International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Comparative Analysis of FPGA with Network on Chip Architecture with Routing Techniques

E. G. Satish[1], Rajeshwari S. B.[2], Vinay T. R.[3], Asmathunnisa N.[4], Kalpitha S. Naik[5], Jagadish S. Kallimani[6]

Submitted: 09/01/2024 **Revised**: 15/02/2024 **Accepted**: 23/02/2024

Abstract: The reprogrammable device architectures may bring one-of-a-kind computational abilities to play out the different undertakings. It will help to execution and vitality effectiveness of equipment utilities with the adaptability of software programming and hardware components In a few areas, they are the best way to accomplish the required, constant execution without manufacturing specially coordinated Integrated Circuits (IC). The IC's usefulness will be fixed and updated amid their operational IC lifecycle for a specific task required. The field of reprogrammable computing device architectures provides a guide to excel information to the body of knowledge Collected in computing architecture and technology, run-time reconfiguration, tools, and applications. The essential focal points of run-time reconfiguration in gadgets are lessened power utilization, equipment reuse, and adaptability.

Keywords: *Run-Time Reconfiguration (RTR), Field programmable gateway array FPGA, Network on Chip, Reprogrammable Computing RC, programmable active memories PAM.*

1. Introduction

The Silicon industry is more with Moore's law which deals with the density of a processor or IC doubling every year. Processor design space was ruled by common bus architecture for years from the start of processor manufacturing. The ability of an FPGA to modify operation during runtime correct FPGA selection is very vital. The essential favourable circumstances of run-time reconfiguration in devices are diminished power utilization, hardware reuse, and flexibility.

The main problem is the speed of reconfiguration. Further, the technology in processor fabrication improved wherein the heat produced by the processor was not able to withstand the fabrication techniques. So, we had multi-core processors coming into the picture. But again, the problem of heat and other factors stopped the fabrication of these processors. To overcome this problem and to get higher performance, concepts of networks were brought into silicon wherein the processing nodes were put in and the nodes were interconnected to form a network, and messages (packets) were passed/routed between the nodes for processing.

Reconfigurable Computing (RC) performing computations with programmable architectures, like FPGAs, inherited a wide range of computing ideas from various departments including customizing the design for hardware, Digital Signal Processing (DSP), multiple computing processors, and Computer-Aided Design (CAD). In this review, we provide an accumulated body of information for reprogrammable architecture devices. As we begin with surveying and exploring new design architectures for Reconfigurable FPGA machines this paper reviews the different and many approaches to program FPGAs using different languages and computation models. Then the requirement on the study of the developments in tools and technology is essential to design RC should be automated and optimized. different tools and representations for Run-Time Reconfiguration (RTR) focus on the important application domains [1].

In this era, there is a need to build and design powerful super-processing Computing devices. Those computing devices will consist of multi-cores that communicate via the interconnection network. The central component on what factor the computer processing performance reflects the connection and communication of the cores. As the various intellectual property blocks are built to run at maximum capacity, the interconnection network they employ determines how well they perform as a whole. this made attempts to study various topologies and to achieve the efficient performance of the computing system. Since the topology is efficient, It will not function effectively

[1]*Assistant Professor, Department of Computer Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore, India*
[2]*Assistant Professor, Department of Information Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India*
[3]*Assistant Professor, Department of Artificial Intelligence and Data Science, M S Ramaiah Institute of Technology, Bangalore, India*
[4]*Research Scholar, Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India and affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India*
[5]*Undergraduate Student, Department of Artificial Intelligence and Machine Learning, M S Ramaiah Institute of Technology, Bangalore, India*
[6]*Professor and Head, Department of Artificial Intelligence and Machine Learning, M S Ramaiah Institute of Technology, Bangalore, India*
satish.eg@nmit.ac.in[1], rajeshwari.sb@msrit.edu[2], vinay.tr@msrit.edu[3], asma.rayan34@gmail.com[4], 1ms22ai022@msrit.edu[5]
*Corresponding Author jagadish.k@msrit.edu[*6]*

unless the routing algorithm is effective. and practical [2]. Hence, it requires to know the topology, Routing Architecture, flow control, and routing algorithm.

