Dual Crossbar Router Architecture for Heterogeneous NoCs

Anjana R Krishnan[1] Revathi K[2]
[1]Student [2]Assistant Professor
[1,2]Department of Electronics and Communication Engineering
[1,2]SEA College of Engineering and Technology, Ekta Nagar, K.R.Puram, Bengaluru, India

Abstract— In this paper we discuss about a heterogeneous adaptable router to decrease the time in an irregular mesh Network on Chip (NoC). A suitable adaptive algorithm is used for routing. The high-performance steps such as throughput, & bandwidth is to be defined to design on time to make sure the performance of Network on Chip. The proposed Dual Crossbar router architecture should have high performance for heterogeneous NoC. A Heterogeneous Network on Chip has Converge Diverge Crossbar (CD-Xbar).

Keywords: SoC, NoC, CD-Xbar, Intellectual Property

I. INTRODUCTION

Gordon Moore Law states the transistor's integrated on an IC's doubles every eighteen months. Moore's also had stated that computers, machines that execute on PC would consume low power at faster speed. The Gordon E Moore was the co-founder of Intel. The law stated by Moore that the growth of micro-processors is exponential. The system on Chip (SoC) is an IC that takes a single platform and integrates the entire components in the PCB. The components that are incorporated within may include central processing unit (CPU), input and output ports, internal memory. The SoC performs a some functions such as signal processing, machine learning, artificial intelligence and wireless communication [1]. The integrated chips are not cost effective and takes less time to manufacture.

II. NETWORK ON CHIP

A NoC is an on chip internet technology which uses System on Chip designs for connecting blocks such as Intellectual Property. NoC is used to simplify the hardware required for routing and switching functions.

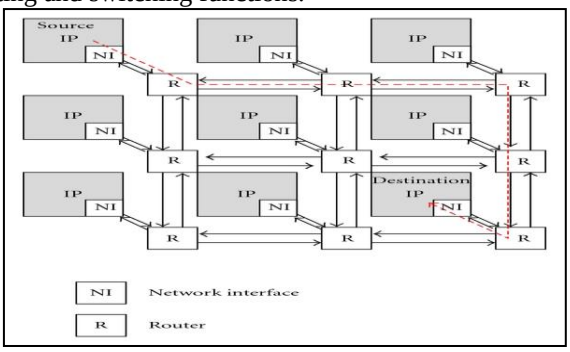

Fig. 1: Network on Chip

The Network on chip has 2 main components they are network interfaces and routers [2].

A. The network interfaces

Network interface does the conversion of the packed based communication to the high-level protocol which the intellectual property module uses. The network interface include two ports they are the NI kernel and the NI shell. The NI kernel packets the messages and schedules to router which implement the end to end flow control. The NI shells implements the connections, ordering of the transaction and other higher-level issues.

B. Router

The router can be inter-connected among themselves and over the network. The routers can have multiple links. The routers send the packets of data from one network interface to another network interface. These packets has one or more flits which has minimum transmission unit [3].

III. NOC ROUTER ARCHITECTURE

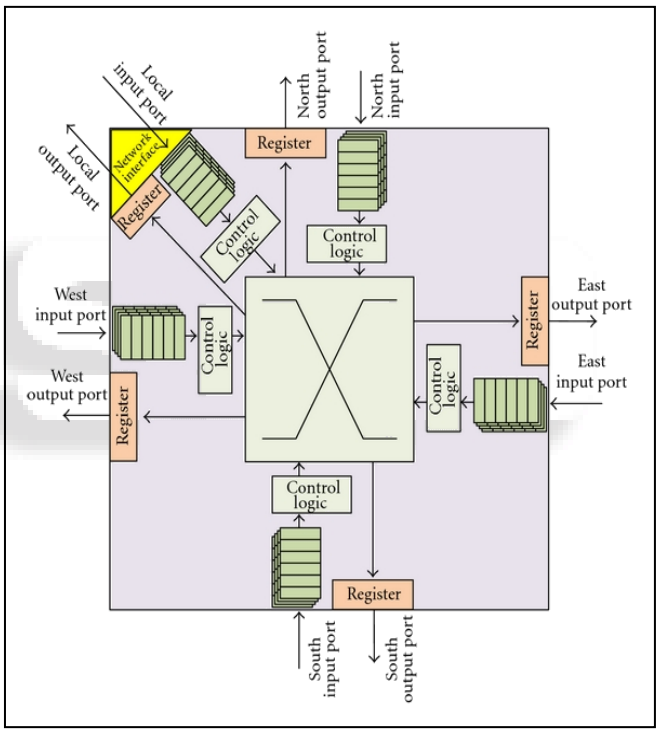

Fig. 2: NoC Router Architecture.

