# A Survey on Bufferless Router Architectures

**Divya Rajan**
New Horizon College of Engineering

***Abstract-*** *As complex SoCs develop, the communication in between the subsystems of SOCs can no longer be controlled by the traditional bus architecture because as the integration increases the bus becomes insufficient and blocks the traffic. Network on Chip replaces the traditional bus architecture by packetizing the data communication between the various segments of the chip. Buffers in on chip network consume significant energy and occupy major area in the chip. It also leads to increased design complexities. Here a study of buffer less routing is consolidated to conclude that buffer less routing is an energy efficient design alternative for on chip cache and processor to cache networks*

## I. INTRODUCTION

Router architecture plays an important role in NOC performance. A conventional input buffered VC (Virtual Channel) NOC router uses input buffers to store incoming packets and have high load handling capacities [1]. They improve the performance but consume significant amount of power. This is mainly because in a buffered network the packets wait in the buffer till it is allocated to the desired output port. It is approximated that they consume atleast 30% to 40% of the chip power [2], [3]. Circuit switching can be considered as an alternative but it has the additional requirements of connection setup and tear down

Buffer less routers are proposed to handle the increasing power concerns. When a flit reaches a buffer less router and if the desired output port is not available then the flit is either dropped or deflected to any available output port. The dropped flit is again retransmitted by the source by means of necessary acknowledgements. This process led to a lot of overhead in terms of a retransmission and acknowledgements. This reduced the desirability for dropping models. On the other hand the deflected flits eventually reach their desired destination.

## II. BUFFER LESS ROUTING ALGORITHMS

Nilsson et al.[4] first proposed the concept of buffer less deflection routers. The concept of buffer less routing was further enhanced by BLESS [5].

### 2.1 BLESS: Baseline Buffer Less Deflection Routing

Bless deals with a sequential port allocation mechanism. Buffer less routing works better when the network utilization is low. Each node in BLESS contains an injection buffer and a reassembly buffer. Each flit is independently routed through different paths in the network. When there is contention between multiple flits for a particular direction then only one flit is assigned the desired direction and the others are deflected.

Bless cannot be applied to networks with directed links. It can be implemented only in networks where every router is connected to every other router in the network and has equal number of input and output ports. Eg Torus, Mesh, Tress etc. Flits can be injected into the network only if atleast one input port is free. The injection process is purely local and a router is not dependent on other routers. BLESS' arbitration policy is rank based. FLIT-BLESS use a set of port prioritization rules. All flits are timestamped. Oldest flits are assigned their desired ports. Router must sort flits by age with the help of three comparator stages thus ensuring livelock freedom. Overall, deflection routing logic based on Oldest-First has a 43% longer critical path than a buffered router.

Packet reassembly is a must as the flits can arrive in a random fashion. When all the flits of a packet arrive they are delivered to a local node for reassembly. Care has to be taken to prevent overflow of the reassembly buffers Some of the major advantages of BLESS include the avoidance of buffers, purely local and simple flow control, simplicity and router latency reduction, area saving and absence of livelock . But BLESS does not support many functionalities available in buffered networks like QoS and different traffic service classes, fault tolerance in the presence of faulty links and routers and congestion awareness[6].

### 2.2 CHIPPER: Cheap-Interconnect Partially Permuting Router

The major disadvantages in BLESS was overcome in CHIPPER: A Low-complexity Buffer less Deflection Router [7] which introduced three key insights to BLESS 1)deflection routing port allocation maps to a permutation network within the router 2) eliminate expensive age based priorities and introduced a token passing scheme 3) use cache miss buffers as reassembly buffers for flow control.

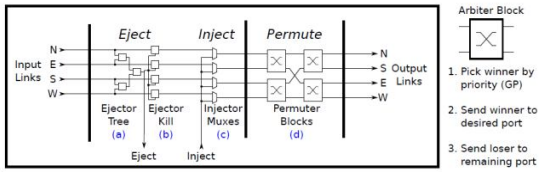


Fig 1: CHIPPER architecture [8]

A new architecture based on permutation network is introduced along with two algorithms 1) Golden Packet and 2) Retransmit once. Priority sort and port allocation are combined in the Permutation network. The permutation network consists of a 2X2 arbiter which first decides the 'Golden flit' through a comparison network and then picks the output port for the winning flit for a given period of time known as the 'Golden epoch'. The Golden Epoch lasts for duration of the circuit diameter number of the clock cycles.All the routers in the network are made aware of this period and they start and stop the golden epoch synchronously. The (source number, packet number) is included in the header and initialised to (0, 0) for the first epoch and then incremented for the subsequent iterations till the end of communication.

Thus CHIPPER guarantees freedom from Livelock and consumes 54.9% less power on an average than buffered networks and 8.8% less than BLESS [7].

| | Buffered | BLESS | CHIPPER |
|---|---|---|---|
| Area | 480174$\mu m^2$ | 311059$\mu m^2$ | 306165$\mu m^2$ |
| Timing(Critical) | 1.88ns | 2.68ns | 1.90ns |

Table 1: Hardware cost comparisons for a single router in a 65nm process

### 2.3 MinBD: Minimally-Buffered Deflection Routing for Energy-Efficient Interconnect

MinBD addresses the drawbacks of BLESS and CHIPPER. For low to medium network load the above mentioned bufferless architecture provides performance at par with buffered networks. But as the network load increases the number of deflections per router increases, reducing performance and efficiency. MinBD introduces a minimally buffered deflection router that combines buffered and buffer less routing techniques.

