A Novel Bufferless Adaptive Router in NoC

Natasha Elizabeth Francis
PG Scholar, VLSI & ES, ECE Department
TKM Institute of Technology
Kollam, India
natashafrancis@gmail.com

Dhanya Oommen
Assistant Professor, ECE Department
TKM Institute of Technology
Kollam, India
dhanyaoommen@gmail.com

***Abstract*—Network-on-Chips (NoCs) are a general purpose, scalable communication framework for on-chip processing cores. Buffered routers dominated NoC designs due to their simple wormhole switching and high load handling capacity. But buffers in on-chip networks consume significant energy, occupy chip area and increase design complexity. A new bufferless deflection router with nearly the performance of conventional buffered routers is proposed. It shows substantial decrease in NoC energy consumption. Bufferless routing relies on deflection of flits that do not make progress. The port allocation of the deflected flits can be accomplished with the help of a partial permutation network. Ideally these deflections are rare. Since the flits are never stored in the router, the buffers of the router are removed to save silicon and energy.**

***Keywords*—On-chip networks, hot-potato routing, flow control, arbitration, permutation, prioritization, flit.**

I. INTRODUCTION

As the number of processor cores and IP blocks integrated on a single chip is steadily growing, a systematic approach to design the communication infrastructure becomes necessary. Communication is becoming a bottleneck in System-on-Chip (SoC) design, and today chip design methodologies are much more communication centric rather than computation centric. Bus based communication systems are replaced with networked architectures because of better electrical properties, higher bandwidth, energy efficiency, scalability, and multiple concurrent communications. Thus, Network-on-Chip (NoC) is the communication infrastructure for Multiprocessor System-on-Chip (MPSoC).

NoC is the communication platform that can be reused for a number of systems. NoC is design dependent and application dependent. The most important factors in NoC are topology, routing, switching and arbitration [10]. Topology determines the arrangement of channels and nodes in the network, analogous to road map. Routing and flow control build on properties of topology. Routing determines the path from source to destination. A flow control policy characterizes the packet movement along the NoC and as such it involves both global (NoC level) and local (router level) issues [6]. Switching defines how the data is transmitted from the source node to the target one. It determines how data flows through routers in the network. Arbitration logic implemented in the router selects one input port when multiple packets arrive at the router simultaneously requesting the same output port [5].

The remainder of the paper is organized as follows. Section 2 focuses on related work. Section 3 presents the new on-chip router architecture with minimum buffering to support deflection routing. Section 4 details the results and discussion. A brief conclusion is drawn in Section 5. This also outlines the further researches.

II. RELATED WORK

First, Inter-chip router designs exploit a large number of buffers [4] which involves increased area and complexity in protocol. Bufferless NoC design has recently been evaluated as an alternative to traditional virtual channel buffered routers by Moscibroda and Hayenga [7], [8]. Deflection routing is being proposed for networks on chips since it is simple and adaptive [9]. A priority based deflection policy that uses global and history related criterion can achieve both better average case and worst case performance than a non-priority policy [3].

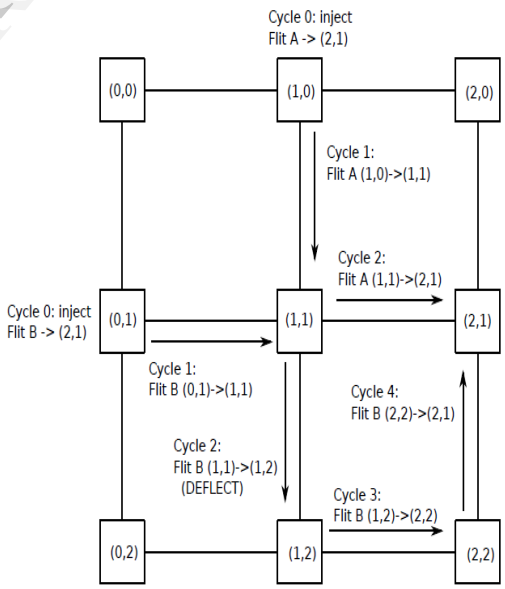

Fig. 1. An example of bufferless routing

The An example of bufferless deflection routing in a 3x3 mesh network is shown in Figure 1. Two nodes, (0,1) and (1,0), inject flits destined to (2,1) in the same cycle 0. When two flits arrive at (1,1) in cycle 1, one wins arbitration for the link toward (2,1), and the other is deflected. In contrast, a traditional buffered network would have buffered one of the flits at the center router (1,1) for one cycle. When two flits contend for one output port, it avoids the need to buffer by misrouting one flit to another port.

