. The authors of [13] propose a hybrid NOC router design that combines the benefits of both synchronous and asynchronous operation. The hybrid router architecture utilizes GALS principles to eliminate global clock distribution issues, while also providing efficient asynchronous communication between routers. This design achieves a balance between performance and power consumption in highly integrated SOC designs. The authors of [14] introduces a bufferless routing algorithm for Network-on-Chip routers. CHIPPER employs a deflection-based mechanism to handle network congestion without the need for on-chip buffers. It achieves low-latency communication and reduces the area and power overhead associated with buffers. The authors of [15] presented another bufferless routing algorithm designed for mesh-based Network-on-Chip architectures. This approach uses deflection routing to steer packets around congested areas, eliminating the need for per-router buffers. BLESS provides a scalable and power-efficient solution for on-chip communication. The authors of [16] focused on bufferless deflection routing in 2D mesh Network-on-Chip architectures. The paper presents a comprehensive analysis of the benefits and challenges of bufferless routing and discusses various design considerations to optimize performance. The authors of [17] evaluated the performance of bufferless deflection routers in NOCs through simulation and analysis. The research explores the impact of different traffic patterns and network sizes on the efficiency and scalability of bufferless routing algorithms. The authors of [18] proposed the Bypass Channel Router, a bufferless NOC router tailored for high-performance many-core chips. The router design leverages a bypass channel mechanism to achieve low-latency data transfer and reduce buffer-related issues. Recently, some researchers have explored the concept of buffer-less routing algorithms as an alternative approach to reduce power and area consumption. The choice between Buffer-less routing and minimally buffered deflection routing depends on the specific requirements and characteristics of the network. While Buffer-less routing is advantageous for lower network loads, minimally buffered deflection routing offers a viable option for networks with higher traffic demands, achieving a balance between buffer-less efficiency and buffering benefits to ensure optimized performance and power consumption in diverse on-chip communication scenarios. routers have gained attention due to their potential to reduce power consumption and improve robustness. Researchers have explored novel asynchronous router architectures and communication techniques [19] introduces an asynchronous NOC router with virtual channels to mitigate the effects of congestion and improve performance. The authors of [20] proposed an energy-efficient asynchronous NOC router design that leverages self-timed circuits to reduce power consumption. The router incorporates novel techniques for dynamic voltage scaling to further improve energy efficiency. The authors of [21] presented a low-power asynchronous NOC router design that employs dual-rail encoding to reduce power dissipation. The GALS approach allows for localized synchronization within the router while enabling asynchronous communication across the network. This paper introduces a buffered crossbar router architecture for on-chip communication. The router employs a crossbar-based switching fabric to enable simultaneous connections between multiple inputs and outputs, allowing for high throughput. Additionally, the router incorporates buffers at the input and output ports to handle contention and improve data flow efficiency. The authors of [22] presented a high-performance buffered crossbar router design optimized for Network-on-Chip applications. The router utilizes a crossbar switch to provide non-blocking communication paths, and it incorporates buffering mechanisms to handle congestion and improve overall system performance. The authors of [23] evaluated the design and performance of a buffered crossbar switching router for Network-on-Chip architectures. The research analyzes the impact of different buffer sizes and configurations on router efficiency and network throughput. The authors of [8] proposed an energy-efficient buffered crossbar router design for on-chip communication. The router employs power-saving techniques, such as adaptive buffering and clock gating, to reduce energy consumption while maintaining high-performance data transfer.

3. Proposed Methodology

The proposed Network-on-Chip (NOC) architecture comprises two main components: one is an area and low-power router design, and the other is the Network Interface (NI) along with a traffic generator.

3.1. Router Design

Routers are fundamental switching elements in the NOC responsible for efficiently forwarding data packets from the source core to the destination core. mesh-based NOC, each router has four directional ports (North, East, West, and South) to facilitate communication between neighbouring routers. Additionally, there is a local port through which the design core is connected to the router. When a data packet is generated by the source design core, it is addressed to the router connected to the destination core.

3.2. IP Core of Router Design

In a Network-on-Chip (NOC) based design, a router or switch is a core element responsible for directing data in the proper direction, enabling efficient communication between different cores or processing elements within the chip. The proposed router architecture of MLPR (Multi-Lane Parallel Router) is depicted in Fig. 1.