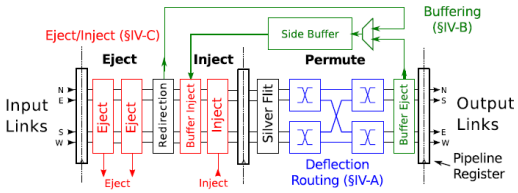


Fig 2: MinBD Router pipeline [8]

MinBD also uses a 4 flit side buffer to store one deflected flit per cycle [8]. This reduces the deflection rate and increases performance.

From the side buffer flits are re injected into the network and thus given a second chance for arbitration. MinBD also introduces the concept of a silver flit in the absence of a golden flit.

MinBD improves performance by 2.7% in a 4X4 network over all workloads. It also has the highest efficiency of almost 42.6% better than the regular input buffered design [8]. However at highest network load MinBD becomes less energy efficient and its efficiency degrades at a very high rate.

### 2.4 DeBAR: Deflection Based Adaptive Router With Minimal Buffering

MinBD introduces the concept of silver flit but the decision of the silver flit is random and local. Therefore there is no guarantee that the flit gets the same silver status in the next router. This may lead to deflection . Moreover the MinBD routers give priority to re injections from side buffer rather than from core buffer. This may lead to starvation[9]. DeBAR uses a minimal set of central buffers to accommodate a fraction of the deflected flits.

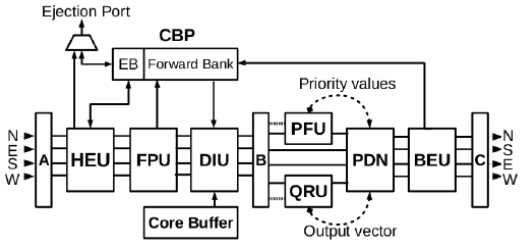


Fig 3: router pipeline for DeBAR.

HEU - Hybrid Ejection Unit, FPU- Flit Preemption unit, DIU-Dual Injection Unit, PFU-Priority Fixer Unit, QRU-Quadrant Routing Unit, PDN- Permutation Deflection Network, BEU-Buffer Ejection Unit, CBP- Central Buffer Pool, A,B and C are the pipeline registers [9]

Four internals channels carry input flits through the router pipeline and at the end of each clock cycle the flits are stored in the corresponding pipeline registers.

Simulation results of an 8x8 mesh network using synthetic traffic patterns shows that DeBAR has a lower average flit latency than MinBD[9] . It has also been observed that as the injection rate increases DeBAR obtains less deflection rate than MinBD [9]. The throughput of DeBAR was also better than MinBD thereby leading to the conclusion that DeBAR is one of the best choices for minimally buffered NoC routers

## III. CONCLUSION

In this paper a survey of four buffer less routers of NoC done and their advantages and disadvantages analysed. Starting from BLESS the first bufferless router, followed by CHIPPER, MinBD and DeBAR which emerged to have the best performance under synthetic traffic.

The conventional NOC hardware do not support any unique multicast handling techniques. Small modifications in the router architecture can support multicast communication which is the need of the hour as highly multithreaded applications increase.

## REFERENCES

[1] W. Dally, “Virtual-channel flow control,” IEEE Transactions on Parallel and Distributed Systems, vol. 3, no. 2, pp. 194–205, 1992.

[2] Y. Hoskote et al., “A 5-GHz mesh interconnect for a teraflops processor,” IEEE Micro, vol. 27, no. 5, pp. 51–61, 2007.

[3] M. B. Taylor et al., “Evaluation of the raw microprocessor: An exposedwire- delay architecture for ILP and streams,” in ISCA, 2004

[4] E. Nilsson et al., “Load distribution with the proximity congestion awareness in a network-on-chip,” in DATE, 2003, pp. 1126–1127

[5] T. Moscibroda and O. Mutlu, “A case for bufferless routing in on-chip networks,” in ISCA, 2009, pp. 196–207.

[6] P. Gratz, B. Grot, and S. W. Keckler. Regional congestion awareness for load balance in networks-on-chip. In HPCA-14, 2008.

[7] C. Fallin, C. Craik, O. Mutlu, “CHIPPER: A Low-complexity Bufferless Deflection Router”, Computer Architecture Lab (CALCM), Carnegie Mellon University, 2011.

[8] C. Fallin et al., “MinBD: Minimally-buffered deflection routing for energy-efficient interconnect,” in NOCS, 2012, pp. 1–10

[9] Jose, B. Nayak, K. Kumar, M. Mutyam. Debar: deflection based adaptive router with minimal buffering, in: Design, Automation and Test in Europe Conference and Exhibition (DATE) IEEE, pp. 1583-1588, 2013.