- Increasing complexity of SoC Architectures
- Heterogeneous Cores
- Requirement of high bandwidth, scalability, and power efficiency
- Traditional Bus architecture is no longer viable.

2. Architecture with Technology

First of all, In the late 1980s it was still conceivable to give equipment separation by collecting bundled coordinated circuits in various routes at the printed integrated circuit chip, board-level chips, The chips developed are limited and the speed and power of the chips are expanded, the cost of getting to capacities on-a-chip developed too. The need for expanding advantage to incorporating numerous functionalities on-the chip. Open doors for board-level separation diminished, expanding in the interest for customization design and on-chip functionality differentiation [3]. FPGAs gave an approach to get different chip usefulness separation without utilizing the custom VLSI manufacture. The test then is in what capacity would it be a good idea for us to calculation can be composed and customization on the FPGA? In what manner to select the specific working of FPGA is fused into a framework? In what capacity can the calculation exploit the FPGA abilities? As FPGAs develop in limit and go up against bigger parts in processing and the challenge for the specialized functioning FPGA is adopted in computing systems. To make the best usage of FPGA capabilities and computation As FPGAs develop in limit and go up against bigger jobs in processing frameworks than their underlying paste rationale specialty, by what method ought to FPGAs advance to help computing, communication, and integration tasks?

FPGAs give the extra chance to bolster the dynamic movement of various errands from programming to equipment amid execution. An autonomous approach is produced in the basic leadership of equipment/programming partitioned on FPGA with powerfully made and the equipment circuits are required. The measure of time needed for amalgamation and basic physical plan restricted the extent of the distinguished undertakings to on-board circuits. FPGAs that can precisely decide the cycle include delicate processors actualized FPGA log at run-time. Field-programmable door clusters are the cutting-edge innovation for designing the best programmable FPGA which interconnects and permits use in the desired applications to be utilized in a wide range of utilizations [4].

2.1 Pioneers

Only a couple of years after the presentation of FPGAs, numerous spearheading endeavours exhibited the effective advantages of reconfigurable FPGA. Scatter few FPGAs with a straight cluster and computing with the supercomputers on a DNA grouping coordinating the issue for a given framework where its risk and cost which is practically identical to technical specific FPGA Applications [5]. PAM-Programmable active memories are distributed FPGAs into a grid and exhibited superior on an accumulation of utilizations in the flag and picture handling, cryptography, logical processing, neural systems, and greater data transfer capacity in image processing [6]. The programmable active memories having high usage of public-key and private-key encryption and decoding the speeds in every stage includes surpassing the execution time and increasing in overall throughput of the customized chips.

2.2 Design Flow

Design entry can be done by using various techniques like Hardware Description Language and schematic based and the combination of both. Design and designers can choose any method. the planning is complicated than the designers algorithmically adopting the better design then HDL is the more sensible opinion. HDLs models a tier of abstraction level that may keep apart the designers from the main points of the hardware deployment. Schematic-based mostly entry offers designers way more convenience into the hardware. it's the better alternative for people who square measure hardware familiarized.

2.3 Synthesis

The process that interprets VHDL or Verilog code into a tool netlist format for the planning. The synthesis approach can check code syntax and evaluate the planning hierarchy, ensuring that the planning is optimum for the planning architecture chosen by the designer.

2.4 Implementation

The Implementation process follows with: Translate, Map, and Place and Route, as in Figure 1. All of the input netlists and restrictions are combined into a logic-style file via the translate technique. This information is recorded in an NGD file (Native Generic Database). This will be accomplished using the NGD Build software. Process restrictions aren't relevant here; instead, assigning the ports inside the style to the physical components (such as pins, switches, and buttons) of the desired device and establishing the planning's time requirements are. Map method The mapping approach separates the complete circuit with logical components into sub-blocks that can be worked into the logic blocks of an FPGA. Place and Route The place and route technique joins the logic blocks and arranges the sub-

blocks from the map method into logic blocks by the restrictions. Static timing analysis Once the MAP or PAR processes have been completed, a static time analysis is performed. The signal route delays of the look obtained from the looking logic are given in the post-MAP timing report. Place and Route temporal arrangement reports should be filed with the temporal arrangement delay information [7].