Recent work by Moscibroda and Mutlu [6] in bufferless routing guarantees that all flits entering a router can leave it, because there are at least as many output links as input links. The bufferless deflection algorithm depicted in Figure 2 considers each flit in turn from highest to lowest priority, to allocate output ports. The flits in priority order require a sort function. This is best implemented in hardware as a sort network, in which each stage selectively performs a series of swaps based on pair wise comparisons [1]. A three stage network is sufficient to sort four flits by priority. Once flits are sorted, logic must allocate ports in priority order. The port allocation can be precomputed for all possible permutations of input flit priorities [2]. However, this incurs significant expense in control logic area. Therefore, this is not considered in the proposed bufferless deflection routing model.

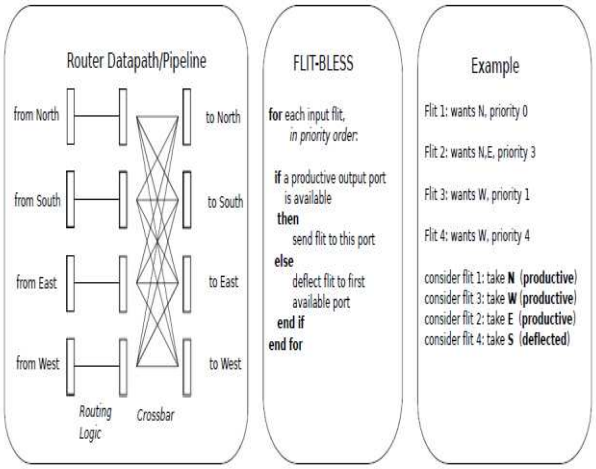

Fig. 2. Summary of bufferless deflection routing as in [6]

III. Router Architecture

The primary function of bufferless router is deflection routing. In earlier work, deflection routing led to inefficient port allocation. This is because of the sequential dependence that deflection implies. But, the sequential port allocation is not necessary for ensuring mutual exclusion on output ports. Therefore, here the deflection routing problem is mapped to a permutation network.

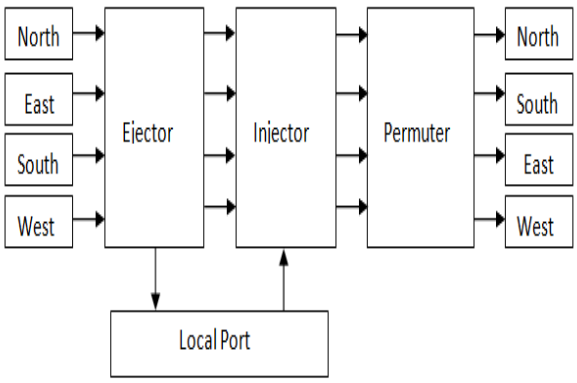

Fig. 3. Bufferless deflection router pipeline

Figure 3 shows the bufferless deflection router pipeline. The router pipeline contains two stages, eject/inject and permute. Ejection and injection is a separate stage prior to the permutation network which performs the arbitration. This stage allows inserting and removing flits from the local port before the four input flits reaches the arbiter. After ejection and injection, the resulting flits then progress down the pipeline into the permute stage. The router's critical path is through the permutation network. Thus, the eject/inject stage does not impact the critical path.

Normally all router designs assume four input ports and four output ports, without regard to the fifth, local, router port. The local port has slightly different behavior. The ejector and injector takes care of the routing of the flits to and from the local port. The particulars of the ejector are the ejector tree and kill blocks. The ejector tree performs the arbitration to identify locally destined flits. When a flit is chosen for ejection, the tree directs it to the local ejection port, and the ejector kill logic removes it from the pipeline. Figure 4 shows the ejector tree and kill blocks of the ejector.