The Network on Chip Router architecture has following components they are input ports, output ports, register, control logic. The port connects to next port on the near to each router through a set of physical inter-connect wires. The routers job is to route the flits coming inside from one input port in to an appropriate output port and then toward the final destinations. For the data packet the corresponding head flit indicates its intended destination. After checking the front flit, the router control logic would detect the output direction to route all the subsequent body and end flits associated with this data packet in accordance to the routing algorithm implemented [4].

Initially the data is obtained through the Input Port, checks from where the data to be sent & then move into router. The Virtual Channel (VC) would store the data from the incoming port. The Virtual Channel has been determined

by the directing calculation in the past switch. The Output Port (OP) and Virtual Channel eighteen design number is determined by the directing calculation and is put away alongside the bounce. In the following stage the neighboring switches report their conditions of the VCs. This progression will ensure that the IP just considers flutters which get an opportunity to be sent. On the off chance that IP 2 of the following switch reports that all its VCs are full, all flutters which need to be steered towards IP 2 doesn't need to be thought of. This progression will make a piece exhibit of Virtual Channels which contain information and gets an opportunity of sent. An authority inside the Input Port, picks one of the mentioning Virtual Channels and the Input Port reports the ideal Output Port to the switch. Many Input Port may demand for a similar course, henceforth next intervention step would be required to determine the contention [5]. The Input Port that won the Input Port/Output Port intervention, transmits the flutter and erases it from the picked Virtual Channel. The bounce will go through the crossbar and is gotten at the Output Port where the relative location is refreshed in header flutter [6]. If there should be an occurrence of relative tending to plot, the location tuple speaks to the separation between from the present hub to the goal [7]. Since the separation changes when dance navigates the NoC, it must be refreshed at each hub the bounce passes. At the point when all the components of the location tuple equivalent 0, the flutter arrived at its goal and is expelled from the system. After the location is refreshed, the flutter is passed on towards the next switch [8].

IV. PROPOSED METHODOLOGY

In this paper, the performance of dual-crossbar over Network on Chip design we propose a new architecture called Dual X-Bar, which has benefits of buffer less and buffers which would enable low latency routing at a network with low load and is limiting buffering capability to handle many packets at a network with high load. This proposed design is combines the primary crossbar and a secondary crossbar. Primary crossbar will ensure the Central Processing Unit to Central Processing Unit & Graphics Processing Unit to Graphics Processing Unit communications Secondary cross-bar is used for Graphics Processing Unit to Memory Controller and Central Processing Unit to Memory Controller Communications. Graphics Processing Units (GPUs) are is generally used in high performance computing systems and data centers for large data processing. A Graphics Processing Unit computes application which comprises of huge quantity of kernels which is comprised of hundreds of threads. These strings are sorted out into helpful string exhibits and are booked on Streaming multiprocessors. So as to expand the computational force in present day elite Graphics Processing Units, the quantity of Streaming Multiprocessors continues expanding. As the quantity of SMs will press the Network-on-Chip (NoC) which will interface the SMs to the last level reserve (LLC) cuts and memory controllers (MCs).

Adaptable Network on Chip topology have been proposed, including cross section, Clos and butterfly. These system topologies were considered for CPU frameworks in which the various CPUs would impart each other to ensure reserve lucidness. When there is no communication between streaming multiprocessors, that is coherence at the streaming multiprocessors side L1-cache is achieved through software issuing flush operations to the shared last level cache. Scalable CPU topologies lead to not used links and are not sufficient enough to employ with both power and area. An X-bar Network on Chip would fit by connecting links to connect the Steaming Multiprocessors to the last-level cache slices and opposite. There is no links to connect the Steaming Multiprocessors between them. Therefore scaling of a cross-bar NoC to large SM counts at high clock frequency is problematic [9].

