A STUDY ON DEFLECTION ROUTERS IN NETWORK ON CHIP

[1]Midhula K S, [2] Sangeetha Jose
[1]MTech Scholar, [2]Assistant Professor
[1]Network Engineering,
[1]Government Engineering College Idukki, Kerala

Abstract: A traditional Network-on-Chip (NoC) router uses input buffers to store incoming packets. These power hungry buffers defect NoC applicability and scalability. Hence buffer-less or minimally buffered deflection routing is the ultimate solution. This paper discusses about different deflection routers in NoC. Buffer-less deflection routing removes input buffers in order to improve energy efficiency and uses deflection to resolve contention of flits. In the case of high network load, deflection causes unnecessary network hops and wasting power which leads to performance reduction. Hence minimally buffered deflection routing is the primary concern to reduce the power. It is a challenge to optimize power consumption in NoC. Different deflection routers are used to reduce power and to improve performance. Minimally buffered, single-cycle deflection routing will overlap the operations (Injection, Ejection, Pre-emption, Re-injection) into a single module execute in a single-cycle and reduces the critical path latency and increases performance.

Index Terms **- Cheap Interconnect Partially-Permuting Router, Minimally Buffered Deflection Router, Deflection Based Adaptive Router, Smart Late Injection Deflection Router, Minimally Buffered Single Cycle Deflection Router**

I. INTRODUCTION

The style of technical world and the intervention of new ideas are improved by the introduction of SoC (System on Chip). SoC have buses and point to point (P2P) connections to connect different cores inside the chip. However, there are some limitations such as high design complexity, long wire delays and poor scalability [1].

Figure 1 shows bus based communication on SoC. In the case of NoC (Network on Chip) based communication, traditional bus structure is replaced with a network of routers which is similar to the Internet. Bus structure is blocked due to increased integration of cores inside a chip.

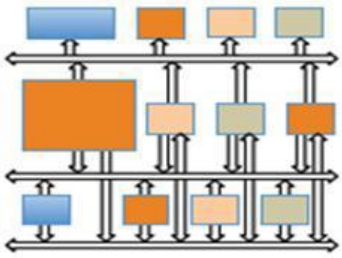

Figure 1. Bus based Communication in SoC

Design of NoCs are critical for system performance. There are different NoC designs such as 2D Mesh, Torus, Tree, Butterfly, Polygon etc. Commonly 2D Mesh used broadly in NoC. In the case of Network on Chip, where multiple cores are connected via routers. Communication between them carried out by packet based mode. Figure 2 shows the Network-on-Chip (NoC) based system. It is an on-chip packet-switched network

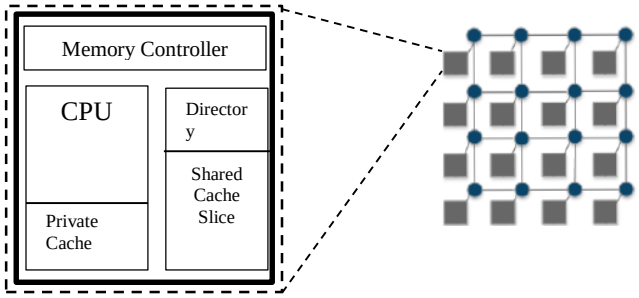

Figure 2. Network-on-Chip (NoC)-based system [1]

connects nodes which often consist of cores, cache slices, and memory controllers. For example, on a miss, a cores private cache might send a request packet to a shared L2 cache slice, and the shared cache might respond with a larger packet containing the requested cache block on an L2 hit, or might send another packet to a memory controller on an L2 miss [8].

II. BUILDING BLOCKS OF NOC

The fundamentals of NoC are Topology, Routing algorithm and Flow control [1].

2.1 Topology

Topology describes how the routers are configured. A network can be regular or irregular and it is non-blocking if it can manage all the requests that are offered to it [3]. Figure 3 shows basic regular network topologies

1. Mesh: A mesh-shaped network consists of m columns and n rows. The routers are situated in the intersections of two wires. The computational resources are near routers. Addresses of routers and resources can be easily defined as x-y coordinates in mesh [1].
2. Torus: A Torus network is an improved version of basic mesh network. A simple torus network is a mesh in which the heads of the columns are connected to the tails of the columns and the left sides of the rows are connected to the right sides of the rows [4].
3. Polygon: It is a circular network where packets travel in loop from router to router. Network becomes more diverse when chords are added to the circle. When there are chords only between opposite routers, then the topology is called as spidergon [16].

