
Impact of deflection history based priority on adaptive deflection router for mesh NoCs

Elizabeth Isaac*

VIT University,
Vellore, 632001, India
Email: elizabeth.issaac@gmail.co.in

*Corresponding author

M. Rajasekhara Babu

School of Computing Science and Engineering,
VIT University,
Vellore, 632001, India
Email: mrajasekharababu@vit.ac.in

John Jose

Department of Computer Science and Engineering,
Indian Institute of Technology,
Guwahati, 781001, India
Email: johnjose@iitg.ernet.in

Abstract: Network on chip (NoC) has been proposed over bus to address the communicational prerequisite of highly dense multi-core systems. NoCs with buffer-less routers gain popularity due to simplicity in the router design, low power consumption and less chip area. The state of the art of deflection router DeBAR employs side buffers instead of input port buffers, that can accommodate one among the deflected flit per router per cycle. In this paper we propose deflection history as a priority metrics for flit selection. We, modify the primitive DeBAR design, and propose priority based deflection based adaptive router (PBDeBAR) that make use of a cost effective priority scheme to choose a flit that has to be moved to the side buffer. Experimental results shows that PBDeBAR reduces latency, deflection rate, buffer occupancy and link usage with respect to the existing minimally buffered deflection routers.

Keywords: buffer-less routing; buffer occupancy; congestion; deflection; link activity; minimally buffered; penalisation; router pipeline; efficiency; side buffer.

Reference to this paper should be made as follows: Isaac, E., Babu, M.R. and Jose, J. (2017) 'Impact of deflection history based priority on adaptive deflection router for mesh NoCs', *Electronic Government, An International Journal*, Vol. 13, No. 4, pp.391–407.

Biographical notes: Elizabeth Isaac is an Assistant Professor at M A College of Engineering Kothamangalam, Kochi, Kerala. She is a postgraduate in Computer Science and Engineering from VIT University. She did her graduation in Computer Science and Engineering from M A College of Engineering Kothamangalam. She is currently a PhD Scholar in the School of Computing Science and Engineering, VIT University, Vellore.

M. Rajasekhara Babu is a Senior Faculty Member at School of Computing Sciences, VIT University, Vellore, India. He completed his PhD from VIT University. He receives his Bachelors in Electronics and Communication Engineering from Sri Venkateswara University, Tirupathi, India and took his Masters in Computer Science and Engineering from Regional Engineering College (NIT), Calicut. His areas of interest include multi core architectures, compilers.

John Jose is an Assistant Professor at Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Guwahati, India. He did his BTech from Cochin University and MTech from VIT University, Vellore. He completed his PhD from Indian Institute of Technology, Madras. His areas of interest include computer architecture, interconnection networks and high performance computing.

This paper is a revised and expanded version of a paper entitled 'DeBAR: deflection based adaptive router with minimal buffering' presented at *Design, Automation and Test in Europe Conference and Exhibition (DATE)*, IEEE, Europe, 2013.

1 Introduction

With the advancement in VLSI technology multiple processing cores can be integrated on a single chip. Multiple cores significantly improve the overall system performance without increasing the operating clock frequency. Unpredictably, the bus dependent communicational prerequisite cannot scale as the number of processor cores ascent. Network on chip (NoC) take over the bus interconnections with an all together enthralled approach to meet with the communicational necessities of modern multicore systems (Dally and Towles, 2001).

Traditional NoC based multicore consists of an array of processing cores that are connected by a network of well structured point-to-point bidirectional links to the routers. 2D-mesh is a commonly preferred topology for such systems as it significantly reduces the design cost. In a mesh topology, each router is connected to an adjacent router located at North, South, East and West directions. Each router is pipelined and takes two or three cycles to forward the packet to the next router. Wormhole packet switching (Dally, 1992; Smai and Thorelli, 1998) is used to forward the packets through the routers. Flits are considered as the smallest indivisible unit of a packet.

