*International Journal on Applications in Engineering and Technology*
*Volume 3: Issue 1: March 2017, pp 12 -16. www.aetsjournal.com* ISSN (Online) : 2455-0523

# A REGIONAL CONGESTION AWARENESS ON VIRTUAL CHANNEL FOR NETWORK ON CHIP ROUTERS

N.Jeevitha ,K.Bhuvaneshwari

***Abstract —*** The advancement technology in the on chip communication, requirement of faster interaction between devices is becoming vital. Hotspot congestion control is one of the most challenging issues when designing a high-throughput low-latency network on the chip (NOC). The implementation of the NoC on a separate layer offers an additional area that may be utilized to improve the network performance by increasing the number of virtual channels, buffers size, or mesh size. Network-on-chip is an embedded switching network which is used to interconnect Intellectual Property (IP) cores in SOCs. NOC will improve the latency and throughput. Experimental results show that increasing the number of virtual channels rather than the buffers size has a higher impact on network performance. The advent of deep sub-micron technology has recently highlighted the criticality of the on-chip interconnects. It is mainly focused on the router design parameters on both system level including traffic pattern, network topology and routing algorithm, and architecture level including arbitration algorithm and buffer mechanism. The parallel router is embedded on a single chip and the network on chip is executed on FPGA to analyze the utilization resources such as area and power.

***Keywords*** — : Network On Chip (NOC), System on Chip (SOC), Intellectual Property, Multiprocessor System on Chip (MPSoc) FPGA.

## I. INTRODUCTION

Continuing Moore's law to 32nm technology and beyond encourages researchers to design scalable architecture with multiple computation cores, also called processing elements (PE), on a single chip. As the architectural focus shifts from monolithic, computation centric designs to multi-core, communication-centric systems, communication power has become comparable to logic and memory power, and is expected to eventually surpass them [1]. Therefore, when area and performance are more important, application specific heterogeneous irregular networks (see Figure 1(b)) are preferred. However, the design of these networks is more difficult and specialized routing algorithms are necessary to prevent deadlock.

In this paper, we introduce a NoC design process and architecture for FPGAs. A distinctive feature of FPGA systems is that they include a combination of *hard* and *soft* functionalities. The hard functionality is implemented in silicon; it typically includes special purpose modules like processors, multipliers, external network and memory interfaces, etc. The soft functionality is configured using programmable elements (gate arrays, flip-flops, etc.) and routing components [2].

Modern FPGAs contain hundreds of thousands of Programmable elements, in addition to special purpose modules. As technology scales, the sheer number of logic units will render a flat FPGA chip design unmanageable. FPGA engineers are already experiencing unacceptable place-and-route times for large designs with tight timing constraints. To remedy this problem, modern FPGA CAD tools, like Xilinx's Plan Ahead, are already supporting a certain degree of hierarchical design: they allow designers to implement independent modules on the chip, which are later connected [3]. We thus envision a future FPGA that is organized hierarchically, whereby the chip is divided into high-level regions (some programmable and some hard), interconnected by a NoC.

When architecting an FPGA NoC, has to decide which functionalities are implemented as hard cores and which are left as soft. There is a tradeoff between the flexibility offered by the soft part and the higher performance offered by hard part. Since inter-module communication is often a bottleneck, it is important design the NoC architecture for high performance. We therefore advocate laying out the network infrastructure, including metal wires and hard-coded routers in silicon [4]. At the same time, in order to allow for maximum flexibility, the NoC infrastructure should be able to accommodate multiple routing schemes and a large

N.Jeevitha , PG Scholar , Electronics and Communication Engineering , IFET College of Engineering , Villupuram.
(Email : jee.nagarajan@gmail.com )
K.Bhuvaneshwari , Assistant Professor , Electronics and Communication Engineering , IFET College of Engineering , Villupuram.

variety of traffic patterns. To this end, we allow network interfaces to be soft. Simplistic routing schemes, like XY, can employ small interfaces, whereas more elaborate source-routing schemes may have the interfaces store large routing tables. Our novel architecture is detailed in Section 2.

The main challenge is exploiting network sources efficiently, i.e., supporting a large number of program designs while investing minimal resources (wires and logic). In this context, there is an inter-play between the link capacity requirements and the routing scheme used to route packets between modules [5,6]. A routing scheme that balances the load over all links readily supports more designs using smaller link capacities than an unbalanced one.