The MLPR router is designed with multiple ports, each serving a specific purpose. Among these ports, four are directional ports, namely North (N), East (E), South (S), and West (W). The presence of directional ports empowers the router to navigate outgoing data packets towards any of the four cardinal directions, determined by the location of the destination core. For instance, when a data packet is destined for a core situated to the east of the present router, the router will dutifully guide the packet through the East port.

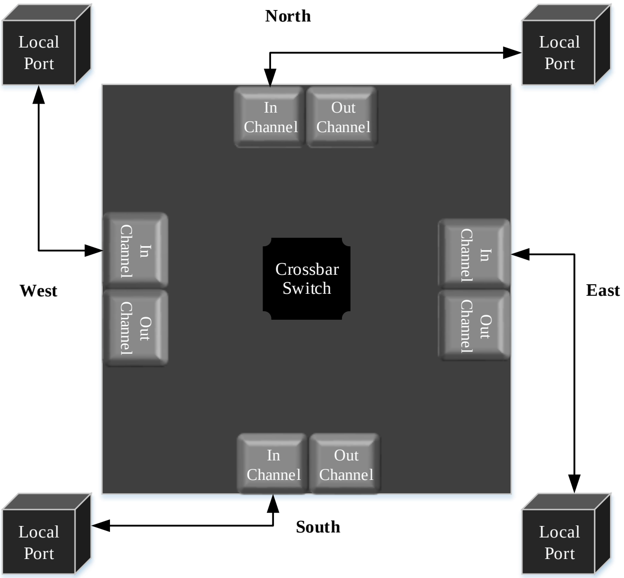

Fig. 1. Architecture of Multi Local Port Router (MLPR)

3.3. Implementation of MLPR

Illustrated in Fig. 2, the block-level schematic of the envisioned router comprises three primary components: The Input Channel, Cross Switch Matrix, and Output Channels. Each of these constituents assumes a pivotal role in enabling data transfer and routing within the router.

- Input Channel: The Input Channel assumes a pivotal role in the MLPR (Multi-Local Port Router), serving as the ingress point for incoming data packets from connected processing elements or design cores. This critical component is tasked with the initial data reception within the router. It diligently gathers data packets from the attached cores and prepares them for further processing, encompassing packet segmentation, header extraction, and other essential tasks for routing information extraction. Within the MLPR router, the Input Channel emerges as a crucial entity, responsible for handling incoming data packets and skillfully managing their transfer throughout the router. Fig. 3 showcases the block diagram of this Input Channel, which is thoughtfully composed of a Buffer, Control Logic, and XY Routing.
- Buffer: Each port embedded in the Input Channel houses a dedicated buffer to diligently store incoming data upon its arrival at the router. These buffers are thoughtfully designed to embrace a First-In-First-Out (FIFO) structure, boasting a depth of 16 bits and a width of 8 bits. As incoming data is received, it finds a temporary abode within the FIFO buffer, eagerly awaiting further processing and onward transmission to the appropriate output channel.
- Control Logic: The Input Channel boasts an area of autonomous control through its exclusive Control Logic, ingeniously implemented as a Finite State Machine (FSM) Controller. This intelligent Control Logic deftly manages various aspects of data transfer and communication within the Input Channel. It diligently governs the read and write operations for the FIFO buffer while orchestrating the request and grant signals for seamless data transfer between the input and output channels. Additionally, the Control Logic deftly handles acknowledge signals for request signals emanating from neighboring routers or processing elements.