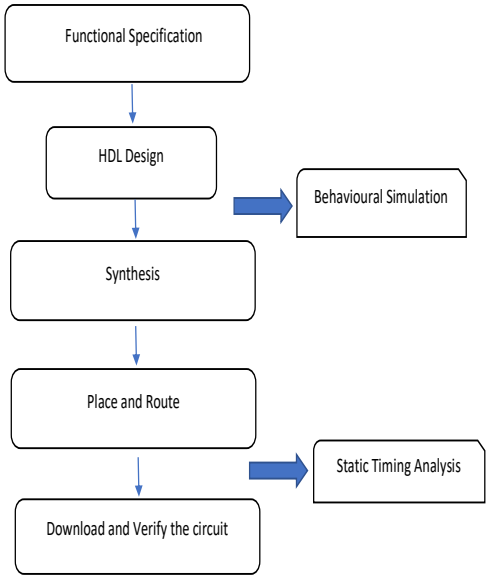

Fig 1: FPGA Flow Architecture

2.5 Particle Accelerators

The main thought at the earliest starting point was that FPGAs act as nonspecific, programmable equipment quickening agents for universally useful PCs. various point units were notable and fruitful at quickening with many specified FPGA applications. Numerous applications may profit on their own. The question is in what manner would it be a good idea for them to be interfaced with the universally useful CPU, and in what manner ought the RC be overseen? At the point when many co-processors are differentiated with chips, Crystal demonstrated how an FPGA that includes the outside co-processor could calculate the bit-level operations very faster comparatively more efficiently than CPUs. As chip configurations are more developed, the research gave new dimensions for FPGAs architectures with chip integrated for reconfigurable d to the model for high-speed computing processors. As it's evident that the processor would turn into a drawback on the off chance that it needed to intercede the development of information receiving and sending in the reconfigurable cluster. Reconfigurable FPGA demonstrated to coordinate every cluster as a different co-processor and require to exhibit the guide access to the memory framework. Multiprocessors may likewise profit by joined quickening agents. Cray coordinated FPGAs offer a parallel supercomputer with FPGA quickening agents and Pass on PC and as a supercomputer with an FPGA speeding up in its operations [8].

2.6 Fabric

FPGAs with the processing units make distinctive requirements with open doors in streamlining rather than utilizing FPGAs for paste rationale or universally useful enlist exchange level execution motors. Along these lines, there is a basic gathering of work researching how reconfigurable displays may progress past standard FPGAs, For the headway of FPGAs for standard utilize, includes the much faster development for making use in specific applications.

3. Computing Specifications and its Representation

From the beginning of Reprogrammable computing, reprogramming was broadly useful in PCs and FPGAs which merge into two altogether different platforms. as structure-oriented programming languages like C were, for the most pa implemented in targeting the microprocessors, and in most FPGA applications architects loads with schematics and composing of Boolean conditions. Hardware Description Languages (HDLs) for example, VHDL and Verilog, were picking up a balance, however HDL union at the time analysis required to create plans with bigger and slower than hand-made outlines. When all is said in done, accomplishing sensible outline execution utilizing the constrained rationale and directing assets in many gadgets required hand-tuning and comprehension of the essential FPGA design architecture.

For reconfigurable devices are open to many applications, with the latest programming environments and the calculation logic is required. the monstrous parallelism accessible is obliged. creators to take up the concise methods for communicating and abusing enormous parallelism much sooner than these issues were standard in the universally useful figuring group. Regardless of every design offering its particular advantages, various structural inquiries stay uncertain for non-concurrent FPGAs. Numerous models depend on rationale logic like those utilized for synchronous outlines and, therefore, the equivalent structural issues, for example, LUT estimate, bunch measure, and directing topology must be examined. Notwithstanding those inquiries, offbeat FPGAs additionally include the test of deciding the proper synchronization approach [9].