Section 3 formally defines FPGA routing (on our suggested architecture) as an optimization problem. In order to study the inter-play between routing and capacity requirements, we define a new concept called *design envelope*, capturing the required capacity for a collection of traffic patterns. The more traffic patterns the envelope accommodates, the more flexibility is offered to the designer configuring the chip. We study the design envelopes required to accommodate a collection of patterns with each of the routing schemes. In Section 4, we present efficient solutions to the FPGA routing problem. Note that traditional routing algorithms like XY lead to unbalanced capacity allocation [7], and are therefore not suitable for programmable chips. It is

possible to improve the balance by splitting the flow, toggling between sending on XY and YX routes approach *toggle XY (TXY)*. However, TXY is not optimal when traffic requirements are not symmetric, (which is very common in HW architectures). We improve it by adding *weights* to flow division (based on the design pattern), and call the resulting algorithm *weighted toggle XY* (WTXY).

## II. PROBLEM STATEMENT

The existing system consists of general router architecture of network-on-chip architecture with XY routing and matrix arbitration. The matrix arbitration takes more area in circuit switching. The EDVC router consists of five input-port modules, an arbiter, and a crossbar switch. While using general router there may cause loss of date, delay and power consumption is more. To decrease the buffer size arbitrarily to reclaim silicon area and minimize power consumption is not a viable solution, because of the intricate relationship between network performance and buffer resources [8-10].

Buffer size and management are directly linked to the flow control policy employed by the network; flow control, in turn, affects network performance and resource utilization.

### XY ROUTING ALGORITHM

Routing algorithm is used to rout the packets from source PE to destination PE. XY routing algorithm is used to rout the packet in proposed router designing. It is a type of kind of distributed deterministic routing algorithms. In 2-D mesh topology NoC, each router can be identified by its coordinate as shown in fig .1. The XY routing algorithm compares the current router address (Cx,Cy) to the destination router address (Dx,Dy) of the packet, stored in the header flit. Flits must be routed to the core port of the router when the (Cx,Cy) address of the current router is equal to the (Dx,Dy) address. If this is not the case, the Dx address is firstly compared to the Cx (horizontal) address. Flits will be routed to the East port when Cx<Dx, to West when Cx>Dx and if Cx=Dx the header flit is already horizontally aligned. If this last condition is true, the Dy (vertical) address is compared to the Cy address. Flits will be routed to South when Cy<Dy, to North when Cy>Dy [9]. If the chosen port is busy, the header flit as well as all subsequent flits of this packet will be blocked. The routing request for this packet will remain active until a connection is established in some future execution of the procedure in this router.

## III. PROPOSED SYSTEM

This project proposed that energy efficient virtual memory based First In First Out scheme to enhance the power efficiency of these NoC crossbars. More importantly these virtual memory circuits producing some delay to read and input signal. For eliminating the impedance based physical buffers presents in the traditional NoC system we are creating FIFO based Virtual NoC buffers which leads to the significant power reduction in the Multicore NoC system. This FIFO based Virtual buffers uses the FSM (Finite State Machine) Logic for maintaining all the input data without any losses [11]. But the major consideration was elimination of buffers will leads to the traffic in the router crossbar.

The Proposed system consists of newly designed parallel router which can support five simultaneous routing requests at the same time for implementing NOC on FPGA. If router packet [12] processing was Busy means the NoC system will Allows packets with Minimum Size and if NoC Router Packet processing Rate was high means it will allows the Packets with

Higher Size based on this algorithm this system will maintains the Minimum Traffic Rate and multicore NoC system.

For Eliminating the High power causing Multipliers presents in the Traditional NoC we are proposing the Xor based Mux and Demux circuits for reducing the overall power usage compare to the common multiplier based Mux and Demux our proposed Xor based Circuits having Lower power dissipation hence overall power usage of the NoC system will reduced.

