# Review On Different Types Of Router Architecture And Flow Control

Omprakash Ghorse[1], Nitin Meena[2], Shweta Singh[3]

[1]*M.Tech Scholar, Electronics Department, IES College of Technology, Bhopal, RGPV Bhopal, India*

[2]*Asst. prof. Electronics Department, IES College of Technology, Bhopal, RGPV Bhopal, India*

[3]*Associate Prof. Electronics Department, IES College of Technology, Bhopal, RGPV Bhopal, India*

***Abstract*— NoC (Network on chip) is a new technology for designing of communication structure in SoC design. Where large number of IP blocks communicates with each other. Router is the backbone of NoC design which handles the communication and connecting with neighbors. Designing of router and its characteristics is directly impact on the performance of NoC. Size of the new era digital system design are shrieked and the system becomes more complex day by day. Before to design any digital system we think two more important factors are power and performance. Only the communication portion in any system consume 50% power of the total power used by the system. Here we show the different types of router architecture which can improve the system performance.**

***Keywords*— Network on chip, Flow control of flit, pipelining;**

## I. INTRODUCTION

Network-On-Chip (NoC) gives a new design to the System-On-Chip (SoC) designing. We can design a high density and complex VLSI technology on a single chip with the SoC.

As the technology improves day by day SoC becomes more heterogeneous IP (Intelactual property) blocks, and they require high performance and flexibility which is becomes more critical in SoC. The traditional bus structure which is used in SoC design is replaced by the NoC which gives several advantages over structure, performances and power.

NoC is consist of three main component switch which is also called as router, network interface (NI) and link. The main component of NoC is router whose main function is to give the route to the packet from its source to its destination. NI is in between routers and IP blocks which is responsible for packetizing and depacketizing of the message. Links are used for communication between neighboring routers. Route of the packet is determine by the router according to the routing algorithm. Routing of packets is an important aspect in NoC. The network throughput and latency is directly depends upon the router and routing algorithm.

## II. WORMHOLE SWITCHING ROUTER ARCHITECTURE

The performance and implementation of **wormhole switching** for relatively less traffic workload, makes it the preferred switching technique as compare to store and forward and virtual cut through switching. In this technique the routing decision is made on reception of the header flit. The data flits simply follow the header flit.

The router implemented using by a wormhole is Generic NoC router . It has five inputs and output ports, each of one is for local processing element (PE) and four for the neighboring routers communication. Each router also has five components: Routing Computation (RC) Unit, Virtual Channel Allocator (VA), Switch Allocator (SA), flit Buffers (BUF), and Crossbar Switch.

In a wormhole routing a message (packet) broken into a multiple flits (flow control digit) for transmission and flow control. The header flit containing all the routing information, governs the route and the remaining data flits follow them in a pipelined fashion. If the header flit is blocked, data flits are also blocked. Wormhole switching makes the end-to-end delay insensitive to the packet destination due to the pipelining of flits, and routers require only small amount of buffer space.

In a wormhole router, the switch arbiter stores priorities between different requestors for fair arbitration. These priorities are dependent upon the outcome of each arbitration.

In a wormhole router, a packet proceeds through the states of *routing*, *switch arbitration* and *switch traversal*, as traced in the flowchart.

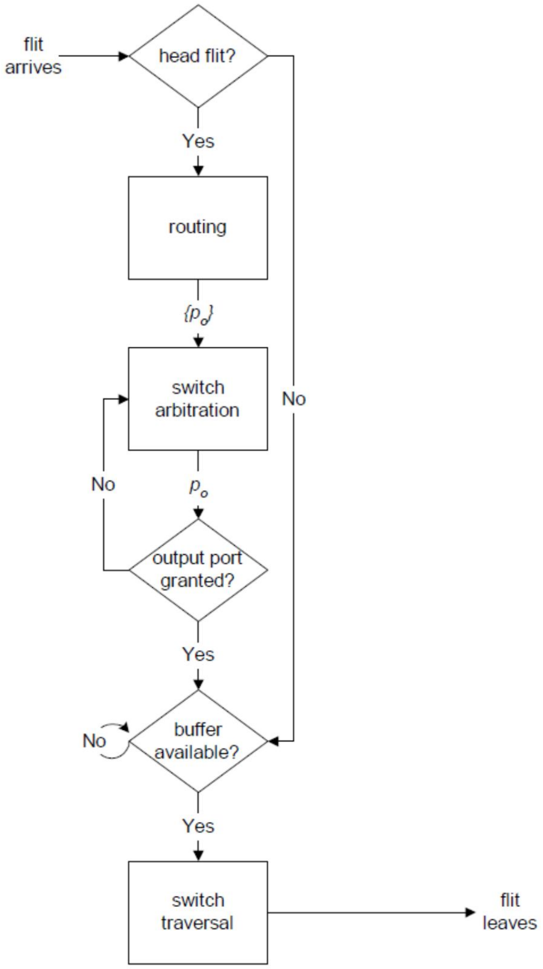

Fig. 1 Flow of a flit through routing, switch arbitration and switch traversal in a wormhole router

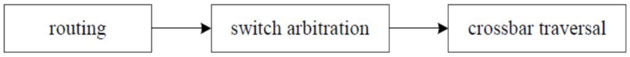

Fig. 2 The atomic module of a wormhole router, and the dependencies between them depicted with arrows

At very high work load traffic, wormhole switching implementation may be suffer from deadlocks and highly unpredictable latencies.