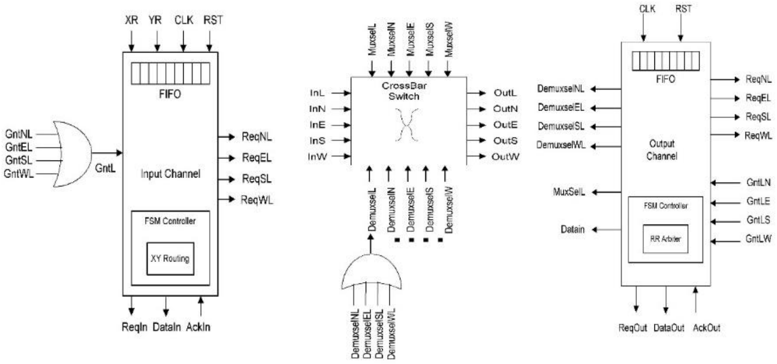

Fig. 2. MLPR Input Channel, Crossbar Switch, and Output Channel

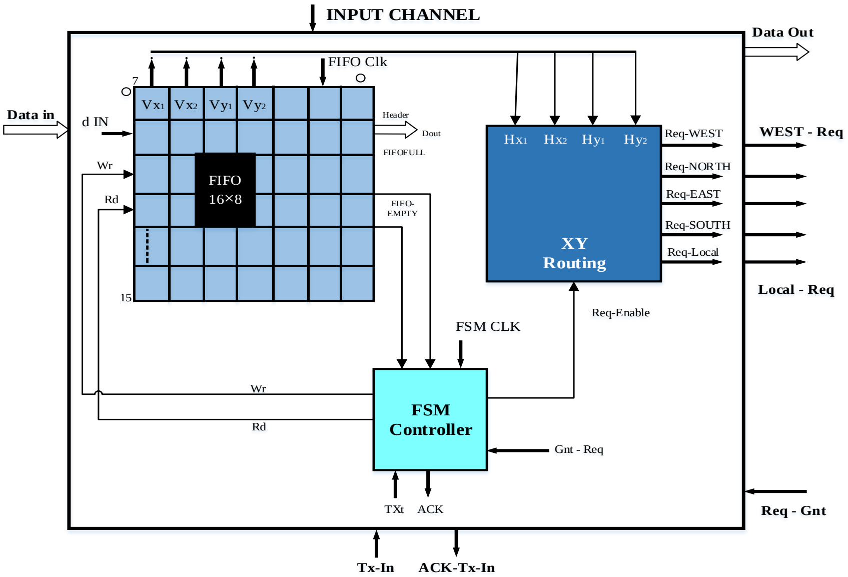

Fig. 3. Input channel block diagram

The data transfer procedure within the Input Channel can be succinctly summarized thus:

The Output Channel consists of the following main components:

- 8-Bit FIFO: Within each Output Channel, a dedicated 8-bit FIFO proudly stands, boasting an impressive depth of 16. This FIFO gracefully undertakes the role of temporarily accommodating data packets before embarking on their journey to neighboring routers or processing elements. When multiple data requests from distinct input channels descend upon the Output Channel, a Round Robin Arbiter (RRA) adroitly orchestrates the arbitration process, selecting the most deserving request for immediate processing and subsequent storage within the FIFO.
- Control Logic (FSM): Nestled within the confines of the Output Channel resides an ingenious Control Logic, brought to life through the implementation of a Finite State Machine (FSM). This illustrious FSM dons the mantle of making crucial arbitration decisions amidst the influx of multiple incoming data requests from diverse input channels. By leveraging the Round Robin Arbiter, the FSM diligently ensures equitable and judicious data transfer across all input channels, thereby fostering a harmonious and balanced ecosystem. Once the RRA grants a request its due, the FSM deftly activates the control bit lines of the Crossbar Switch, masterfully paving the way for the establishment of the required connection that facilitates seamless data transfer between the input and output channels.
- Round Robin Arbiter (RRA): The Round Robin Arbiter, enlisted as an indispensable ally of the Control Logic FSM, emerges as a fair and impartial judge, conscientiously selecting and prioritizing data requests from the multitudes of input channels. The RRA leaves no room for favoritism, ensuring that every input channel relishes an equal opportunity to partake in the delightful dance of data exchange with the Output Channel.
- Handshake Mechanism: Following the gracious receipt of a data packet from one of the input channels, lovingly embraced within its FIFO, the FSM gracefully sets the wheels in motion for data transmission towards the neighboring router, employing the eloquent dance of a handshake mechanism. This charming gesture guarantees not only reliable data transfer but also impeccable synchronization between the Output Channel and the esteemed neighboring router.
- Crossbar Switch Control: Exercising its divine authority, the Output Channel gallantly orchestrates the setting of control bit lines within the Crossbar Switch, thereby meticulously forging the requisite connection between the input channel yearning for data transfer and the Output Channel itself. This monumental feat blesses data with its intended route, ensuring a seamless and precise journey from the input channel to its cherished destination, the Output Channel.

The Output Channel plays a critical role in managing data transfer from the router to the neighboring routers or processing elements. Its Control Logic and Round Robin Arbiter ensure fair arbitration among input channels, enabling efficient and reliable data routing within the MLPR router and the Network-on-Chip architecture XY Routing is an essential part of the MLPR router's functionality, where data packets are directed through the router based on their destination coordinates.