V. PROPOSED ARCHITECTURE

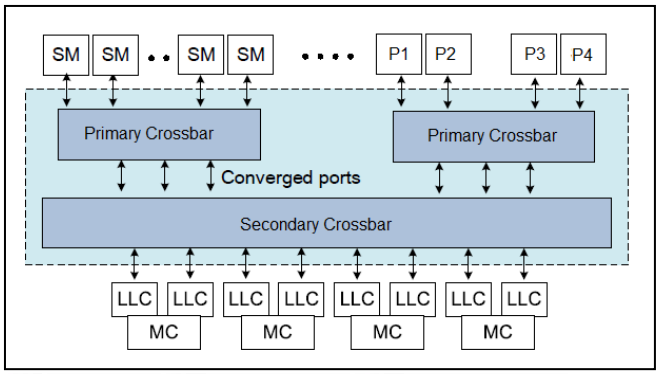

Fig. 3: Top Level Architecture of Proposed Dual Crossbar

CD Xbar has two sorts of cross-bars they are neighborhood crossbar and worldwide crossbar. A neighborhood crossbar will unite input ports from the Streaming Multiprocessors in to a merged port. The worldwide crossbar separates joined ports to the last level reserve (LLC) cuts and memory controllers. CPU-GPU heterogeneous frameworks are rising as structures of decision for superior vitality proficient figuring. A GPU-figure application commonly comprises of various portions that are made out of (up to a huge number of) strings. These strings are composed into helpful string exhibits (CTAs) and are planned on gushing multiprocessors (SMs) [10]. For GPUs, the crossbar is the most cost-productive NoC by just supporting correspondence between Streaming multiprocessors and last level reserve cuts, and not among SMs nor last level store cuts. In a traditional crossbar Graphics Processing Unit Network on Chip, the solicitation arrange associates all the gushing multiprocessors to all the last level store cuts through a completely associated crossbar. The reacted organize does the opposite, which interfaces all the last level store cuts to all the gushing multiprocessors. Because of the tremendous hole between the Streaming multiprocessors tally and last level store cut check, a full X-bar displays the natural confinement which is noteworthy part of the system is under-used, just a predetermined number of connections are adequately utilized in each cycle. Numerous connections in a crossbar Network on Chip are not used to sufficient. This gives a chance to a devise to unite or wander Network on Chip topology which accomplishes much better equipment usage while accomplishing comparable execution as a completely associated X-bar. Double X-bar comprise of a few nearby

crossbars and a worldwide crossbar. SMs are associated with neighborhood crossbars and last-level reserve cuts are associated with the worldwide crossbars. The component of the merge and separate topology is the utilization of joined ports as middle of the road ports. The complete number of united ports is a harmony between the quantity of gushing multiprocessors and last-level reserve cuts. There are less merged ports than gushing multiprocessors to improve equipment use. There are more united ports than last-level reserve cuts to maintain a strategic distance from blockage.

Double crossbar bunches Streaming multiprocessors into a few gatherings and every single gushing multiprocessor in a solitary gathering utilize a nearby crossbar to interface with the worldwide crossbar. The neighborhood and worldwide crossbars are input lined crossbar switches. The connection bearing is from gushing multiprocessors to LLC in the solicitation arrange. In the answer arrange, a united port interfaces a yield port of the worldwide crossbar to an information port of a nearby crossbar. The connection course is from level store cut to Streaming multiprocessors. The joined ports additionally overcome any barrier between the Streaming multiprocessors and the level reserve cut cuts by first contracting all the more gushing multiprocessors ports to a couple combined ports through a nearby crossbar which are then separated through the worldwide crossbar to the level store cut cuts. The quantity of gushing multiprocessors are continue expanding in the GPUs. In the ongoing days Processors and recollections of CPUs are likewise turning out to be power hungry.

The comparison of mesh and CD-Xbar over Normalized power is shown in fig 4.

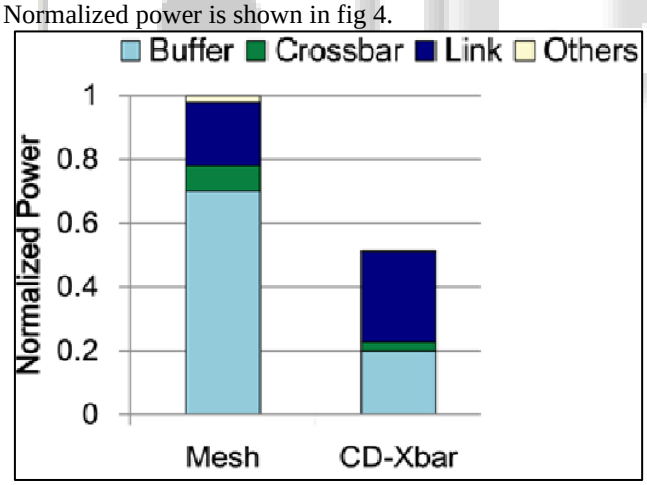

Fig. 4: Power Consumption

The comparison of Active silicon area versus the mesh and CD-Xbar is shown in Fig5.