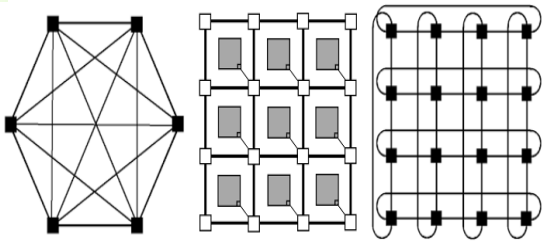

Fig. 3. Polygon, Mesh and Torus topologies in NoC

2.2 Routing Algorithm

A routing algorithm determines how the data is routed from sender router to receiver. Router. There are mainly three types of routing algorithms. They are briefly described below:

1. Deterministic routing algorithm: In this routing algorithm, the flits takes same path from one source router to another destination router for all the cases. That means, the path is deterministic path.

2. Oblivious Routing algorithm: In this approach, the flits takes different paths from one source router to another destination router. There is no deterministic path.
3. Adaptive Routing Algorithm: In this approach, the flits take different paths from one source router to another destination router with considering the network behavior.

2.3 Flow Control

Flow control will determine how packets are transmitted inside a network. There are different modes with respect to this flow control [17]. They are briefly described below:

1. Store-and-Forward Routing: Store-and-forward is the simplest routing mode. In which, packets are moved in one piece, and entire packet has to be stored in the routers memory/buffer before it can be forwarded to the next router.
2. Virtual Cut-Through Routing: Virtual cut-through is an improved version of store-and-forward mode. Packet is stored in the router until the forwarding begins. Forwarding can be started before the whole packet is received and stored to router.
3. Wormhole Routing: In wormhole routing, packets are divided into small and equal sized flits (flow control digit or flow control unit). A first flit of a packet is routed similarly as packets in the virtual cut-through routing. After first flit, the route is reserved to route the remaining flits of the packet.

In any networks, there are number of situations at contention occur. Contention is nothing but, it is the condition more than one flits demands for same output link. In this situation, it is necessary to find one flit to forward. There are some mechanisms to specify it.

1. **Drop One:** In this mechanism, one flit creating contention will be dropped out. At next cycle the sender wants to resend the flit again to the network. There is no guarantee that sender located in nearby location. Hence, there is a chance for increasing latency. Figure 4 shows the drop one mechanism.

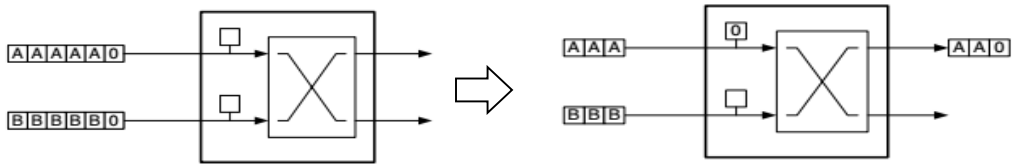

Figure 4. Drop one mechanism [16]

2. **Buffer One:** When contention situation arises, then one of the flit will be stored in buffer and other flit will gets the output port in this cycle. On next cycle the buffered flit will be injected to the network and it will go to arbitration again. Figure 5 shows that, the average packet latency increases with increasing the buffer size.

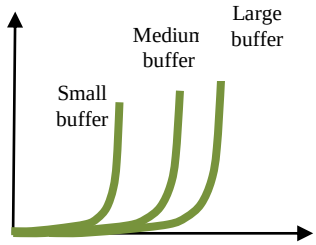

Figure 5. Buffer size vs Injection rate

3. **Misroute:** In this mechanism, the contention situation will be handled by permitting one flit through the requested output port and other flit will go through the free non-productive port, and it will reach its destination later. Figure 6 will show the misroute mechanism.

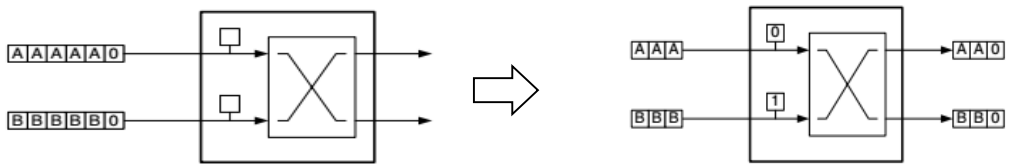

Figure 6. Misroute mechanism [16]

III. RELATED WORKS

This section provides an overall view of different deflection routers. Figure 7 shows different types of routers. They are Buffered deflection router, Buffer-less deflection router and Minimally Buffered deflection router.