Dispatch of packet to its destination depends on routing logic dwelling in the NoC router. Routing algorithms are not only Deterministic, but can also be non-deterministic like Oblivious and Adaptive. In deterministic algorithm, a rigid path is established between

the source-sink pair. But in oblivious routing algorithms, a route is chosen from convenient multiple routes without considering the network state. In adaptive algorithm, the network feasibility is taken into account from the multiple possible path between the source and destination (Dahir et al., 2013; Azampanah et al., 2013).

Typically an NoC router consists of buffers in the input ports for the flits to reside. In buffered routing, if one or more flits compete for the same output port, the winning flit continue its journey through the assigned output port while the losing flits stay back in the respective buffers. Router buffers accommodate the flits until they get a productive output port. Flits get forwarded to the down stream router only if a free buffer space is available at the down stream router.

The input buffers helps in effective bandwidth utilisation by decoupling storage resources from transmission resources (Jose et al., 2013). Even though buffers improve transmission bandwidth they increase both on-chip network area and power. Studies show that these input buffers dissipate 22% of the router power (Vangal et al., 2008) and consume 75% of NoC area (Gratz et al., 2006). In addition to this the buffered routing adds supplementary control logic to the router design to keep record of the migration of the flits in and out of the buffers.

Buffer-less routing is a promising, cost effective alternative for power consuming input buffered NoCs. Buffer-less NoCs are designed to achieve less area and power consumption by compromising on the peak network throughput. Contention happen when two or more flits request for the same output port. Since there are no buffers in routers, once contention arises, the router decides either to drop (Gomez et al., 2008) or to deflect (Jose et al., 2013; Dally and Towles, 2003; Moscibroda and Mutlu, 2009; Fallin et al., 2011) the selected flits. Deflection routing works on the principle that all the incoming its are passed to one of the available ports without considering whether the port is productive or not. Single flit packets with necessary header information is the upcoming standard in buffer-less routing (Fallin et al., 2011).

We propose a priority based deflection based adaptive router (PBDeBAR) which is an enhancement of our previous work, DeBAR: deflection based adaptive router (Jose et al., 2013). PBDeBAR uses a priority scheme based on the deflection count of flits. Our experiments on 8×8 mesh network with synthetic traffic patterns (Dally and Towles, 2003) and SPEC 2006 CPU benchmark mixes relegated as real workload (<http://www.spec.org>) reveal that PBDeBAR perform superior than DeBAR in terms of a latency, deflection rate, buffer occupancy and link usage.

2 Buffer-less deflection routers: an overview

Buffer-less routers are gaining popularity and are preferred over buffered routers for larger NoCs as buffers are power hungry, consume large chip area and buffer management circuits are complex. Automatic flow control (AFC) (Jafri et al., 2010) is a hybrid approach that switch between the buffered and buffer-less mode based on network load by using the power gating technique. The flexi-buffer (Kim et al., 2011) design uses fine grained power gating thereby adjusting the size of the active buffers adaptively. In both these techniques the chip area remains due to the presence of the buffers.

The central and the ring deflection algorithms proposed in Oxman et al. (2012) use sequential port allocation techniques, which increase the router critical paths. The ring algorithm deflect flits away from the centre of the mesh thereby reducing the formation

of hotspots. An exhaustive study on the congestion issues in buffer-less NoCs on system performance at both the network and application levels is done in Nychis et al. (2010). Various design parameters of the buffered and the buffer-less NoCs are discussed in Lu et al. (2006).

The buffer-less deflection routing gained importance with the introduction of the 2-stage BLESS (Moscibroda and Mutlu, 2009) router micro architecture. In BLESS routing, flit ranking and port prioritisation is done in the first stage of the router pipeline. Second stage takes care of port allocation. All flits will be designated with an output port based on the age priority of flits. This demands few bits in the flit header to store the flit age. In a cycle, in every router, a maximum of four flits can enter the router pipeline. When the flit is locally destined, it moves through the ejection port. After the port allocation every flits in the router gets an output port. Those flits which get their desired port are called productively assigned flits and the others are called deflected flits. In BLESS, a sequential port allocation circuit using a crossbar is used. This increases the critical path delay of the router pipeline.