The Input Channel's XY Routing process involves a sequence of distinctive stages:

- Horizontal Displacement: When the Input Channel's FIFO reaches its capacity, it assesses the X-coordinate of the target router (H_x) against the locally stored X-coordinate of the present router. Should H_x exceed the router's X-coordinate (X), it indicates an eastern destination. Consequently, the data packet is directed to the East port of the router. Conversely, if H_x is less than the router's X-coordinate, the packet is channeled to the West port.
- Vertical Displacement: In situations where H_x matches the router's X-coordinate, the packet is deemed to have arrived at the target column of routers and is ready for vertical displacement. At this stage, the Y-coordinate of the destination router (H_y) is cross-checked with the router's local Y-coordinate (Y). Should H_y exceed Y, the packet proceeds to the North port for vertical movement. Conversely, if H_y is less than Y, the packet is forwarded to the South port.
- Final Destination: An H_y equivalent to the router's Y-coordinate signifies that the packet has reached the destination router. Thus, the packet is sent to the local port of the router, culminating the routing process.

A crucial optimization decision has led to substantial resource savings in the router design within the XY Routing process. By forwarding packets horizontally until they reach the target column and then routing them vertically to the destination router, there's no need for North or South input ports to request access to the East or West output ports, respectively. Consequently, the FSMs (Finite State Machines) of the mentioned output channels (East and West) are simplified, as they no longer require servicing the aforementioned input ports.

The result of this optimization is a noteworthy reduction in area utilization and a decrease in the number of clock cycles needed to fulfil requests. The implementation of a Multi-local port router benefits from minimal area overhead and maintains an acceptable level of performance. In this way, the MLPR router proves to be a more efficient and effective solution for data routing tasks.

3.4. Traffic Generator

The traffic generator (TG) simulates flows of data coming out of a heart sent to the communication architecture.

A deterministic TG allows for some sort of model the communications that the IP blocks connected to the NoC emit from the trace left by these. This type of TG can generate precise transactions over time, size and the idle time that correspond to the behaviour of an IP connected to the NoC. This type is used for a complete system (type and number of nodes) and for an application given. The advantages of these traffic generators are high accuracy and the factor acceleration for emulation compared to the simulation of all traffic.

In this work, deterministic traffic generators are used, with the aim of defining performance of NoC for spectral applications. Two packet formats are available. The first format, Data in an FPGA (case of implementation of the NoC on a mono-FPGA architecture). The second is dedicated to data transfers over several FPGAs (multi-FPGA architecture). Within our emulation platform, each packet consists of two main sections: a header and a data portion. These sections collectively store vital information necessary for proper functioning. Notably, the header contains the destination node address (Dest) and the initiator's address (Source). Additionally, it houses the Initiator Clock (Clk_init), a significant flit utilized for latency evaluation. The data within Clk_init corresponds to the clock stroke when the packet is dispatched. Furthermore, the packet size (Sz_pckt), as well as the Ext_cpt (solely applicable for multi-FPGA setups) denoting the number of cycles for inter-FPGA packet transfers, and the count of transmitted packets (Nb_pckt) are also included.

Moving on to Fig. 4, behold the Signals and Parameters governing the Generic Traffic Generators. Behold a visual representation of the TG, seamlessly integrated into our flow. The TG operates by generating control signals, namely router_rx and router_ack_rx, while simultaneously producing data packets at the data_in output. Notably, the packet size aligns harmoniously with the bus size, facilitating efficient data transfer.

Significantly, the various packet quantities and formats are subject to the specific traffic scenario outlined in the package Data_transfer, elaborated on later in this paper. It is pertinent to mention that our refined TG is coded in generic VHDL, strategically inserted into the TG and TR library within the flow.

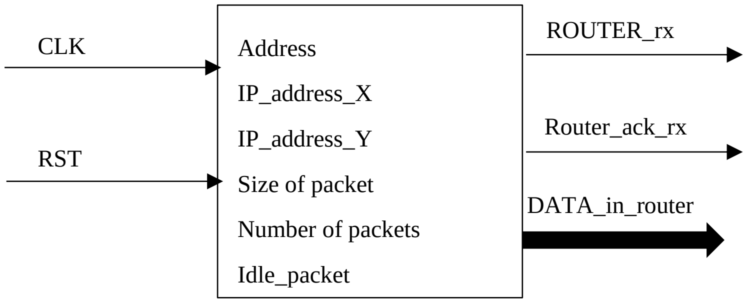

Fig. 4. Signals and Parameters for Generic Traffic Generators

4. Results and Discussion

The proposed Asynchronous NOC architecture with Buffered Router is modelled in VHDL using Xilinx ISE 14.7 with the Virtex 7 series, using the xc7a100t device, and using the 3csg324 package. ISE Simulator (iSIM) is used for simulation.