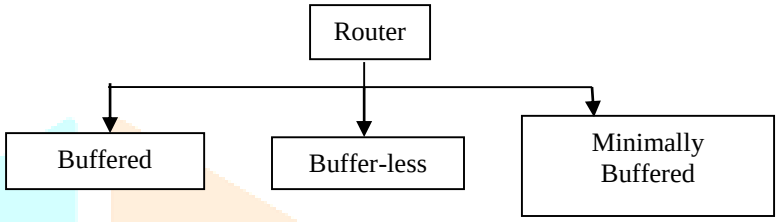

Figure 7. Different types of Routers

Conventionally buffered routers will buffer every flit enters the router from an input port before the flits can arbitrate for output ports. Buffered NoC router consumes a significant amount of energy and die area. Hence it is important that get rid from these power hungry buffers from a network. The mechanisms have been proposed to make conventional input-buffered NoC routers to more energy-efficient routers [4] are buffer-less deflection routing and minimally buffered deflection routing.

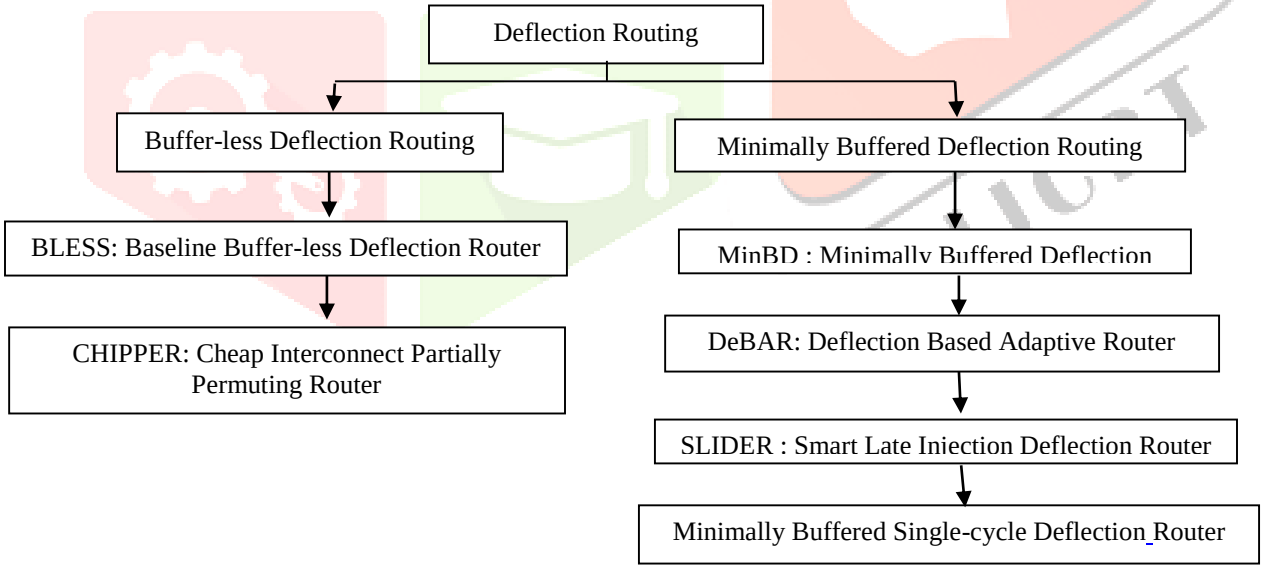

Figure 8. Different types of deflection routers

Figure 8 shows different types of buffer-less and minimally buffered deflection routers. BLESS and CHIPPER are buffer-less deflection routers. Minimally buffered deflection routers include MinBD, DeBAR, SLIDER and MinBSD.

IV. BUFFER-LESS DEFLECTION ROUTING IN NOC

4.1 BLESS

Buffer-less deflection routing [5] was first proposed by P. Baran. Buffer-less deflection routing eliminates in-network buffering. When contention occurs for a network link, then a buffer-less deflection router sends some traffic to some another output link instead. Hence use of buffers is replaced by occasional extra link traversals [8] or deflections.

BLESS is the baseline buffer-less deflection router [2]. The fundamental routing unit, the flit is a packet fragment transferred by one link in one cycle. Flits are routed independently in BLESS [8]. Since flits are routed independently, they must be reassembled after they are received. BLESS assumes the existence of sufficiently sized reassembly buffers at each node in order to reassemble and reconstruct arriving flits into packets.

In each cycle, flits arriving from neighbor router enter current router pipeline, and must leave the router at the end of the pipeline. Every input flit is assigned to some output port. Router output port assignment is done through two stages, flit ranking and port selection. In each cycle, the flits arrived at the router are first ranked based on its priority. At the same time, the router computes a list of ports which would send the flit closer to its destination called productive port for each flit. Once the flit ranking and each flits productive output ports are available, then the router assign ports to each flit. Port assignment starts from the highest ranked flit one at a time to lowest. When a node generates a packet, then the BLESS router inject that flits into the network. The router has an injection queue where flits wait until injection condition is met. Otherwise it is starved. It must remove a flit from the network when the flit arrives at its destination [6]. A BLESS router makes a local decision to inject a flit whenever there is an empty slot on its input ports.