CHIPPER (Fallin et al., 2011) is a buffer-less router that employs a parallel port allocation scheme to compensate the pipeline delay of BLESS. CHIPPER uses a golden priority scheme and ensures that the golden flit (there will be only one golden flit in the network at a time) is not deviated. The golden flit scheme used for flit prioritisation does not ensure 100% livelock freedom and progress. The golden flit scheme is very simple since a flit is chosen randomly and is globally prioritised over all the other flits. The permutation deflection network (PDN) in CHIPPER, considerably reduces the critical path delay at the expense of increased deflection rate. The PDN is a two-stage arbitration circuit that performs parallel allocation of output ports. Each PDN consists of four 2×2 arbiters. Each arbiter takes two of the incoming flits and identifies the highest priority incoming flit and assign it with desired output port. In the mean time, the other flit gets the remaining output port.

Buffer-less deflection routers experience high deflection rate at higher injection rate (Moscibroda and Mutlu, 2009; Fallin et al., 2011). To address this performance issue, MinBD (Fallin et al., 2012) makes use of a minimal side buffering technique that stores one among the deflected flits per cycle in a buffer. In addition to golden packet scheme a silver flit is also randomly chosen for each router. Since the silver flit status is not propagated to the neighbouring routers, MinBD cannot guarantee the timely progress of flits towards the destination. minimally buffered single-cycle deflection router (MinBSD) (Jonna et al., 2014) diminishes the critical path lag. The number of pipeline registers is reduced to two thereby making the router to operate in a single cycle. In MinBSD injection from the side buffer and the core buffer happens in all cycles with the help of the 3×2 arbiter.

The ultimate stage in a smart late injection deflection router (SLIDER) (Bhawna et al., 2013) router pipeline is the injection, hence it is named as late injection. Injection occurs depending on the buffer occupancy level, thereby making it to work on 2-modes- Restricted Injection, Non-Restricted Injection. Restricted Injection occurs only when a productive port is available for the buffered flits. About three fourth of the time the SLIDER operates under the Restricted Injection. Non-Restricted Injection happens when the buffer is partially full.

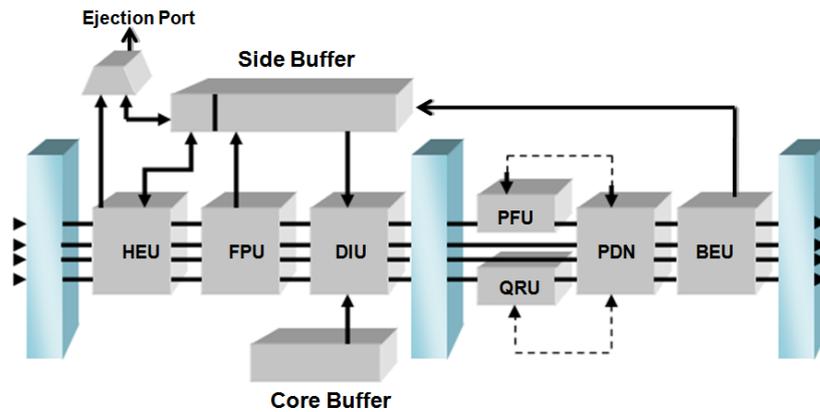
DeBAR (Jose et al., 2013) improves the performance by addressing the limitations of MinBD. DeBAR is considered as the best side buffered deflection router in terms of efficiency and performance at moderate network traffic. SLIDER outperforms DeBAR by virtue of exploiting the concept of Late injection. We focus our attention only in improving the performance of traditional DeBAR architecture by cost effective optimisations on

existing units in the DeBAR pipeline. Identifying few limitations in DeBAR design we put forth a couple of economical mechanisms to enhance the performance of DeBAR. Thus a global fairness priority mechanism can be used to maintain fairness in a network there by improving the performance (Hanmin and Kiyong, 2016). We describe the architecture of DeBAR in the next section, which will help the reader to appreciate the limitations identified by us in the Section 4.