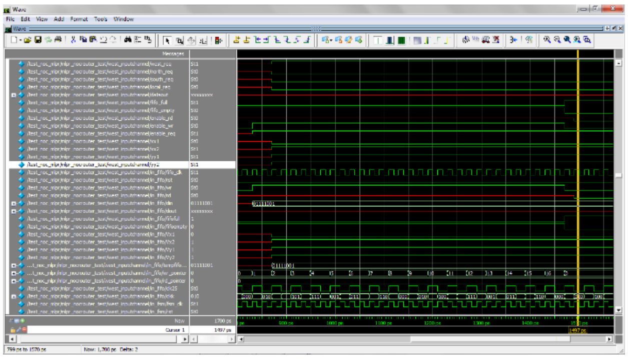

Fig. 5. Simulation Result of West Node Input Channel

Fig. 5 shows the simulation result of west node Input Channel. It is responsible for taking input from processor and making initial handshaking. Once handshaking is established then this channel start processing the information. The XY router take decision of routing of data. There are four directions: east, local, south and north, which is defined then it generates the grant signal which further moves to cross bar switch.

The above simulation result is given for generating grant signal as a decoder and further this grant signal act as selection for multiplexer. Based on the multiplexer the control and data can processed to further to respective node.

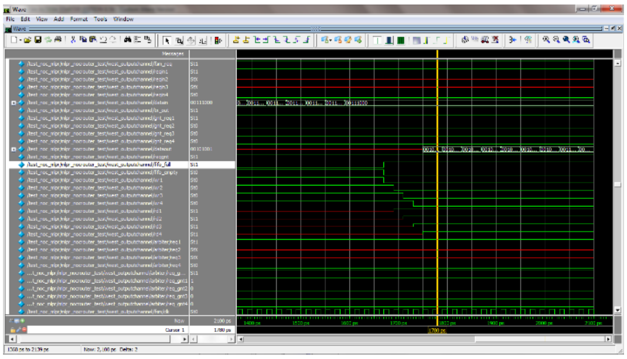

Fig. 6. Simulation Result of Output Channel

The above simulation result shows the simulation behaviour of out channel, Arbiter will take care of the request and further based on grant data comes from crossbar and is stored in FIFO of the out channel. This block is responsible for handshaking between internal node and outside node.