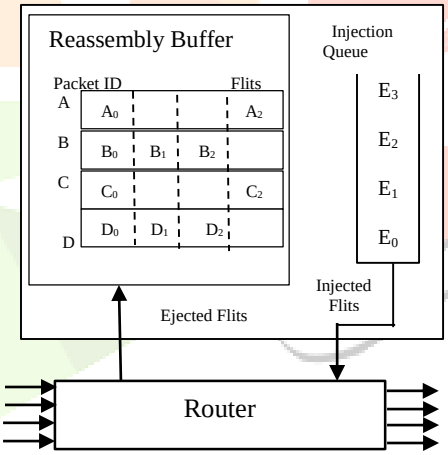

Figure 9. Reassembly buffers and injection queue in a BLESS NoC

Figure 9 depicts the reassembly buffers as well as the injection queue of a node in a BLESS NoC. When a flit arrives at its destination router then the router removes the flit from the pipeline and places it in a reassembly buffer, where it waits for the other flits from its packet to arrive and reassemble. Once all flits in a packet have arrived then that packet is delivered to the local node. BLESS has some disadvantages [7].

4.1.1 Disadvantages

1. It increases the average packet latency because deflected flits will take a longer path to the destination than necessary.
2. Since buffer-less routing deflects individual flits, flits of a packet can arrive out-of-order and at significantly different points in time at the receiver (destination node). Hence it increases buffering at receiver side.

4.2 CHIPPER: Cheap Interconnect Partially Permuting Deflection Router

CHIPPER is another buffer-less deflection router proposed to address implementation complexities of previous buffer-less deflection routers. CHIPPER router has smaller and simpler deflection routing logic than BLESS. CHIPPER introduces Golden packet prioritization mechanism and Retransmit Once [9] mechanism.

4.2.1 Golden Packet-based deflection arbitration

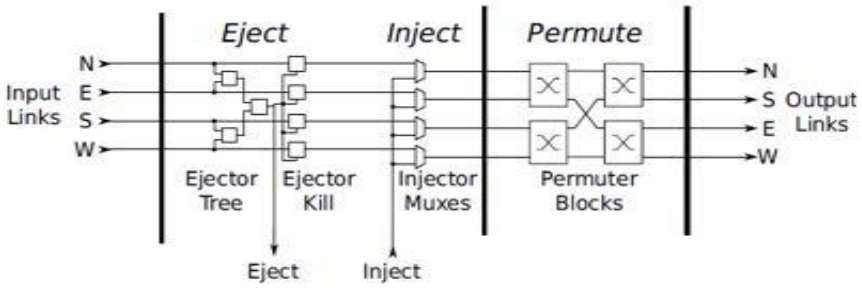

Figure 10. CHIPPER micro-architecture

When a group of flits arrived at a router then it will provide a total priority order among all flits. When the highest priority flit is delivered, then another flit will attain the highest priority. Enforcing a total priority order will create a significant complexity in a BLESS router. CHIPPER incorporates golden packet prioritization mechanism. A single packet is globally assigned as golden packet and all its flits are become golden flits. Flits in golden packet are prioritized over other flits in the network. When a packet becomes golden, then it remains so for a golden epoch, which is a time length L that is long enough so that the packet can reach any destination in the network from any source [9].

As Figure 10 shows, the permutation network allows a flit from any input port to reach any output port. First stage arbitration happened when the flits are arrived at the router. The winning flits are sent toward the second stage arbiter block, which is connected to the requested router output [8]. In the second stage, flits are arbitrates again and flits leave the second stage proceed directly to the router outputs via a pipeline register. There is no need for crossbar. CHIPPER has a shorter critical path than BLESS router because the arbiter blocks in each stage work in parallel, and the flits need not be sorted by priority first.

4.2.2 Retransmit-Once

When no output port is available in a router then arriving packet is dropped at the receiver. The receiver notes this dropped packet and once reassembly buffer space becomes available, reassembly logic in the receiver reserves buffer space for the previously dropped packet. The receiver requests a re-transmission of that packet from the sender. Thus one retransmission is necessary for any packet [9]. Retransmit-Once ensures that senders do not want to buffer data for retransmission. CHIPPER also has disadvantages.

4.2.3 Disadvantages

1. High deflection rate since golden packet has only priority.
2. 95% of flits reaches its destination without being a golden packet. Hence, there is no significant change on performance.