Wormhole router architecture increase latency, low complexity and high buffer utilization, but less efficiency.

## III. VIRTUAL CHANNEL ROUTER ARCHITECTURE

When a physical channel is divided in to a multiple number of logical channels. These logical channels are called as virtual channel(VC). The concept of ***virtual channels*** *is used by* Dally and Seitz to create a deadlock free deterministic routing. A virtual channel has its own queue, but it shares the bandwidth of the physical channel in a time multiplexed fashion.

Power efficiency is one of the most important issue in early system design which is directly impact on the system performance. For current process technologies, *dynamic power* is the primary power source consumed in CMOS circuits.

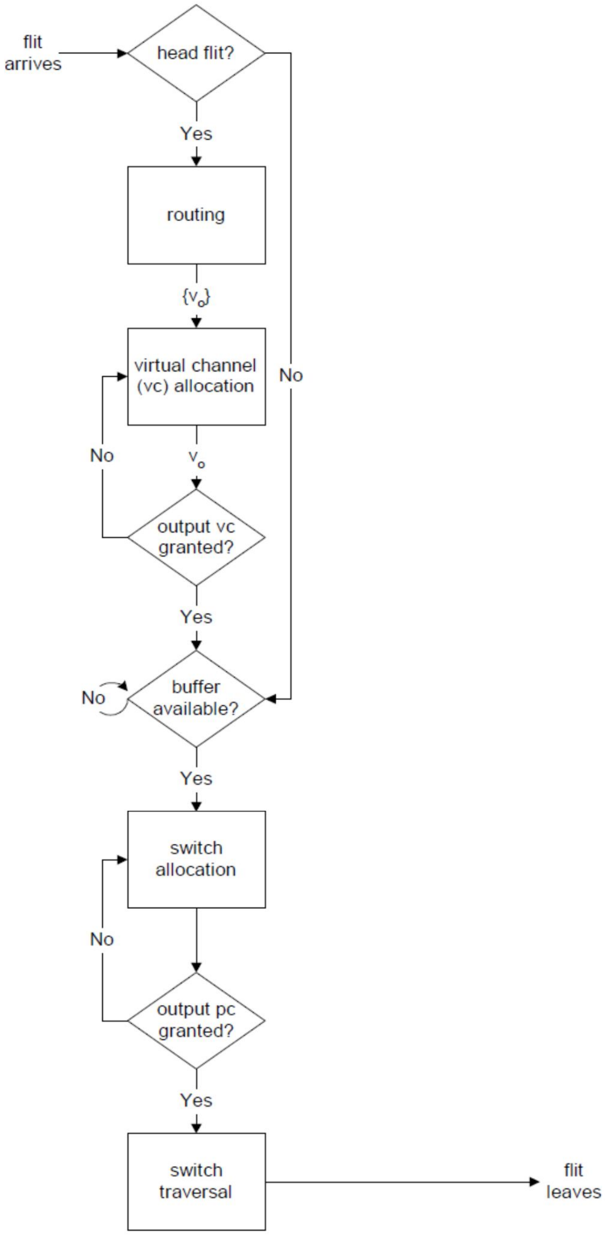

Fig. 3 The flow of flits through the states of routing, virtual-channel allocation, switch allocation and switch traversal in a virtual-channel router is depicted

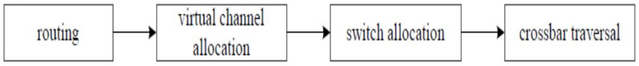

Fig. 4 Atomic modules of a virtual-channel router, with dependencies shown as arrows between them.

Hence, Figure shows the flow of a flit in a virtual-channel router through the routing computation logic, the virtual-channel allocator, the switch allocator, and the crossbar switch module. It also shows the dependencies between these modules. A flit needs to know which virtual channels to request for before it can move on to virtual-channel allocation, and it obtains this information from the routing logic, thus the dependency of virtual-channel allocation on routing logic. Thereafter, after securing a virtual channel, it arbitrates for a physical channel through the switch allocator. Naturally, the flit needs to win the switch allocation before it can traverse it. The switch arbiter stores priorities between different requestors for fair arbitration. These priorities are dependent upon the outcome of each arbitration.