3 DeBAR architecture

The pipeline router architecture of DeBAR is shown in the Figure 1. DeBAR is a 2-stage deflection router which uses a side buffer to accommodate a fraction of misrouted flits. At the beginning of the clock cycle, the flits reaches the hybrid ejection unit (HEU) from the pipeline register A. HEU takes care of the ejection of two flits at a time. In dual injection unit (DIU) not all side buffer but also the core buffer progressively inject the packets in alternate cycle. If all the pipeline links are busy, in order to avoid the never-ending waiting of the flits dwelling not only in the core but also in the side buffer, a forced removal using random metrics is employed by flit preemption unit (FPU). Computation of flit priority and output port is done in the second stage by priority fixer unit (PFU) and quadrant routing unit (QRU), respectively. Output port allocation is carried out by the PDN based on the priority and route obtained from the PFU and QRU. One of the flits is randomly shifted by the buffer eject unit (BEU) to the side-buffer.

Figure 1 DeBAR architecture (see online version for colours)



4 Motivation

DeBAR, 2-stage router make use of the principle of side buffering. Side buffer helps to reduce the deflection rate by storing one of the deviated flits in each and every router. DeBAR employs the following two units to preempt the flits to the side buffer.

- *FPU*: To overcome the starvation of flits not only in the side buffer but also in core buffer, FPU allow the flit to be preempted to the side buffer.
- *BEU*: Non-productive port assign by the PDN force the BEU to move one of the flits to the side buffer.

Identified limitations of DeBAR designs are discussed below.

4.1 Penalisation of deflected flits by preemption

The FPU unit in the DeBAR makes use of random preemption logic to move a flit to the side buffer from the router pipeline which makes an idle channel for flit injection through DIU. When we consider a condition in DeBAR, the four internal channels of a router even though busy they are not designed for ejection. Hence the side buffer cannot re-inject its flit into the router pipeline. To prevent the situation of ceaseless waiting of the flits residing in both the buffers, a flit is preempted randomly by FPU to the side buffer. A random selection the flit from the router pipeline can move a heavily deflected flit (in the past) to the side buffer thereby penalising the flit again.

4.2 Penalisation of flits by repeated side buffering

The BEU unit of DeBAR classifies the flits into deflected and non-deflected flits. As mentioned earlier the side buffering reduces the deflection rate. But the BEU moves the random flit from the deflected group to the side buffer. When buffered flits are reinstated, there is a high chance that they may be buffered to the side again after allocating the flits to the port by PDN. Flit's priority is not affected even when they are reinstated to the router's pipeline framework. This is because DeBAR employs a hops-to-destination priority scheme which give high priority to the flits with least hops-to-destination. The hops-to-destination is not changed for a re-injected flit. Hence the penalised (by deflection) flits are penalised again (by side buffering).

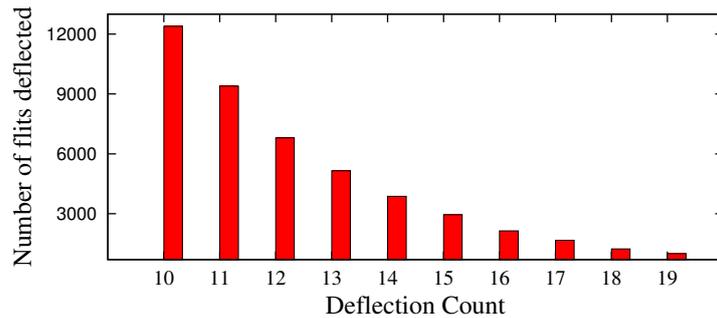
The impact of random selection of flit on the performance of the network can be explained using the deflection count (DC). DC can be explained as the total number of deflections incurred during a flit's journey towards its destination. Figure 2 shows the frequency of Deflection count of flits in an 8×8 mesh NoC to show the uniform traffic behaviour before saturated. There exists a linear relationship between the Injection rate and latency and the point at which this latency shoots up can be called pre-saturation injection rate. In the graph we plot only the count of flits that have encountered more than 10 and less than 20 deflections. The count of flits with less than 10 deflection (not plotted in the graph) is very high. We observe that 24% of the injected flits get deflected more than 5 times out of which about 7% get deflected more than 10 times. From this statistics, we can infer that there are flits that suffer heavy deflection in the network. This leads to the starvation of such flits and increase in the average latency. So we propose that the flits that are heavily deflected should be given extra care so that they do not suffer more deflection.