V. MINIMALLY-BUFFERED DEFLECTION ROUTING

5.1 MinBD: Minimally-Buffered Deflection Routing for Energy-Efficient Interconnect

Rather than using buffer-less deflection routing, MinBD uses small side buffer [10] along with deflection routing logic. The key principles for the MinBD are as follows

1. It is better to buffer the flit and arbitrate again in later cycle rather than deflecting it. That is, some buffering will avoid many deflections which reduces critical path latency.

2. Buffering every flit leads to unnecessary power overhead and buffer requirements. Hence flits will be buffered only is necessary.
3. Quick ejection of flits at its destination from the pipeline decreases complexity.

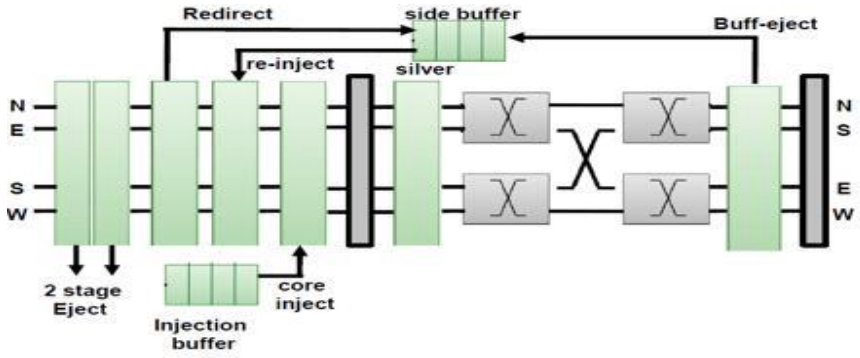

Figure 11. MinBD router architecture

The main modules of MinBD architecture are

1. Deflection Logic: MinBD uses permutation network consists of two input blocks arranged into two stages of two blocks each (4*4). This arrangement helps to send flit from any input port to any output port. Each arbiter will determine which flit has a higher priority and allow that flit in the direction of its preferred output.
2. Side Buffer: MinBD introduces a "side buffer" that buffers only flits that otherwise would be deflected [8]. It reduces the deflection rate. The arbitration and routing logic has determined which flits to deflect. The buffer ejection block recognizes the flits that have been deflected and picks one such deflected flit per cycle. Then it removes that deflected flit from the router pipeline and places in to side buffer as long as the side buffer is not full.
3. Injection and Ejection: MinBD router contains injection and ejection blocks in first pipeline stage. When a set of flits arrives on router input ports then these flits first pass through the ejection logic. Ejection logic examines the destination of each flit and if there is a flit addressed to the local router, then it is removed from the router pipeline and sent to the local network node. If more than one such locally addressed flits are present, then ejection block will pick one with highest priority.

Figure 11 depicts that MinBD router can buffer up to one flit per cycle in a single FIFO queue side buffer [8]. From the side buffer, flits are re-injected into the network at subsequent cycle. It is clear that some flits have been deflected in buffer-less deflection router are removed from the network temporarily into the side buffer and given a second chance for them to arbitrate for their productive output by re-injection.

MinBD prioritization rules [11] are:

1. Golden Tie: Ties between two Golden flits are resolved by sequence number (first in Golden Packet wins).
2. Golden Dominance: If one flit is Golden, it wins over any Silver or Ordinary flits.
3. Silver Dominance: Silver flits win over Ordinary flits.
4. Common Case: Ties between Ordinary flits are resolved randomly.

5.1.1 Disadvantages

1. Channel Wastage: The flits in the router pipeline channel reach the Permutation Deflection Network (PDN) where they are allocated with the output ports. At PDN, if a flit does not get its productive port it is moved out of the router pipeline to the

side buffer [11]. This creates an idle channel.

2. Older flit penalization: Injected and re-injected flits participate in the arbitration process for acquiring the output ports. During the arbitration there is a chance where high priority newly injected flits can deflect the incoming older flits.
3. Unnecessary flit movements: Injected and re-injected flits participate in the arbitration process for acquiring the output ports. During the arbitration if they fail to win their desired port (deflected), they become potential candidates for side buffering. Hence there is a path for the injected flits to move from core buffer to side buffer.