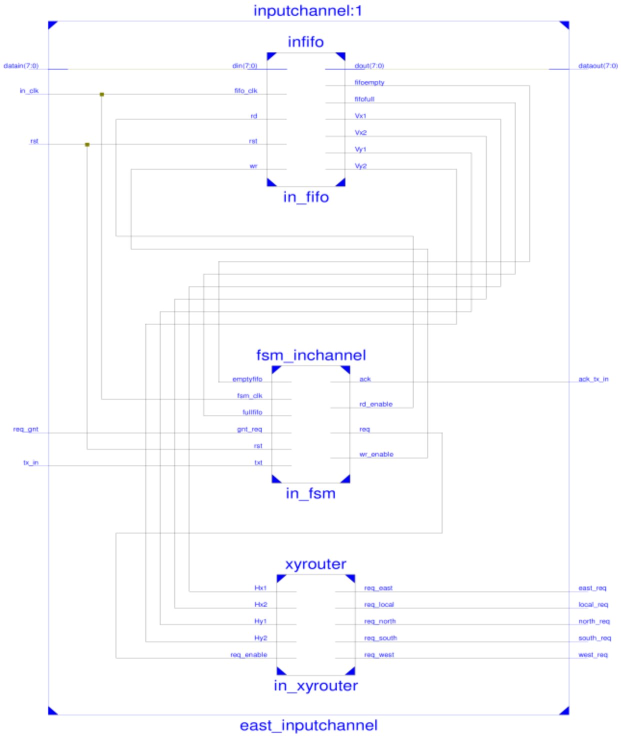

Fig. 7. Input Channel of Router

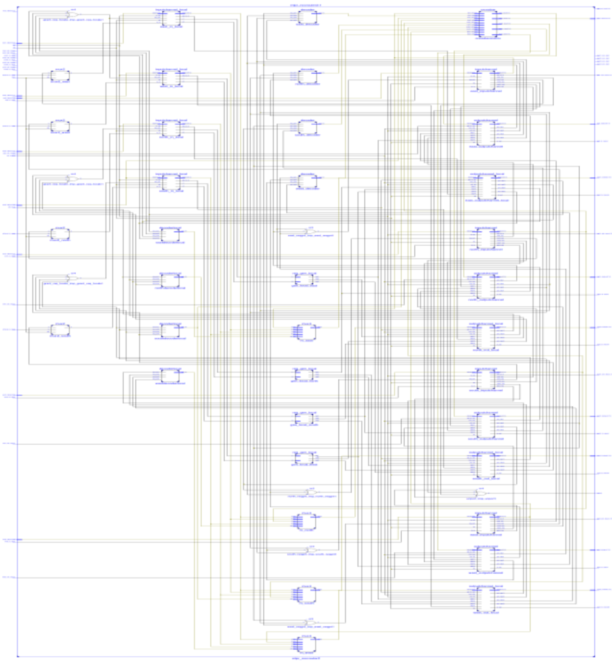

Fig. 8. RTL View of MLPR NOC Router

Selected Device: 4vsx25ff668-12

Table 1. Device utilization summary

Logic Utilization	Used	Available	Utilization
No. of Slice Registers	2360	10240	23%
Number of Slice LUT's	4277	20480	11%
Number of fully used LUT-FF pair	2349	20480	20%
Number of bonded IOBs	170	320	53%

Form Table 1, the provided logic utilization results for the proposed Asynchronous NOC architecture with Buffered Router implemented on the Xilinx Virtex 7 FPGA (xc7a100t device, 3csg324 package), the design appears to be relatively well-optimized and efficiently utilizes the available hardware resources. The design uses 2360 out of 10240 available Slice Registers, which accounts for 23% of the available resources. Slice Registers are used to store data and control signals, so the usage percentage indicates that a moderate amount of sequential logic is present in the design. Similarly, 4277 out of 20480 available Slice LUTs have been utilized, representing 11% of the total resources. Slice LUTs are used for implementing combinational logic functions, and the utilization percentage suggests that the design makes efficient use of these resources. Out of the Slice LUTs and Flip-Flop pairs, 2349 pairs have been fully utilized, which makes up 20% of the available pairs. Fully utilized LUT-FF pairs indicate optimized logic design, with efficient mapping of combinational logic to the available resources. Regarding the bonded IOBs, the design uses 170 out of the 320 available, resulting in a utilization of 53%. This suggests that the design interfaces with external devices or other FPGA components effectively, using slightly over half of the available IOBs.

Overall, the utilization percentages indicate a well-optimized design that efficiently uses the available resources, leaving room for further enhancements or additional features if needed. However, it's essential to consider any performance requirements and constraints specific to the application to ensure the design meets its objectives while staying within the FPGA's capabilities.

Table 2. Comparative analysis of NoC router implementation

Resources used in FPGA	Proposed	[24]	[25]	[26]	[27]	[28]
Number of Slice of Registers	2360	2978	3690	2408	2367	3258
Number of Slice of LUTs	4277	3544	2980	3964	5789	3913
Number of fully used LUTs and FF pairs	2349	2895	2400	3429	3558	2124
Frequency	223.098MHz	180MHz	210MHz	175MHz	198MHz	168.4MHz
Power	38mW	56mW	--	74mW	54 mW	43mW

Based on the comparative analysis of the NOC router implementation, it is evident that there are two different router designs being compared. The "Proposed" router implementation has been compared against an alternative design represented by the "[24]" data. The "Proposed" router utilizes 2360 slice registers and 4277 slice LUTs on the FPGA. In contrast, the alternative router design denoted by "[24]" uses 2978 slice registers and 3544 slice LUTs. This indicates that the "Proposed" router uses fewer slice registers but slightly more slice LUTs compared to the alternative design. It suggests that the "Proposed" router might be more efficient in terms of sequential logic utilization but requires slightly more combinational logic resources. When considering fully used LUTs and flip-flop pairs, the "Proposed" router again outperforms the alternative design. It fully utilizes 2349 LUT-FF pairs compared to 2895 pairs in the other design. This shows that the "Proposed" router has better optimization and mapping of combinational logic to the available resources. Regarding operating frequency, the "Proposed" router achieves 223.098MHz, while the alternative design operates at a slightly lower frequency of 180MHz. This suggests that the "Proposed" router design may offer better performance and higher throughput, as it can process data at a faster rate. In terms of power consumption, the "Proposed" router demonstrates superior efficiency, consuming only 38mW compared to 56mW in the alternative design. Lower power consumption is desirable as it leads to reduced heat dissipation and can contribute to better overall system energy efficiency.