Although the VCFC (virtual channel flow control) is able to increase the network throughput while reducing the transmission delay. The use of VCFC decouples the buffers from the physical communication links and routers. As such, when a certain VC is congested, the packets in other VCs can still progress through some links in the network and so the network throughput can be significantly improved.

Virtual-channel flow control exploits an array of buffers at each input port. By allocating different packets to each of these buffers, flits from multiple packets may be sent in an interleaved manner over a single physical channel. This improves both throughput and latency by allowing blocked packets to be bypassed.

The basic steps followed by a virtual-channel router are shown below:

A. **Routing**. When the first flit of a new packet arrives at any input port of the router. First the routing field is examined and a set of valid output virtual-channels on which the packet can be routed is generated. The number of output VCs generated by the routing logic will depends on the routing function.
B. **Virtual Channel Allocation**. After the routing computation packet is to made an attempt to allocate an unused VC. A request which is made for one of the virtual-channels returned by the routing function. Allocation involves arbitrating between all those packets requesting the same output VC.
C. **Switch Allocation**. Each packet maintains state indicating the availability of buffer space at their assigned output VC. When flits are waiting to be sent, and buffer space is available, an input VC will request access to the necessary output channel via the router's crossbar. On each cycle the switch allocation logic matches these requests to output ports, generating the required crossbar control signals.
D. **Crossbar Traversal**. Flits that have been granted passage on the crossbar are passed to the appropriate output channel and flit is forwarding to the next hop.

## IV. SPECULATIVE VIRTUAL CHANNEL ROUTER

In a VC router, a head flit has to ensure that it has first reserved an output virtual channel for the packet before it can request for its own passage through the crossbar and leave for the next hop. Therefore a dependency exists between virtual-channel allocation and switch allocation. This serialization process of arbitration of a virtual channel and the switch allocation significantly increases the latency of a virtual-channel router.

A speculative virtual-channel router take away this dependency. In this we **assumes** a flit will succeed in its virtual-channel allocation, and proceeds to request for crossbar switch passage in parallel. If the flit is really allowed an output virtual channel, and it won the switch arbitration too it can immediately traverse the crossbar and leave for the next hop. if the speculation turns out to be incorrect, the switch passage is simply wasted. This greatly shortens the critical path of a virtual-channel router, potentially reduces the router pipe line. A shorter router pipeline results in reduces network latency and higher throughput due to quicker buffer turnaround.

To avoid any negative impact on throughput, a speculative virtual-channel router should be conservative, prioritizing non-speculative requests over speculative ones. Thus, speculative switch request will never be granted if there are other non-speculative requests, and crossbar switch slots will never be wasted if there are non-speculative flits waiting to use it.

In a speculative virtual channel router, the switch allocator speculatively assumes that a packet will succeed in obtaining a free output virtual channel from the virtual-channel allocator, and thus, proceed to request for the desired output port before it has secured an output virtual channel. The virtual-channel allocation and speculative switch allocation states thus proceed in Parallel, the speculation is conservative i.e., it will never reduce router performance. This speculative architecture reduces a virtual-channel router's pipeline as compare to the wormhole router, effectively reducing the zero-load latency of a virtual-channel network to that of a wormhole network.

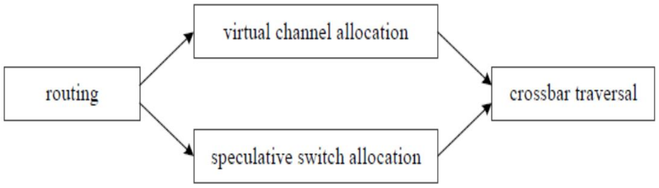

Fig. 5 Atomic modules and dependencies of a speculative virtual-channel router. The dependency between virtual-channel allocation and switch allocation is removed with speculation