5.2 DeBAR: Deflection Based Adaptive Router

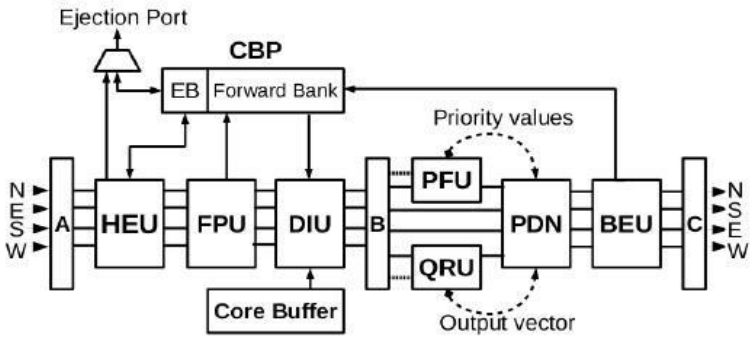

Figure 12. DeBAR Architecture

Figure 12 shows DeBAR architecture. DeBAR consists of Hybrid Ejection Unit (HEU), Flit Preemption Unit (FPU), Dual Ejection Unit (DIU), Core Buffer Pool (CBP), Priority Fixer Unit (PFU), Quadrant Routing Unit (QRU), Permutation Deflection Unit (PDU), Buffer Ejection Unit (BEU), Forward Buffer and Ejection Bank (EB). Hybrid Ejection Unit can eject flit either directly through the ejection port or through the Ejection Bank of the Core Buffer Pool then to Ejection port in subsequent cycle. CBP contain EB and Forward bank as side buffer. Flit Pre-emption Unit is used to preempt the flit from the pipeline to forward buffer to make space when there is no port is available in order to inject the flit into the router DIU will be used. DIU handle the flits injected from core buffer and forward buffer. Priority fixer unit is used to fix the priority among flits in the pipeline. Quadrant routing unit is used to compute the quadrant of the output port of each flit. Permutation deflection unit has a header enhancer to add the priority and the quadrant routing value into the header and a flit marker to mark the flit for deflection. Buffer ejection unit eject the flit which is not marked for deflection. Marked will be stored into the forward bank.

5.3 SLIDER : Smart Late Injection Deflection Router

SLIDER is Smart Late Injection Deflection Router which has the following features

1. Parallelized independent operations: SLIDER has Parallel independent operations of routing, ejection, and prioritization.
2. Selective flit pre-emption mechanism: Pre-emption is the process of preventing a flit from moving out through its assigned output port, which is needed to reduce deflection and to reduce starvation of flits in side buffer/core buffer.
3. Smart Late Injection: Injection and the re-injection stages kept late in the pipeline so as to minimize channel wastage and prevent intra-router flit movements.

SLIDER will overcome the limitations of prior deflection router smartly by making the injection unit at last stage or after the pre-emption of flits. In Figure 13, the operations such as routing, prioritization and ejection are carried out independently and the results are going to the permutation deflection network [11]. This PDN will leads the flit to pipeline register. Injection only happen after the pre-emption of flits to side buffer. The selective preemption of flit is the process of preventing flit from moving out through

its assigned output port. It is also used to make space for a starving flit waiting in the router buffer (Core buffer/ Side buffer) which is called as forced removal.

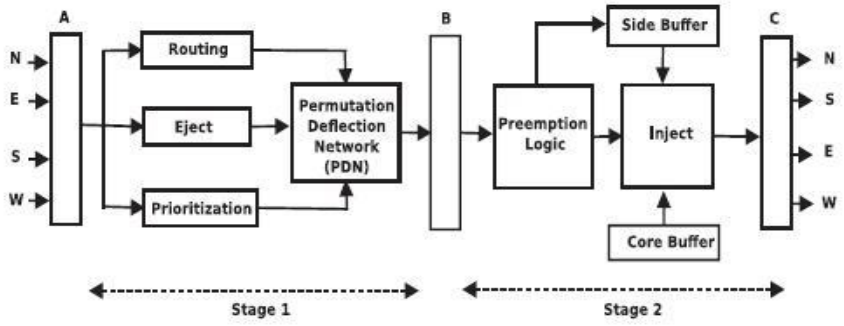

Figure 13. SLIDER architecture

5.3.1 Disadvantages

Disadvantages of SLIDER is flit ejection, injection, preemption, side buffer ejection and re-injection operations are handled by different modules. Hence it makes the routers bulky and power hungry. The structural dependency of these modules makes the existing deflection routers to operate in two cycles.

5.4 Minimally Buffered Single-Cycle Deflection Router

MinBSD is minimally buffered, single cycle, deflection router. It incorporates different operations (Injection, Ejection, Preemption, Re-injection) in a single module to handle the traffic effectively and ensures smooth flow of flits through router pipeline [12]. It performs overlapped execution of in-dependent operations. It employs an innovative PDN that incorporates the functionalities of injection, pre-emption, re-injection and side buffer ejection, all in a single module. Figure 14 shows the MinBSD router architecture. The detailed operation of each unit is explained below:

- Routing Unit(RU): It computes the productive port for each incoming flit. It first extracts the destination (Dx, Dy) address field from each flit and compares the address with current node (Rx, Ry) address and determines the productive port.
- Prioritization Unit(PU): It performs the priority computation for each incoming flit based on hops-to-destination. The flit having less distance from this router have high priority.