5. Conclusion

In this concluding segment, we embark on an exploration of the potential that Field-Programmable-Systems-on-Chip (FPSoCs) hold as reliable and efficient digital systems, tailored for modern applications with intensive computational requirements and compact form factors. As a strategic means to surmount the limitations posed by traditional bus-based and point-to-point on-chip communications in System-on-Chip (SoC) designs, the Network-on-Chip (NoC) emerges as the favored interconnect approach. Nevertheless, extensive research is indispensable to thoroughly investigate the design prospects of FPGA-based NoCs and to devise more effective resolutions to current NoC challenges.

In the field of FPGA-based NoCs, our research makes a substantial contribution by proposing efficient and area-optimized solutions for NoC router design. Rigorous implementation and evaluation of our proposed router design on Xilinx Spartan 3 FPGAs have unequivocally demonstrated its feasibility and potential for practical applications in FPSoCs. By tenaciously delving into the design potential of FPGA-based NoCs and introducing even more streamlined solutions, we can confidently anticipate further enhancements in the performance and capabilities of FPSoCs, particularly concerning computationally intensive applications.

Acknowledgements

The authors would like to express their gratitude to MIT Academy of Engineering, Alandi, Pune for all of their assistance and encouragement in carrying out this research and publishing this paper.

Author contributions

Saylee S. Bidwai is the principal author responsible for the study's conception and design, overseeing experimental procedures, and composing the manuscript. She adeptly generated graphical representations, and made substantial contributions to manuscript development. She was actively engaged in study design, offering invaluable insights during results analysis, and precisely revising the manuscript. Dr. Sridhar Iyer and Sandeep S. Bidwai served as project supervisors, each contributing critical assessments of the manuscript, with Dr. Sridhar Iyer serving as the research supervisor and Sandeep S. Bidwai as the co-supervisor.

Conflicts of interest

The authors declare no conflict of interest.

References

[1] Jain, A., Dwivedi, R. K., Alshazly, H., Kumar, A., Bourouis, S., & Kaur, M. (2022). Design and simulation of ring network-on-chip for different configured nodes. *Computers, Materials & Continua*, *71*(2), 4085-4100.

[2] Kumar, N. A., Priyan, S. V., Venkatramana, P., & Nandan, D. (2022). Routing Strategy: Network-on-Chip Architectures. *VLSI Architecture for Signal, Speech, and Image Processing*, 167-197.

[3] Naqvi, M. R. (2021). Low power network on chip architectures: A survey. *Computer Science and Information Technologies*, *2*(3), 158-168.

[4] Yazdanpanah, F. (2023). A two-level network-on-chip architecture with multicast support. *Journal of Parallel and Distributed Computing*, *172*, 114-130.

[5] Tran, L. D., Felipe, A. L. S., & Matthews, G. I. (2022, December). The Need for 2-Phase Design Paradigms in High-Throughput GALS Network-On-Chip Architectures. In *2022 RIVF International Conference on Computing and Communication Technologies (RIVF)* (pp. 305-310). IEEE.

[6] Tahanian, E., Rezvani, M., & Fateh, M. (2021, March). A novel wireless network-on-chip architecture for multicore systems. In *2021 26th International Computer Conference, Computer Society of Iran (CSICC)* (pp. 1-8). IEEE.

[7] Charles, S., & Mishra, P. (2020). Reconfigurable network-on-chip security architecture. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, *25*(6), 1-25.

[8] Sharma, S. K., Jain, A., Gupta, K., Prasad, D., & Singh, V. (2019). An internal schematic view and simulation of major diagonal mesh network-on-chip. *Journal of Computational and Theoretical Nanoscience*, *16*(10), 4412-4417.

[9] Sacanamboy, M. (2022). Heuristic algorithm for task mapping problem in a hierarchical wireless network-on-chip architecture. *Cluster Computing*, 1-17.