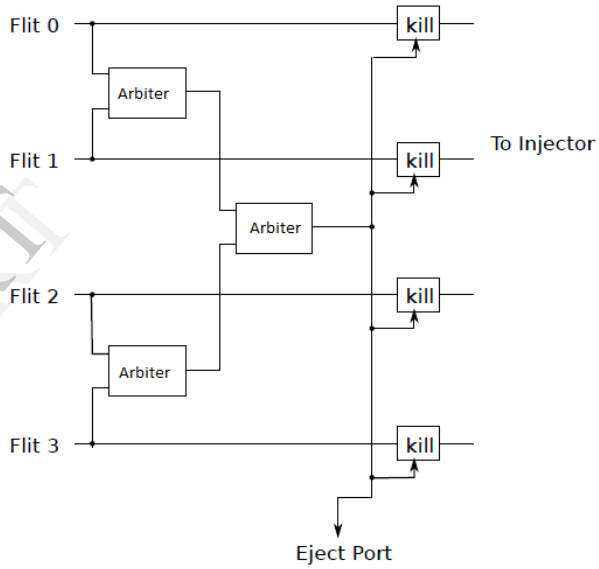

Fig. 4. Ejector tree and kill blocks

The arbitration rules for the ejector block are exactly identical to the permuter arbiter block with the additional stipulation that only locally destined flits are considered. Finally, the ejector kill blocks remove ejected flits from the pipeline before they advance to the injector and permuter units.

The injector injects flits from the local port into the router pipeline. The injection can only occur when there is a free slot. The free slot for injection occurs because of an empty input link or because of an ejection. An open slot is indicated by a cleared valid bit. A detail of the injector logic is shown in Figure 5.

A permutation network directs deflected flits to free ports in an efficiently parallelized way. The permutation network is composed of 2x2 arbiter blocks that either pass or swap their arguments. This idea leads to a new router architecture.

A. *Packet Prioritization*

A single packet is prioritized above all other packets for long enough that its delivery is ensured. This packet is known as the Golden Packet. If every packet in the system eventually receives this special status, then every packet will eventually be delivered. The Golden Packet prioritization just requires a small comparator and the prioritization rules are designed to be very simple. These rules guarantee delivery of the golden packet. The golden packet always wins against other traffic. The contests between two non-golden flits are decided pseudo randomly. The Golden Tie rule is that if two flits are golden, the lower-numbered flit which is the first in the golden packet wins. This is a rare case when two flits of the golden packet contend. At that time the earlier flit using its in-packet sequence number is prioritized.

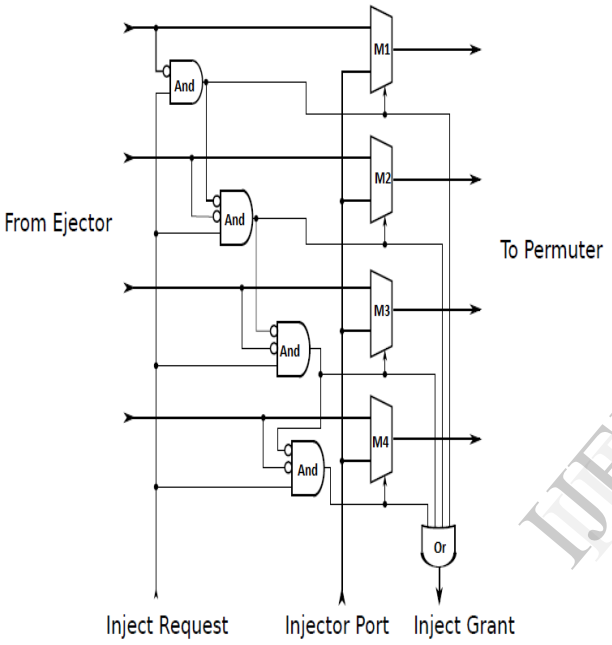

Fig. 5. Injector Control Logic

IV. Results And Discussion

Verilog HDL is used to design the router and the behavioral simulation is performed using ModelSim SE 6.2c design suite from Mentor Graphics to validate the functionality of the design.