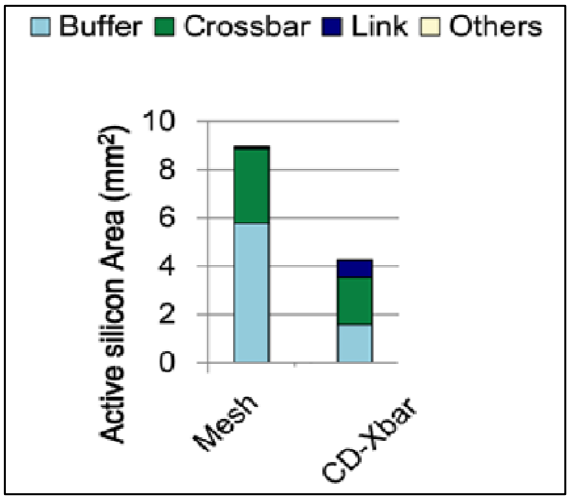

Fig. 5: Silicon Area

VI. CONCLUSION

A proposed dual crossbar merges input ports from the Streaming multiprocessors into purported met ports the worldwide crossbar separates these combined ports to the last-level store cuts and gushing memory controllers. Contrasted with a work with a similar separation transmission capacity, CD - Xbar decreases Network on Chip dynamic silicon zone. The proposed architecture reduces the hardware cost compared to mesh. The power consumption is decreased when compared to mesh.

REFERENCES

[1] CD-Xbar: A Converge-Diverge Crossbar Network for High-Performance GPUs", Xia Zhao, Sheng Ma, Zhiying Wang, Natalie Enright Jerger, and LievenEeckhout, IEEE TRANSACTIONS ON COMPUTERS, 2018

[2] "On-Chip Communication Network for Efficient Training of Deep Convolutional Networks on Heterogeneous Manycore Systems"Wonje Choi, KarthiDuraisamy, RaduMarculescu, IEEE Transactions on Computers 2018, DOI:10.1109/TC.2017.2777863

[3] LulwahAlhubailand NaderBagherzadeh, "Power and Performance Optimal NoC Design for CPU-GPU Architecture Using Formal Models"2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), DOI: 10.23919/DATE.2019.8714769

[4] "BiNoCHS: Bimodal network-on-chip for CPU-GPU heterogeneous systems", by Amirhossein Mirhosseini, Mohammad Sadrosadati, Thomas F. Wenisch, Published in Eleventh IEEE/ACM International Symposium 2017, DOI:10.1145/3130218.3130222

[5] "Achieving High-Performance On-Chip Networks With Shared-Buffer Routers", by Anh T. Tran, Member, IEEE, and Bevan M. Baas, Published in IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS 2013, DOI: 10.1109/TVLSI.2013.2268548

[6] [6]A. K. Ziabari, J. L. Abell´an, Y. Ma, A. Joshi, and D. Kaeli, "AsymmetricNoC Architectures for GPU Systems," in Proceedings of theInternational

Symposium on Networks-on-Chip (NoCS), pp. 25:125:8,September 2015.
[7] X. Zhao, S. Ma, Y. Liu, L. Eeckhout, and Z. Wang, "A Low-Cost Conflict-Free NoC for GPGPUs," in Proceedings of the Design Automation Conference (DAC), pp. 34:1–34:6, June 2016
[8] W. W. L. Fung, I. Sham, G. Yuan, and T. M. Aamodt, "Dynamic Warp Formation and Scheduling for Efficient GPU Control Flow,"in Proceedings of the International Symposium on Microarchitecture
[9] A. Arunkumar, E. Bolotin, B. Cho, U. Milic, E. Ebrahimi, O. Villa, A. Jaleel, C.-J. Wu, and D. Nellans, "MCM-GPU: Multi-Chip-Module GPUs for Continued Performance Scalability," in Proceedings of the International Symposium on Computer Architecture (ISCA), pp. 320–332, June 2017.
[1] N. E. Jerger, T. Krishna, and L. Peh, On-Chip Networks: Second Edition. Morgan and Claypool Publishers, 2017.