4. Applications

As the pioneers demonstrated that reconfigurable computing was useful in many specific applications and its usage is easily adaptable and gives the best user experience

has summed up and expanded this comprehension. Application-centred research demonstrates to us the possible sorts of advantages of reconfigurable designs.it describes the scope to individuals comprehend a variety of classes of utilizations and many specific applications with high qualities are probably going to profit from RC. These applications demonstrate to us industry standards to take care of issues on RC. Frequently the best plan and calculation for a reconfigurable design are altogether vary from partners for processor engineering. A hefty portion of these arrangements indicates the way abusing the monstrous principles of parallelism programming that Moore's Law applied to keep on making accessible to us. named Run-Time Reconfiguration (RTR). It completely misuses the crossbreed nature of Field Programmable Gateway Arrays, taking into consideration equipment association changes as required amid various periods of the calculation. Subsequently, RTR makes a particular prerequisite for many applications matchings to organize these many progressions. The advantage of the approach in the calculation to the close immediate requirements for the specified application, diminishing the area and vitality required for the plan. as many advantages should be calculated and weighed against the extra specifications required to meet and additional setup data and the time and vitality expected in exchange of data.

In many wide applications value and common sense utilization of run-time reconfiguration-RTR remains a challenge in many applications. For a few applications, it is conceivable to pick up the upside of customization to the requirement for run-time rationale alteration just by reinventing the information recollections or ceasing calculation and statically stacking one of the various precomputed arrangements. Furthermore, as depicted in this segment, the appropriateness of RTR requires extra plan gathering steps and run-time execution of the executives. Distinctive applications that are appropriate for RTR regularly require unmistakably characterized squares of calculation that can be powerfully adjusted amid execution [10].

The utilization of asynchronous design components will improve the general execution of FPGA engineering is the utilization of offbeat plan components. Expectedly, advanced circuits are intended for synchronous tasks and thusly FPGA designs have concentrated principally on executing synchronous circuits. asynchronous structures are proposed to improve the vitality productivity of asynchronous FPGAs since offbeat plans offer conceivably lower vitality as vitality is expended just when vital. Likewise, the offbeat structures can rearrange the plan procedure as per user requirements. Bit-level control is upheld productively on FPGAs, and mass encryption is frequently agreeable to gushing information through an exceptionally pipelined Data path.

Encryption/Decryption Acceleration - Private Key: In the previous papers described the advantages of reconfigurable implementations of data secure algorithms like DES- the Data Encryption Standard. Overcoming the configurability in FPGA -field programmable gateway array, it is conceivable to practice the Encryption/Decryption Acceleration DES circuit around a specific data timetable to limit the zone to produce high throughput [11]. The encrypted throughput with a reprogrammable and reconfigurable FPGA was implemented It was later demonstrated that committed Digital signal channels and memory obstructs in present-day FPGAs could be misused to accomplish Encryption/Decryption Acceleration AES encryption throughput is achieved. Others have demonstrated elite on IDEA.

Encryption Breaking: The efficient throughput and high performance of FPGAs on normal tasks make them productive with high usage and are necessary to search the encryption attack. FPGAs were utilized for a key hunt in the protected attachments secured layer, secure shell, and wired proportionate protection. Exceedingly effective FPGA -PAM equipment executions of strainer activities will figure up with the substantial counts were created The Workshop on Embedded Systems and Cryptographic Hardware gathers an abundance of extra work on executing cryptographic capacities on FPGAs.

4.1 Pattern-Task Matching

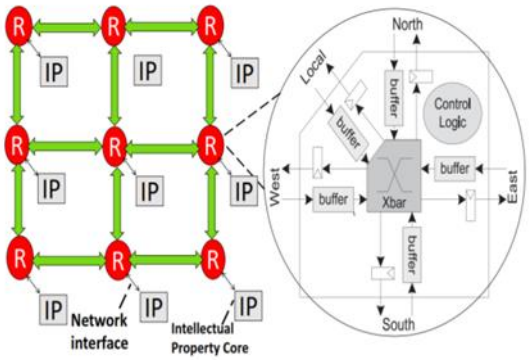

Fig 2: Network on Chip Architecture

Tasks-pattern matching, as it is very important to distinguish particular patterns within a heterogeneous huge data set, it takes more time in computation, but the computational tasks are very often to exploit the normal qualities of FPGAs. Coordinating the various applications that bolster the specific target acknowledgment inside pictures, bundle sifting in system information streams, and natural arrangement recognizable proof in substantial genomic databases have all ended up being good application bases for reprogrammable computing. As of recently in this presentation, a step-by-step outline of complex chip design, directing engineering, and programming stream of FPGAs is exhibited.