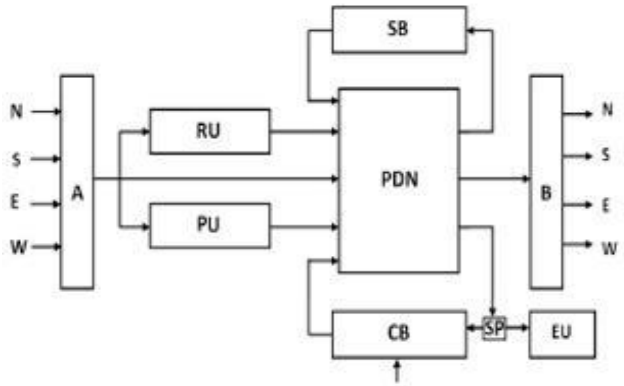

Figure 14. MinBSD architecture

- Permutation Deflection Network (PDN): MinBSD replace the conventional 4x4 PDN used in MinBD with a 6x6 PDN. It consists 6 inputs and 6 outputs. It is used for parallel allocation of the ports. There are three PDNs depending on the location of

router in the network. They are Edge PDN, Corner PDN and Center PDN for edge routers, corner routers and all other remaining routers in the network respectively.

Figure 15 shows center PDN for center router. Center PDN has 6 inputs and 6 outputs and has 6 arbiters (L1, L2, L3, R1, R2, R3) arranged in two stages as 3x2 arbiters. 4 inputs from the neighbors, one input is from the Core Buffer (CB) and the other input is from the Side Buffer (SB). Out of 6 outputs, 4 outputs go to the neighbors, one output goes to the Splitter (SP) and the other output goes to the SB.

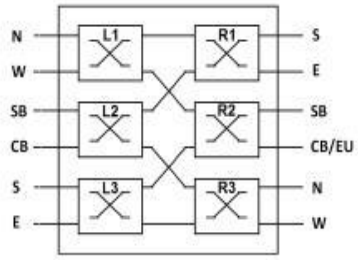

Figure 15. Center PDN

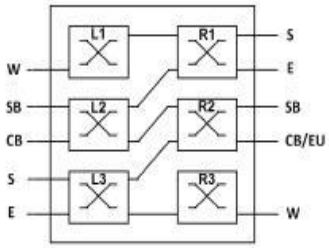

Figure 16. Edge PDN

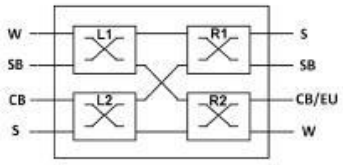

Figure 17. Corner PDN

Figure 16 shows edge PDN for edge routers. Edge PDN has 6 arbiters arranged in two stages as 3x2 arbiters but with 5 inputs and 5 outputs. CB injects flit from odd cycles and SB injects flit from even cycles.

Figure 17 shows corner PDN for corner routers. Corner PDN has 4 arbiters arranged in two stages as 2x2 arbiters with 4 inputs and 4 outputs. Injection of flits from CB done in odd cycles and from SB in even cycles into arbiters L2 and L1, respectively, which prevents buffer to buffer flit movement cycles [12].

- Side Buffer (SB) and Core Buffer (CB): SB is used to store the pre-empt flits out of the arbiter R2. Flits in SB are injected in every cycle into PDN. CB is used to store the flits from the local node and the flits through the eject channel of arbiter R2 whose eject flag is not set. Flits in CB are injected in every cycle into PDN.
- Splitter(SP): SP forwards any flit out of the eject channel of arbiter R2 in PDN either to Eject Unit (EU) or to CB based on the eject flag.
- Ejection Unit (EU): EU forwards a flit received to the eject port. It has only one ejection port per router.

VI. COMPARISON

There are different types of deflection routers in NoC. Buffer-less deflection routers such as BLESS and CHIPPER having high deflection rate. In order to reduce the deflection rate minimally buffered deflection routers are introduced.

MinBD, DeBAR, SLIDER and MinBSD are the examples of minimally buffered deflection routers. MinBD combines deflection routing with a small buffer, such that some network traffic that would have been deflected is placed in the buffer instead. DeBAR will adaptively select the routing mechanism and introduced first ejection setup in first cycle. SLIDER having different features like, late injection, selective pre-emption etc. Then MinBSD for mesh NoCs replaces traditional two-cycle deflection router architecture with simple, low complex, high speed (reduced cycle time), single cycle router architecture with an innovative PDN. MinBSD is having low latency than DeBAR and MinBD.

Table 1. Comparison of different deflection routers.