[10] Amin, W., Hussain, F., Anjum, S., Khan, S., Baloch, N. K., Nain, Z., & Kim, S. W. (2020). Performance evaluation of application mapping approaches for network-on-chip designs. *IEEE Access*, *8*, 63607-63631.

[11] Wang, K., Louri, A., Karanth, A., & Bunescu, R. (2019, March). High-performance, energy-efficient, fault-tolerant network-on-chip design using reinforcement learning. In *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)* (pp. 1166-1171). IEEE.

[12] Wang, L., Wang, X., & Wang, Y. (2019). An approximate bufferless network-on-chip. *IEEE Access*, *7*, 141516-141532.

[13] Xiang, X., Sigdel, P., & Tzeng, N. F. (2019). Bufferless network-on-chips with bridged multiple subnetworks for deflection reduction and energy savings. *IEEE Transactions on Computers*, *69*(4), 577-590.

[14] Venkataraman, N. L., Kumar, R., & Shakeel, P. M. (2020). Ant lion optimized bufferless routing in the design of low power application specific network on chip. *Circuits, Systems, and Signal Processing*, *39*, 961-976.

[15] Mandal, S. K., Krishnakumar, A., & Ogras, U. Y. (2021). Energy-efficient networks-on-chip architectures: Design and run-time optimization. *Network-on-Chip Security and Privacy*, 55-75.

[16] Kunthara, R. G., Neethu, K., James, R. K., Sleeba, S. Z., & Jose, J. (2019, October). DoLaR: double layer routing for Bufferless mesh network-on-chip. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)* (pp. 400-405). IEEE.

[17] Arulananth, T. S., Baskar, M. S. M. U. S., SM, U. S., Thiagarajan, R., Rajeshwari, P. R., Kumar, A. S., & Suresh, A. (2021). Evaluation of low power consumption network on chip routing architecture. *Microprocessors and Microsystems*, *82*, 103809.

[18] Xiang, X., Sigdel, P., & Tzeng, N. F. (2019). Bufferless network-on-chips with bridged multiple subnetworks for deflection reduction and energy savings. *IEEE Transactions on Computers*, *69*(4), 577-590.

[19] Zitouni, A., & Chemli, B. (2021). Asynchronous dynamic arbiter for network on chip. *International Journal of Computer Applications in Technology*, *67*(4), 370-382.

[20] Siddagangappa, R. (2022, May). Asynchronous NoC with Fault tolerant mechanism: A Comprehensive

Review. In *2022 Trends in Electrical, Electronics, Computer Engineering Conference (TEECCON)* (pp. 84-92). IEEE.

[21] Thonnart, Y., Vivet, P., Agarwal, S., & Chauhan, R. (2019, May). Latency improvement of an industrial SoC system interconnect using an asynchronous NoC backbone. In *2019 25th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC)* (pp. 46-47). IEEE.

[22] Gogula, S., & Damodaran, V. (2023, April). Design of a VLSI Router for the Faster Data Transmission Using Buffer. In *2023 2nd International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)* (pp. 1-5). IEEE.

[23] Fard, E. S., Jamali, M. A. J., Masdari, M., & Majidzadeh, K. (2022). An efficient NoC router by optimal management of buffer read and write mechanism. *Microprocessors and Microsystems*, *89*, 104440.

[24] Patil, T., & Sandi, A. (2022). Design and implementation of asynchronous NOC architecture with buffered router. *Materials Today: Proceedings*, *49*, 756-763.

[25] Katta, M., Ramesh, T. K., & Plosila, J. (2021). SB-Router: A swapped buffer activated low latency network-on-chip router. *IEEE Access*, *9*, 126564-126578.

[26] Fard, E. S., Jamali, M. A. J., Masdari, M., & Majidzadeh, K. (2022). An efficient NoC router by optimal management of buffer read and write mechanism. *Microprocessors and Microsystems*, *89*, 104440.

[27] Jafarzadeh, N., Jalili, A., Alzubi, J. A., Rezaee, K., Liu, Y., Gheisari, M., ... & Javadpour, A. (2023). A novel buffering fault-tolerance approach for network on chip (NoC). *IET Circuits, Devices & Systems*, *17*(4), 250-257.

[28] Nagaraju, S., Balasundaram, S., Balasundaram, R., & Kumar, R. K. (2022). Energy Efficient FSM based Elastic Buffer Routing Computation for NoC. *Journal of Optoelectronics Laser*, *41*(9), 2022.