Table 1: NOC protocol Level and their Functionalities

NOC Protocol Level	Functionality
Application Layer	Application and Operating system of NOC
Presentation Layer	Traffic Characterization
Transport Layer	Between cores packet transmission
Network Layer	Packet Routing
Data Link Layer	Flits Flow Control and Transmission
Physical Layer	Phits Transmission

5. Network on Chip Architecture with Routing Techniques

Many topologies are available for NOC architecture to communicate information from the IP cores to the router through the network interface. different interconnection networks like mesh, torus, butterfly, ring, etc for the implementation of the network on a chip the Researchers have come up with all these NOC topologies. CDMA (Code Division Multiple Access) principles can be adopted by implementing the star-based Network on Chip [5].

In NOC routing architecture, packet-switched networks are used. This has led to the development of efficient principles for designing routing bridges [7]. Assuming the mesh topology routers have 4 inputs and 4 outputs from/to each other. Routers enable a variety of functions - from simple switching to intelligent routing. Because embedded systems are typically small and power-constrained, but still require high data rates, high-bandwidth routers are necessary to design with hardware usage in mind.

The NOC architecture consists of three main parts: the routers, link, and network interface architecture are the vital parts of the network on the chip as in Figure 2. Depending on the routing strategy, the router coordinates data packet traversal from the source port to the destination port. The router consists of four input and output ports with one local port. The network interface, also known as a network adaptor, acts as a link between the router and the devices connected to it. According to the switching approaches, it performs two operations: protocol conversion and data packet creation. The link provides a data transmission channel between the network's various devices [12]. The router is a critical component in NoC architecture, as it has a significant impact on NoC performance. Buffer depth, routing topology, routing algorithm, and arbitration all affect router performance Network Interface (NI) is the backbone of NoC since it connects all of the NoC routers and also connects each NoC router to the regional IP core. The routing algorithm, on the other hand, is a crucial component of NoC since it creates a schedule for sending each packet to its intended destination. Designing the router, Network interface, and routing algorithms, as well as performance, are some of the appropriate issues faced in NoC. of an interconnection network depends on three key components, and they are as follows:

- Topology
- Routing Algorithm
- Flow Control mechanism

NOC shown in Figure 2, consists of an array of routers each connected to a processing element (PE) or IP Cores. A processing element or core can be an analog or digital processor, field-programmable gate array (FPGA), memory, and another subsystem for processing multimedia and other applications. With the help of a local network interface, each element is connected to its router. The routers, in turn, are connected through a bidirectional physical link. The application that needs access, transfers its data to its connecting router, which, in turn, transmits the data packets to another router on the destination path. The router reads the destination address from the header file of incoming packets in selecting the next possible route. The destination router using the network interface delivers this data to the core connected to it. The same procedure of transmission is followed back by the destination node to the source core. Thus, the network on a chip has used the concept of packet-switched networks to provide efficient communication between the cores on a chip [13].

Further, similar to computer networking, the on-chip router of NOC is built on the open system interconnection (OSI) model where specific functions of each layer are well defined as shown in Table 1 Various protocol layers of NOC model 4 shows the use of packet switching in NOC, a comparison is also shown in Table 2 between NOCs and general computer [11]. Table 2 Similarities and differences between Computer Network and NOCs.

5.1 Topology

Topology describes the process of establishment of connections between various nodes with Interconnection networks which provides the communication links. The direct and indirect topologies are the two significant classifications used in interconnection networks. The nodes in a direct topology are directly connected to their immediate neighbours, A layer of switches exists between the nodes in an indirect topology [14].

5.2 Routing Algorithm

The topology describes the physical paths that connect two nodes, and the topology can include multiple physical paths connecting nodes. The routing algorithm specifies

the path that a packet should take from its source to its destination. The routing algorithm is responsible for connectivity, adaptivity, deadlock, live lock, and fault tolerance in interconnection networks [2]. Routing algorithms de on the path. The routing algorithms are classified based on the path chosen by the routing algorithms as bellow:

5.3 Deterministic Routing Algorithm

In a deterministic routing algorithm, the router chooses a fixed path to a specific destination, regardless of network circumstances. Deterministic routing algorithms are simple to develop and do not cause deadlock, but they will not improve performance due to a lack of homogeneity in load distribution across the network's links. [1][3].