Deflection Router	Features	Advantages	Disadvantages
BLESS Baseline Deflection Router	1. Flit Ranking 2. Port Selection	1. Simpler and more energy efficient NoC design	1. Sequential port allocation 2. Increases critical path delay 3. Expensive priority arbitration 4. Reassembly buffer cost is high
CHIPPER Cheap Interconnect Partially Permuting Router	1. Parallel port allocation 2. Golden Packet prioritization	1. Reduced average network power 2. Shorter critical path 3. Smaller die area	1. High deflection rate 2. Increased latency 3. Reduced throughput
MinBD Minimally Buffered Deflection Router	1. Side Buffer is used 2. Silver flit prioritization	1. Reduced deflection rate 2. Reduced critical path latency	1. Silver flit selection is local 2. High deflection rate at high network load
DeBAR Deflection Based Adaptive Router	1. Hybrid flit ejection 2. Dual flit ejection	1. Minimal central buffering 2. Reduced average flit latency	1. Channel wastage 2. Unnecessary flit movement 3. Older flit penalization
SLIDER Smart Late Injection DEflection Router	1. Smart late injection 2. Selective flit pre-emption	1. Reduced deflection rate 2. Increase throughput 3. Reduced power consumption	1. High average flit latency
MinBSD Minimally Buffered Single Cycle Deflection Router	1. Innovative PDN	1. Low average flit latency 2. Reduced die area 3. Reduced power consumption	1. Structural limitations

VII. CONCLUSION

This survey paper aims to give an overall study on deflection routers. Deflection routers are used to avoid the buffering and dropping of packets. Deflection routers are mainly two types based on the use of side buffer. Buffer-less deflection routers have no buffers to store and forward flits but the fact is, deflection rate is high. Hence, minimally buffered deflection routers are used.

REFERENCES

[1] Ville Rantala, Teijo Lehtonen, Juha Plosila, "Network on Chip Routing Algorithms", TUCS Technical Report No 779, August 2006.

[2] D. Muralidharan, Dr. R. Muthaiah, "Bufferless Routing Algorithms: A survey", IJAER, ISSN 0973-4562 Volume 11, Number 6, pp 3811-3813, 2016.

[3] Li-Shiuan Peh, Stephen W. Keckler, Sriram Vangal, "On-Chip Networks for Multicore Systems", Springer Science, 2009.

[4] Igor Stojanovic, Milica Jovanovic, Sandra Djosic and Goran Djordjevic, "Improved deflection routing method for bufferless networks-on-chip", 2013 IEEE.

[5] P. Baran, "On distributed communications networks", IEEE, Communication Systems, 1964.

[6] Yu Cai, Ken Mai, Onur Mutlu, "Comparative Evaluation of FPGA and ASIC Implementations of Bufferless and Buffered Routing Algorithms for On-Chip Networks", IEEE, ISQED 2015.

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

[8] Chris Fallin, Greg Nazario, Xiangyao Yu, Kevin Kai-Wei Chang, Rachata Ausavarungnirun, "Bufferless and Minimally-Buffered Deflection Routing", Palesi and Daneshtalab (eds), 241-275, 2014.

[9] C. Fallin, Chris Craik, Onur Mutlu, "CHIPPER: A low-complexity bufferless deflection router", HPCA-17, 2011.

[10] C. Fallin, Greg Nazario, Xiangyao Yu, Kevin Chang, Rachata Ausavarungnirun, Onur Mutlu, "MinBD: Minimally-buffered deflection routing for energy-efficient interconnect in NOCS", 2012, pp. 110.

[11] Bhawna Nayak, John Jose, Madhu Mutyam, "SLIDER: Smart Late Injection DEflection Router for Mesh NoCs", 2012 Sixth IEEE/ACM International Symposium on Networks-on-Chip.

[12] Gnaneswara Rao Jonna, John Jose, Rachana Radhakrishnan, Madhu Mutyam, "Minimally Buffered Single-Cycle Deflection Router", DATE, EDAA, 2014.

[13] Michelogiannakis, G. Daniel Sanchez, William J. Dally, Christos Kozyrakis, "Evaluating bufferless flow-control for on-chip networks", NOCS 2010.

[14] John Jose, Bhawna Nayak, Kranthi Kumar, Madhu Mutyam, "DeBAR: Deflection Based Adaptive Router With Minimal Buffering", DATE13, 2013.

[15] Maksat Atagoziyev, "Routing Algorithms For On Chip", Thesis, 2007.

[16] S. Swapna, "Efficient Router Design for Network on Chip", Dept of ECE, NIT ROURKELA, Thesis, 2013.

[17] W. J. Dally, B. Towles, "Principles and Practices of Interconnection Networks", 2004.