## IV. FLOW PROCESS

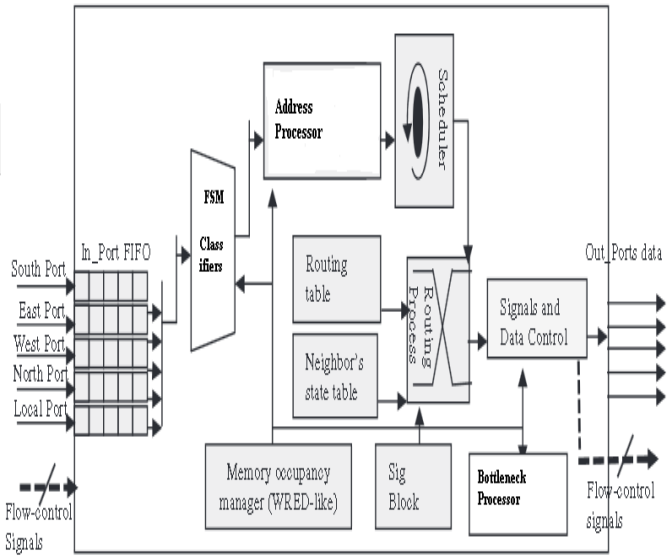



**Fig. Proposed System Architecture**

### *A. Buffer less Architecture*

The First in First out (FIFO) circuit [13] used to stores all the incoming asynchronous packets from the variable input cores. The overall capacity of the Network on Chip cores directly depends on the size of FIFO circuit. Basically the FIFO circuit independent of physical properties and its properties are maintained separately. Those FIFO based Architectures are used to overcome the power related problems produced by the traditional NoC systems.

### *B. Finite State Machine Classifiers*

The FSM classifiers are used to identifies the packet origin and responsible output port for the current packet and depends upon that it will add the address with current data. This address fields are used to transfer the packet to responsible output port without any losses. Later this address fields will be separate by the address Processors. Also it will maintain the virtual memory processor for maintaining the data without any losses. This virtual memory Architecture will helps to reduce the overall power usage of the Network on Chip Systems [14,15].

### *C. Address Processors*

The address processors will process the incoming packets and forwards all the packets to their destinations without any losses in the packets. While processing the packets it will check the address presents in the packets and remove the address fields from the packets and forward those packets to the corresponding output ports with the help of routers. Overall memory used by the inputs ports are managed and controlled by the memory occupancy manager.

### *D. Router Scheduler*

The Router scheduler uses the Bottle neck algorithm for overcoming the traffic created by the buffer less architecture. Here we are applying the weight based algorithm for creating different weight for the router state and depends upon that we are applying the packets with the different sizes. This scheduling process also helps to reduce the overall complexity of the system [16].

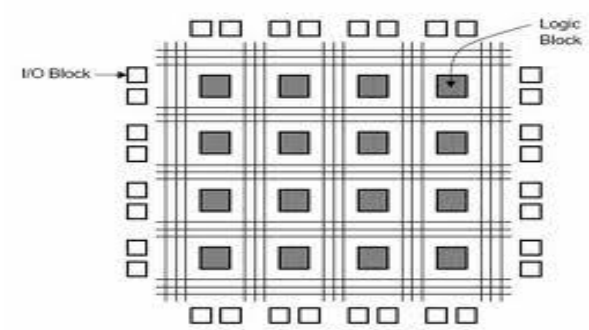
### *E.FIELD PROGRAMMABLE GATE ARRAY*

A field-programmable gate array (FPGA) is an integrated circuit designed to be configured by the customer or designer after manufacturing hence "field-programmable". A field-programmable gate array (FPGA) is a semiconductor device that can be configured by the customer or designer after manufacturing-hence the name "field-programmable". FPGA’s contain programmable logic components called "logic blocks", and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together"-somewhat like a one-chip programmable breadboard. Contemporary FPGAs have large resources of logic gates and RAM blocks to implement complex digital computations. As FPGA designs employ very fast I/Os and bidirectional data buses it becomes a challenge to verify correct timing of valid data within setup time and hold time [1720].

### *F.FPGA*

Planning enables resources allocation within FPGA to meet these time constraints. A few "mixed signal FPGAs" have integrated peripheral analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) with analog signal conditioning blocks allowing them to operate as a system-on-a-chip. Such devices blur the line between an FPGA, which carries digital ones and zeros on its internal

programmable interconnect fabric, and field-programmable analog array (FPAA), which carries analog values on its internal programmable interconnect fabric.




## V. RESULT AND DISCUSSION

**Performane Discussion**

The below figure shows the schematic diagram for proposed system with five inputs and corresponding five outputs, including clock,reset and acknowledgement.

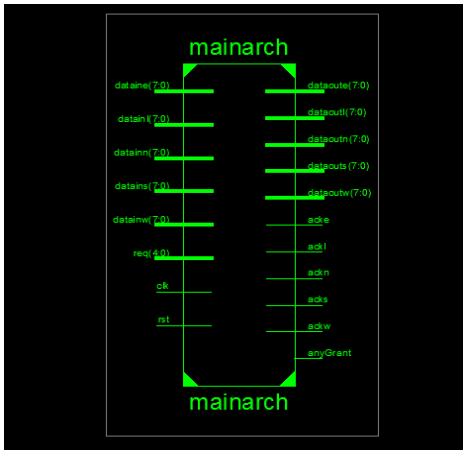



Fig. Schematic diagram

| Device Utilization Summary (estimated values) | | | |
|---|---|---|---|
| Logic Utilization | Used | Available | Utilization |
| Number of Slices | 1329 | 23872 | 5% |
| Number of Slice Flip Flops | 1579 | 47744 | 3% |
| Number of 4 input LUTs | 1065 | 47744 | 2% |
| Number of bonded IOBs | 92 | 469 | 19% |
| Number of GCLKs | 1 | 24 | 4% |

Fig. Design summary

The above figure shows the overall Area utilized by the proposed Buffer less logic based NoC cores. These results showing that the overall Area utilized by the proposed NoC was having very low area when compare to the existing systems.