5.4 Oblivious Routing Algorithm

The oblivious routing techniques are unaffected by the network's present state. The deterministic routing algorithms are a subset of the oblivious routing algorithms, but the oblivious routing algorithm ignores traffic in favour of balancing network load by examining traffic to any random node and routing it to the destination node. It may also include more hops in comparison to the shortest path between the source and destination [1][4].

5.5 Adaptive Routing Algorithm

The adaptive routing algorithms are dependent upon the state of the network, in which the network performance will be considerably increased. In basic, the adaptive routing algorithm does not follow a single path like the next-hop but returns multiple paths [8]. The popular routing algorithms are the IX/Y Routing algorithm and the Odd-even routing algorithm.

The header information stored in the routing table in the routing algorithm plays a significant part in the process implementation. They are further classified into two categories:

5.6 Table-based Routing Algorithm

The path from the source node to the destination nodes via intermediate nodes information is stored in a table. The cost and scalability of the routing depending on the storage of the route memories are proportional to the total number of nodes interconnected in the network [9].

5.7 Finite State-based Routing Algorithm (FSB)

The FSB routing algorithm attempts to lower the memory consumption, by routing a packet from the source node to identified destination node based on logic or mathematical formulation [10].

5.8 Flow Control Mechanism

The Flow Control mechanism plays an important role in the allocation of the virtual channels to the packet, traversing in the network. The flow control aids in achieving the minimum latency in delivering the packets with a gain of the maximum throughput. The most widely used Techniques and methods in the flow control mechanism are [15]:

- Buffer less Flow Control Method: In this buffer less flow control mechanism, If the desired channel is not available at the time, the packet may be misrouted or dropped [11].
- ON/OFF Flow Control: It is a buffer-based approach. Under this method, the receiving node will send a stop signal if a lack of space arises, and Only after getting the continue signal will the flow begin again [1].
- Credit-based flow control: Also, a buffer-based approach; the credit-based flow system sender is limited to sending flits only in the occupancy of credits from the receiver side based upon the buffer allocated to the sender to control the buffer overflow. The number of flits sent in a particular The credit available with the router delivering the flits will determine the direction. [2].

Techniques for switching are another essential component of NoC design. The switching mechanism determines the data flow in the network through routers. Below is a description of the various kinds of switching procedures that are employed.

Circuit Switching: Before data transmission, in circuit switching, the physical path that connects the source and destination nodes is set aside. There is very little likelihood of packet loss because there is a dedicated path. But occasionally, the path set aside for one connection is blocked from being utilized by another until it is released, resulting in the inefficient use of bandwidth [16].

Packet Switching: Using the packet switching technique, information messages are divided into packets at the source node and then sent separately via several routers from the source to the destination node. Depending on the network's current state, each packet travels a different path. Because it may facilitate simultaneous data exchange between a large number of source-destination pairs, packet switching is the most widely used approach in network observability. Packet switching is classified as [17]:

Wormhole Switching: The packets are divided into smaller flow control units of a defined length. in wormhole switching (flits). The header flit in this switching approach contains the data needed to choose a path from source to destination. Only the header flit requires some time to decide on a path, while the subsequent flits in the same packet simply follow the header flit's path, lowering average message latency dramatically. The buffer size needed at the router is lowered to the size of the flit rather than the size of the packet since the packets are split up

into smaller size flits. Wormhole switching's fundamental disadvantage is that when the header flit is blocked during transmission, the entire packet is blocked. Because of the interdependencies between the links, this switching is more prone to a deadlock [18].

Store and Forward Switching: Before being forwarded to the following node in the path, the entire packet is stored at the node using in-store and forward switching. A buffer big enough to hold the entire packet should be present on every node along the target path. With the space and power constraints of on-chip networks, store and forward switching adds a lot of overhead and reduces performance [19].