The ejection arbitration validates three different cases, first is when both flits are locally destined, the next one is when one flit is golden and finally when neither flit is golden. When both flits are golden three other conditions has to be checked. Firstly, when both locally destined flits are golden then the lower flit within the packet is picked. Next, when one flit in both the flits destined for the local port is golden. In that condition, the golden flit is picked. Lastly, when no flit is golden one flit among the two locally destined flits is picked pseudo randomly. When only one flit is locally destined that flit is picked by the ejection arbiter. No flits are picked when no flits are locally destined.

The injection from the local port is handled as a separate stage prior to the permutation network which performs the arbitration. The flit from the local port is injected into the router pipeline only when the inject request is high and when there is a free slot. The permutation network is responsible for the routing of the flits from the four ports, namely the North, the East, the South and the West.

The bufferless deflection router simulation result which shows the ejection and the injection from the local port and the permutation routing of the flits from the other four ports is shown in Figure 6. The routing is based on the packet prioritization scheme.

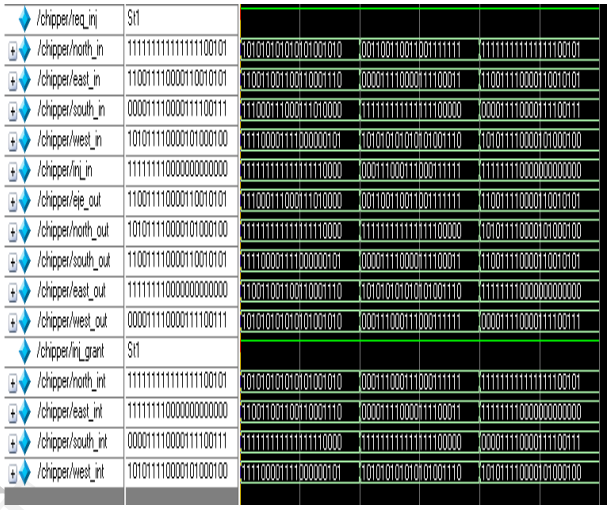

Fig. 6. Router Top Module

V. Conclusion

The successful simulation of this router design for bufferless deflection networks indicates that the performance for systems with low to medium network load is not degraded. Also, since buffers are not used within the system it can be safely concluded that this router drastically reduces network power and area consumption and hence reduces the hardware cost. This design replaces the router core with a partial permutation network and employs Golden Packet, an implicit token-passing scheme for cheap livelock freedom. Thus, makes the network truly bufferless.

Though this design is simple, there is a large design space that spans the gap between large, traditional buffered routers and simple deflection routers. Several directions are possible for future work. The Golden Packet prioritization can be extended to any non minimal adaptive interconnects in order to provide livelock freedom. Likewise, the basic permutation network structure can be used with other priority schemes by modifying the comparators in each arbiter block. Also, at high network load inefficient routing arbitration can lead to a higher than necessary deflection rate, and a simple change to the router design reduces this problem.

References

[1] John Jose, Bhawna Nayak, Kranthi Kumar, Madhu Mutyam, "DeBAR: Deflection based adaptive router with minimal buffering", *in Design Automation and Test in Europe Conference*, pp.1583-1588, March 2013.

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

[3] Chris Fallin, Greg Nazario, Xiangyao Yu, Kevin Chang, Rachata Ausavarungnirun, Onur Mutlu, "MinBD: Minimally-Buffered Deflection Routing for Energy-Efficient Interconnect", *in Sixth IEEE/ACM International Symposium on Networks on Chip (NoC)*, pp.1-10, May 2012.

[4] Robert Mullins, Andrew West, Simon Moore, "Low-Latency Virtual-Channel Routers for On-Chip Networks", *ISCA,* 2004.

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

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

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

[8] M. Hayenga, N. Jerger, and M. Lipasti, "Scarab: A single cycle adaptive routing and bufferless network," *in 42nd Annual IEEE/ACM International Symposium on Microarchitecture,* pp. 244–254, December 2009.

[9] J. Duato and S. Yalamanchili, "Interconnection networks: an engineering approach," *Morgan Kaufmann,* 2003.

[10] W. Dally and B. Towles, "Principles and Practices of Interconnection Networks," *Morgan Kaufmann*, 2004.