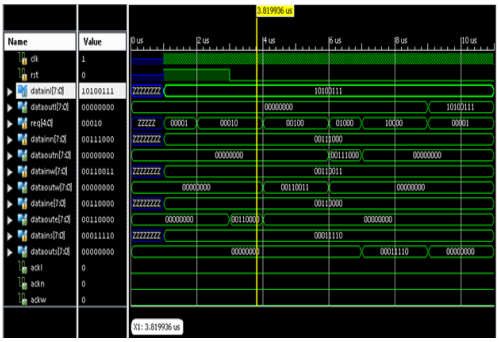


Fig. Simulation diagram

The proposed scheme was implemented and simulated using the Xilinx 14.2 tool and results achieved clearly showing that the overall power was reduced when compare to the conventional Network on Chip Architecture and also the proposed Buffer less NoC based Architecture having the best Power Efficiency ratio against the all possible at stages in communication processors.

| Method | Frequency(MHz) | Dynamic Power (mW) |
|---|---|---|
| Proposed Scheme | 1GHz | 11 |
| Conventional Scheme[1] | 1GHz | 30.96 |

And the below fig shows that the overall power usage of the proposed Buffers less NoC with scheduling algorithm based router. These results shows that power dissipation of the proposed scheme was 0.011W that is extremely reduced compare to previous conventional schemes.

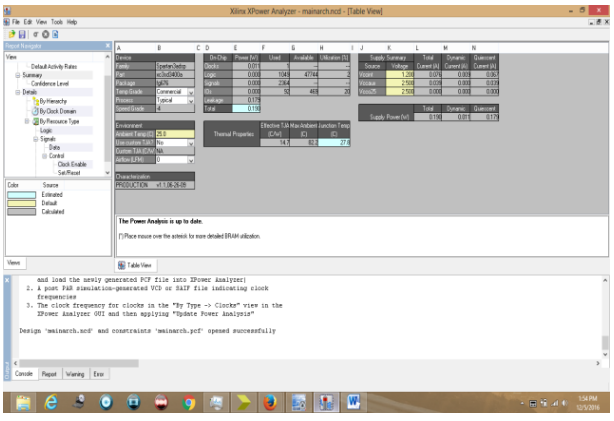


Fig. Power Utilization

## VI. CONCLUSION

In this paper network have implemented buffer less architecture for Multi core NoC system for power efficient communication in Multi core systems. Also we proposed that Improvised bottleneck algorithm for congestion avoidance in Network on Chip core. The simulation shows the functionality of NoC router. It is hard to do the comparison directly with other works, so the router area and the frequency of the proposed router are considered. This flow control schemes uses intelligent packet control scheme it allows different size of packets with respect to the different processing complexity of the router. Conventional crossbar switch was redesigned by XOR based MUX and DEMUX for better scaling in Power usage.

This three proposed schemes are applied in the Multicore NoC system and results showing that overall power usage of the system was reduced nearly 30% when compare to the conventional routing schemes. Also the proposed scheme having best throughput with negligible delay.

## FUTURE ENHANCEMENT

The future work is to identify the fault while processing. While processing the data parallelly some faults may occur, this paper may detect the fault and solve the fault. The process by which floating buffers are associated with output ports and the design of heterogeneous adaptive routers are the future works. Heterogeneous adaptive routers are the important aspect to account for heterogeneous processing elements and also asymmetric aspects of a NoC.

## REFERENCES

[1] Masoud Oveis-Gharan and Gul N. Khan," Efficient Dynamic Virtual Channel Organization and Architecture for NoC Systems," in in Ryerson University Toronto, Ontario M5B 2K3 Canada, Feb. 2016, Pages:465-478.

[2] Ahmed Aldammas.: The efficiency of buffer and buffer-less data-flow control schemes for congestion avoidance in Networks on Chip. In: Journal of King Saud University – Computer and Information Sciences, pp. 96–112 (2016).