Virtual Cut-through Switching: The data in the form of packets is forwarded in Virtual Cut-through switching, the size of the required buffer at the router is the same as the packet size. Before the complete packet is received at the current router, the packets are able to pass through to the subsequent router's input. only in the event that the subsequent downstream router has sufficient storage to hold every packet it receives, significantly lowering latency compared to the store and forward switching strategy. As soon as the router receives the header and data bytes, it can begin forwarding them. it receives, significantly cutting latency compared to the store and forward switching approach. When there is a heavy network load, the Virtual Cut-through switching operates similarly to the store and forward switching technique [20][21].

6. Conclusion

In a Computing system, the reprogrammable computing RC plays a vital role and best alternative to ASICs and fixed microprocessors. With the increase in the utilization of energy and to improve the reliability concerns, the core of computing infrastructure will include heterogeneous fabrics of complex chips and accelerators having reconfigurable capacity. RC has turned into a feasible other option to ASICs and settled microchips in designing the frameworks. With expanding vitality and dependability worries, there is each motivation to trust its significance will develop as innovation scales. As we go on to bigger or complex with heterogeneous chips textures and quickening agents, reconfigurable computing devices will keep on merging into the standard figuring framework

The field has collected more proof of the energy efficiency and effective performance of Field-programmable gateway arrays are compared to processors with a large range and variety of applications implemented. Many critical walks in the catch of utilizations, C-to-gates compilation, in-stream representation with Designs of FPGAs were the main gateways that can be designed by the client to actualize the requirements that the individual needs. In this connection, a chip is an improved CPU or Central Processing Unit

There is still much to learn and far to go before the field starts to develop. Usability, simplicity of-investigate, availability, and a moderate alter incorporate troubleshoot cycle remain challenges that must be tended to accomplish more extensive interest RTR and Hybrid architectures, while alluring, require better deliberation models and support. Exhibits and examples of power usage for complex applications on a huge scale are still required, just like a superior comprehension with similar qualities and shortcomings of the reprogrammable computing devices. Developments in every aspect of Reconfigurations are accounted for frequently. In this section, at first, a brief presentation of conventional rationale and directing architectures of FPGAs is exhibited. Later, extraordinary strides required in the FPGA configuration stream are itemized. At last different methodologies and technologies which have been efficiently and effectively utilized to lessen a few dis-preferences of FPGAs and ASICs, without bargaining their significant advantages are depicted.

In today's technology, maximizing the usage of existing sources rather than adding new resources might make the Network on Chip an excellent answer for present applications. The NOC is primarily concerned with computation and communication, and it is very much in line with increasing design complexity and falling system productivity. NOC architectures and hardware-related challenges will be addressed by the researchers. Still, a clear strategy for modeling, building, and implementing NOC architecture is lacking. Some criteria that need to be discussed in further depth include the application and feasibility of NOC applications. We need to explore a low-cost, small-area, and power-efficient solution for NOC to be effective in the embedded systems sector.

References

[1] C. Ababei, E client, *Congestion-Oriented Custom Network-on-Chip Topology Synthesis*, in Proc. International Conf. Recon durable Computing and FPGAs (ReConFig), 2010, pp. 352{357.

[2] P. J. Burke et al., *Quantitative Theory of Nanowire and Nanotube Antenna Performance*, IEEE, Trans. Nanotechnology, vol. 5, no. 4, pp. 314{334, July 2006.

[3] N. Concer, A. Vesco, R. Scopigno, and L. P. Carloni, A Dynamic and Distributed TDM Slot-Scheduling Protocol for QoS-Oriented Networks-on-Chip, in *Proc. IEEE 29th International Conf. Computer Design (ICCD)*, 2011, pp. 31{38.

[4] A. Bouhraoua, O. Diraneyya, and M. E. Elrabaa, A Simple ed Router Architecture for the Modi ed Fat Tree

Network-on-Chip Topology, in *Proc. NORCHIP*, 2009, pp. 14.

[5] K. Chang, S. Deb, A. Ganguly, X. Yu, S. P. Sah, P. P. Pande, B. Belzer, and D. Heo, Performance Evaluation and Design Trades for Wireless Network-on-Chip Architectures, in *Accepted for publication in ACM Journal on Emerging Technologies in Computing Systems*, Sept. 2011.