4.3 Penalisation of preempted flits by side buffering

In order to avoid the starvation of flits in both the side and core buffers a flit is randomly preempted from FPU to side buffer. The re-injected flits also take an active participation in the process of arbitration for gaining the output ports. Failure to win the output port during

arbitration results in the subsequent buffering in the side buffer by BEU. So the re-injected flits ultimately returns to the side buffer, when it is assigned an unfruitful output port. We identify 22% of such cases of unnecessary internal flit movement in uniform traffic in 8×8 mesh network at presaturation load. The movement of flits from buffer to buffer leads to the unneeded power usage without any forward progress of the flits.

Figure 2 Frequency of Deflection count of flits in an 8×8 mesh NoC to show the uniform traffic behavior before saturated (see online version for colours)

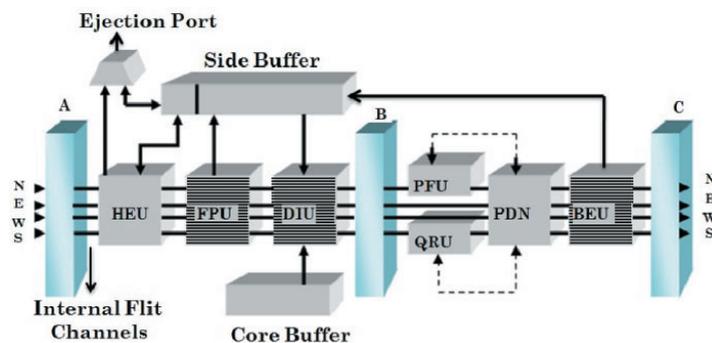


5 PBDeBAR architecture

The basic working of the PBDeBAR is same as that of the DeBAR. PBDeBAR contains a few additional logical modules that improves the performance. The shaded region in the Figure 3 shows the three impact points in the DeBAR architecture that we modify to obtain the PBDeBAR design. The basic differences between DeBAR and PBDeBAR

- choice of metric used for the selection of flits at two points in the router pipeline(FPU and BEU)
- in the manipulation of priority in DIU.

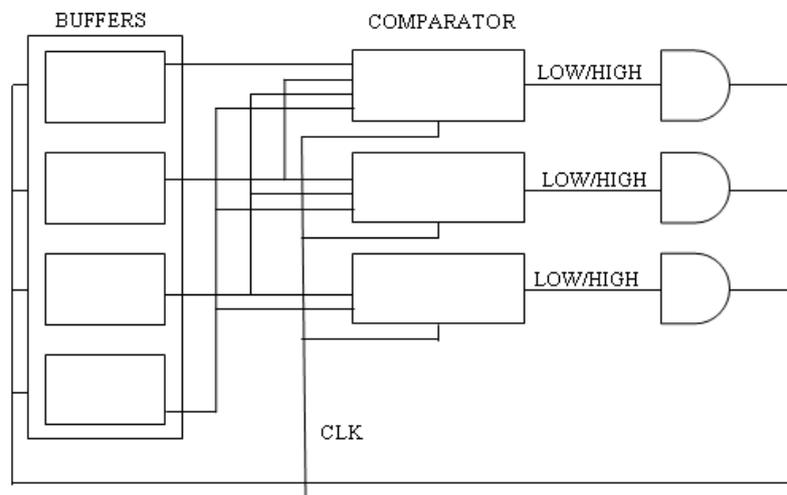
Figure 3 PBDeBAR architecture (see online version for colours)