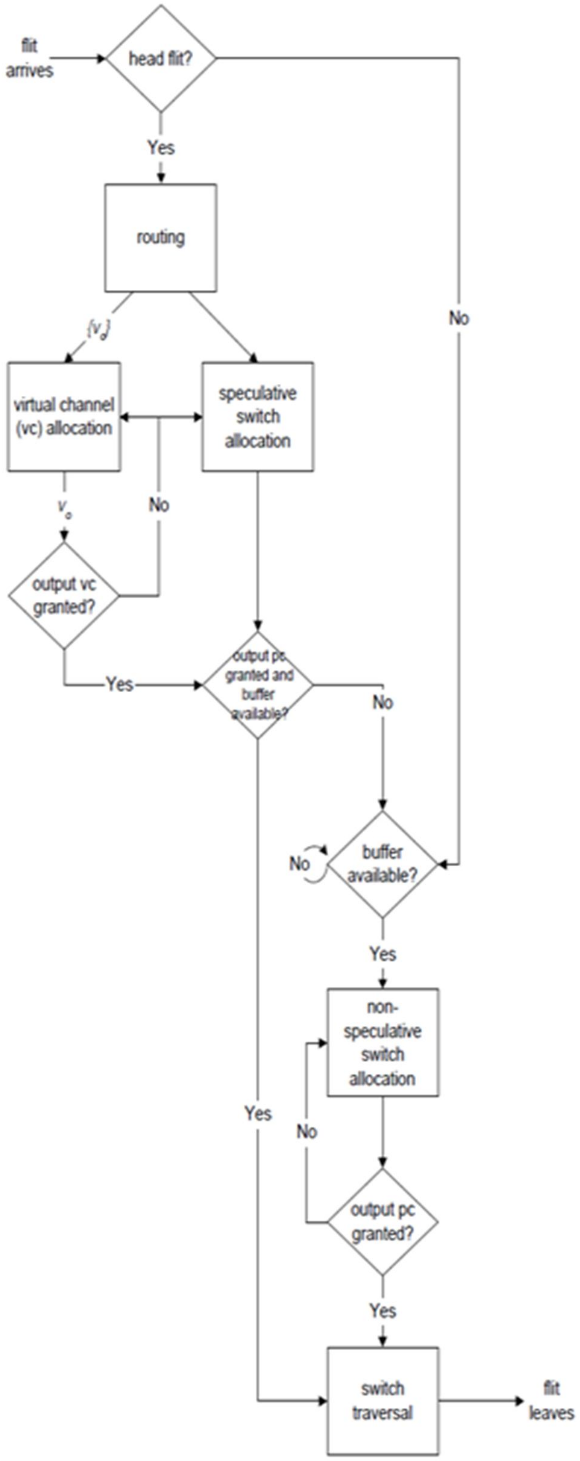

Fig. 6 shows Flow of a flit through a speculative virtual-channel router**.**

## V. CONCLUSION

As we see that there are many types of router architecture in NoC are used. First we see that the wormhole architecture this is simple in implementation but it is suffers from deadlocks when the network is in heavy workload traffic so that latency is increases which degrade the system performance. It also has a low efficiency.

In VC & speculative architecture uses three pipeline stages where as in wormhole router uses four pipeline stages to transfer a flit from one hop to the next hop. Reducing pipelining increases the network throughput and reduces the latencies.

In addition to reducing the latency of a conventional virtual-channel router, a speculative virtual-channel router also increases network throughput, with its faster recycling of buffers.

## ACKNOWLEDGEMENT

I am study of many papers which is on network on chip are published by many authors. I am thankful to all that give me the more knowledge about NoC router.

## REFERENCES

[1] Li shiuan peh” flow control and micro architecturereal mechanisum for extnding the performance of interconnection network” in 2001.

[2] *L.Rooban, S.Dhananjeyan*“Design of Router Architecture Based on Wormhole Switching Mode for NoC” International Journal of Scientific &Engineering Research Volume 3, Issue 3, March-2012

[3] *N. Kavaldjiev, G.J.M. Smit, P.G. Jansen* “A Virtual Channel Router for Onchip Networks” Proceedings, IEEE International SOC Conference, 12–15 September 2004, pages: 289–293.

[5] *Robert Mullins, Andrew West and Simon Moore* **“**Low-Latency Virtual-Channel Routers for On-Chip Networks” Computer Laboratory, University of Cambridge

[6] *Ville Rantala Teijo Lehtonen Juha Plosila* “Network on Chip Routing Algorithms” TUCS Technical Report No 779, August 2006 page: 09.

[7] *Arnab Banerjee, Robert Mullins and Simon Moore* “A Power and Energy Exploration of Network-on-Chip Architectures” IEEE 2007

[8] Mostafa S. Sayed, A. Shalaby, M. El-Sayed Ragab, Victor Goulart, “Congestion Mitigation Using Flexible Router Architecture for Network-on-Chip”2012 IEEE.

[9] Son Truong Nguyen Shigeru Oyanagi “The Design of On-the-fly Virtual Channel Allocation for Low Cost High Performance On-Chip Routers”2010 IEEE.

[10]. Nicopoulos et al., “ViChaR: A Dynamic Virtual Channel Regulator for Network-on-Chip Routers,” in Proc. of the 39th Int’l Symp. on Microarchitecture, 2006.

[10] Daniel U. Becker, William J. Dally “Allocator Implementations for Network-on-Chip Routers”2009 IEEE.

[11] Ioannis Nousias, Tughrul Arslan “Wormhole Routing with Virtual Channels using Adaptive Rate Control for Network-on-Chip (NoC)”2006 IEEE.