[6] M. F. Chang et al., CMP Network-on-Chip Overlaid with Multi-Band RF-Interconnect, in *Proc. IEEE International Symp. High-Performance Computer Architecture (HPCA)*, Feb. 2008, pp. 191{202

[7] G. Stitt, B. Grattan, J. Villarreal, and F. Vahid, Using on-chip configurable logic to reduce embedded system software energy, in *Proc. FCCM, 2002, pp. 143–151.*

[8] E.G. Satish, Sanju V, Srinivasa N, Navya Y.U, "A Research Review on the wireless network on chips", Vol 9, Special Issue, International Journal of Computer Engineering and Applications, ISSN 2321-3469, Sept 2015.

[9] E.G Satish, Ramachandra A C, Faulty Node Detection and Correction of Route in Network-on-Chip(NoC) in 3rd International conference on Innovative Data Communication Technologies and Application (Springer Lecture Notes –Indexed by Scopus,Inspec,EI Compendex. Innovative Data Communication Technologies and Application. Chapter 57 DOI : 10.1007/978-981-16-7167-8

[10] E.G Satish, Ramachandra A C, *Design of an Efficient Traffic Balance and Fault Tolerant Routing Algorithm for Network on Chip*, Design Engineering, ISSN: 0011-9342,1485-1500. http://www.thedesignengineering.com/index.php/DE/article/view/2148.2021

[11] E.G Satish, Ramachandra A C, A Comparative Performance Analysis of Routing Topology for NOC Architecture, in Sixth International Conference on Emerging Research in Computing, Information, Communication and Applications ERCICA - 2020.(Scopus Indexed). chapter 34 DOI : 10.1007/978-981-16-1342-5

[12] S. Trimberger, Three ages of FPGAs, Proc. IEEE, vol. 103, no. 3, pp. 318–331, Mar. 2015.

[13] J. Williams et al., Characterization of fixed and reconfigurable multi-core devices for application acceleration, ACM Trans. Reconfig. Tech. Syst., vol. 3, no. 4, 19:1–19:29, Nov. 2010.

[14] Y. Chang, C. Chiu, S. Lin, and C. Liu, On the Design and Analysis of Fault-Tolerant NoC Architecture Using Spare Routers, in *Proc. 16th Asia and South Paci c Design Automation Conference (ASP-DAC)*, 2011, pp. 431{436.

[15] L. Benini and G. DeMicheli, Networks on Chips: A New SoC Paradigm, *Computer*, vol. 35, no. 1, pp. 70{78, Jan. 2002.

[16] J. M. Arnold, S5: The architecture and development flow of a software configurable processor, in Proc. ICFPT, Dec. 2005, 125–128.

[17] C. Ababei, E client Congestion-Oriented Custom Network-on-Chip Topology Synthesis, in *Proc. International Conf. Recon durable Computing and FPGAs (ReConFig)*, 2010, pp. 352{357.

[18] N. Bagherzadeh and M. Matsuura, Performance Impact of Task-to-Task Communication Protocol in Network-on-Chip, in *Proc. Fifth International Conf. Information Technology: New Generations*, 2008, pp. 1101{1106.

[19] https://allaboutfpga.com/fpga-design-flow/

[20] Agarwal, A, Raton, B, Iskander, C, Multi systems, H & Shankar, R 2009, Survey of Network on Chip (NoC) architectures &contributions, Journal of Engineering, Computing, and Architecture, vol. 3, no.1, pp. 21-27.

[21] Agyeman, MO, Wan, XJ, Vien, QT, Zong, W, Yakovlev, A, Tong, K,& Mak, T 2015, *On the design of reliable hybrid wired-wireless network-on-chip architectures*, Proceedings of International Symposium on Embedded Multicore/Many-core Systems-on-Chip, pp. 251-258.

Table 2: Computer Networks Vs NOC

Similarities	Differences
The architecture consists of router, links, and processor cores	Computer Networks are used for common applications while NOC is designed for a specific application
Header flits transfers routing information to the routing nodes	Low power design techniques are required to conserve energy
Communication protocols like routing, switching policies, flow control, and arbitration are the same	Heavy communication protocols are not supported
Based on Packet switching networks	Computer networks are based on inextensible routers whereas router topology is fixed for a particular design