[3] MasoudOveis-Gharan and Gul N. Khan," Index-based Round-Robin Arbiter for NoC Routers," in Ryerson University Toronto, Ontario M5B 2K3 Canada, May 2015, Pages:62-67.

[4] MasoudOveis-Gharan and Gul N. Khan,"Efficient Virtual Channel Organization and Congestion Avoidance in Multicore NoC Systems," in Ryerson University, 350 Victoria St, Toronto, ON M5B 2K3 Canada, July. 2014, Pages: 30-35.

[5] John Jose, BhawnaNayak, Kranthi Kumar, MadhuMutyam, "DeBAR: Deflection based adaptive router with minimal buffering", in Design Automation and Test in Europe Conference, pp.1583-88, March 2013.

[6] C. Nicopoulos, D. Park, J. Kim, N. Vijaykrishnan, M. Yousif , Chita R. Das, "ViChaR: A Dynamic Virtual Channel Regulator for Network-on-Chip Routers," in the Pennsylvania State University Park, PA 16802, USA, **SEP.2013,** Pages: 333 – 346.

[7] Chris Fallin, Greg Nazario, Xiangyao Yu, Kevin Chang, RachataAusavarungnirun, OnurMutlu, "MinBD: Minimally-Buffered Deflection Routing for Energy-Efficient Interconnect", in SixthIEEE/ACM International Symposium on Networks on Chip (NoC), pp.1- 10, May 2012.

[8] Jing Lin, Xiaola Lin, Liang Tang, "Making-a-stop: A new bufferless routing algorithm for on-chip network", Elsevier Publish Journal ofParallel and Distributed Computing, pp.515-524, 2012.

[9] M. Evripidou, C. Nicopoulos, V. Soteriou, and J. Kim, "Virtualizing virtual channels for increased network-on-chip robustness and upgradeability," in Proc. IEEE Comput. Soc. Annu. Symp.VLSI,Amherst,MA, USA, Aug. 2012, pp. 21–26.

[10] N. Alfaraj, J. Zhang, Y. Xu, and H. J. Chao, "HOPE: Hotspot congestion control for Clos network on chip," in *Proc. 5th IEEE/ACM NoCS*, Pittsburgh, PA, USA, May 2011, pp. 17–24.

[11] N. Alfaraj, J. Zhang, Y. Xu, and H. J. Chao, "HOPE: Hotspot congestion control for Clos network on chip," in Proc. 5th IEEE/ACM NoCS, Pittsburgh, PA, USA, May 2011, pp. 17–24.

[12] Rahmani, A.-M., Latif, K., Vaddina, K.R., Liljeberg, P., Plosila, J., Tenhunen, H.: Congestion aware, fault tolerant and thermally efficient inter-layer communication scheme for hybrid NoC-bus 3D architectures. In: Proceedings of the 5th ACM/IEEE International Symposium on Networks-on-Chip, pp. 65–2 (2011).

[13] G. Michelogiannakis, D. Sanchez, W. J. Dally and C. Kozyrakis, "Evaluating Bufferless Flow Control for On-Chip Networks," *NOCS*, 2010.

[14] Cota E, Morais Antony, SoaresLubaszewski, "Reliability, Availability and Serviceability of Networks- on- Chip," *Springer*, 2010.

[15] Y. Xu, B. Zhao, Y. Zhang, and J. Yang, "Simple virtual channel allocation for high throughput and high frequency on-chip routers," in Proc. IEEE 16th Int. Symp. High Perform. Comput. Archit., Bengaluru, India, Jan. 2010, pp. 1–11.

[16] T. Moscibroda and O. Mutlu."A case for bufferless routing in on-chip networks," *ISCA-36*, June 2009.

[17] Feero, B.S., Pande, P.P.: Networks-on-chip in a three-dimensional environment: a performance evaluation. IEEE Trans. Comput. **58**(1), 32–45 (2009).

[18] Carloni, L.P., Pande, P., Xie, Y.: Networks-on-chip in emerging interconnect paradigms: advantages and challenges. In: Proceedings of the 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip, pp. 93–102 (2009).

[19] L. Mingche, G. Lei, S. Wei, and W. Zhiying, "Escaping from blocking: A dynamic virtual channel for pipelined routers," in Proc. Int.Conf. Complex, Intell.,Softw. Intensive Syst., Barcelona, Spain, 2008,pp. 795–800.

[20] M. Lai, Z. Wang, L. Gao, H. Lu, and K. Dai, "A dynamically-allocated virtual channel architecture with congestion awareness for on-chip routers," in Proc. 45th ACM/IEEE DAC, Anaheim, CA, USA, Jun. 2008,pp. 630–633.