5.1 Deflection computation unit (DCU)

We include a 5-bit DC field in flit header. DCU is a light weight logical sub-module in PBDeBAR. DCU is added in FPU and BEU, which computes the deflection history of all the flits in the router pipeline. This computed value is used in the priority scheme to move a flit to the side buffer. Whenever a flit gets deflected the deflection count is incremented. The role of DCU in PBDeBAR is to assign a priority to all the incoming flits in the pipeline router based on the DC. The detailed architecture of DCU unit is shown in Figure 4. The flits are compared and the lowest/highest priority is stored in the top buffer through the feedback mechanism.

Figure 4 DCU unit



Along with this we suggest a priority enhancement for the re-injected flits. We summarise these three policies as follows

- preemption policy
- buffer ejection policy
- re-injection policy.

5.2 Preemption policy

We modify the preemption policy to ensure that a heavily deflected flit is not preempted. DCU extract the DC from the flit header and the flit with the least DC is moved to the side buffer. Thus the random selection of flit in FPU of DeBAR is replaced with a priority scheme based on deflection count.

5.3 Buffer ejection policy

Buffer ejection policy ensures that a heavily deflected flit are not allowed to deflect again. We give them a chance to escape from the deflection by side buffering. We propose another

DCU sub-module unit in the BEU as well. The parameter to keep a flit in the side buffer is decided by the number of deflections acquired during its journey. Thus the flits with non-productive port is moved to a buffer in the side. It is to ensure that the flit's latency is not increased due to a 3-hop deflection (A flit which is deflected needs min 3-hops to come back to original path). In the buffer ejection unit (BEU) of the router pipeline, the flits that are marked with 1 are assigned with non-productive port. These flits get redirected away from the destination router. The priority is given to the packet that was deflected more number of times in the past. Flits that are not buffered comes out through the assigned output port.

5.4 *Re-injection policy*

The highest priority flits always gain the productive output port in PDN. The rest of the flits can be deflected through non-productive output port or redirected to side buffer. In PBDeBAR the PFU assigns highest priority to the side buffered flits. Rest of the flits (flits that reach FPU from the initial units of the router pipeline) are sorted based on the hops to destination as in DeBAR. In PBDeBAR the priority level 0 is reserved for the re-injected flits from the side buffer. As stated above the heavily deflected flit moves to the side buffer from the BEU. The enhanced priority scheme ensures the progress of the re-injected flits by assigning a productive port to it.

6 **Experimental methodology**

Booksim (Jiang et al., 2013), a cycle driven tool is used to design an 8×8 mesh NoC system. This set up consist of 64-routers each of which is connected to packet generation core. In traditional Booksim, each of this router is designed as a 2-cycle input buffered switch in sufficient detail and accuracy. The traditional router pipeline of NoC router which consist of routing unit, switch allocator, virtual channel allocator and crossbar is replaced by a set of five units (FPU, BEU, PFU, PDN and DEU) organised as 2-stage pipeline as mentioned before. The flit channel has a definite parameter, with data field of 128-bit and header field of 12-bit. We reconfigure the baseline Booksim simulator to compare the PBDeBAR with DeBAR design for conducting experimental analysis.

6.1 *Traffic patterns and workloads*

To analyse the performance of the routers necessary traffic has to be induced into the network. Traffic patterns are classified into synthetic and real. Synthetic traffic patterns are abstract models of message passing in NoCs, but realistic traffic patterns are generated from trace of real applications running on NoC based MPSoCs.

Uniform, tornado and bit-complement are synthetic patterns which can be used as a touch stone strategies to measure the performance improvement of PBDeBAR over the predominant existing works for 8×8 mesh network.

The traffic patterns generated by real workloads are also put-forth to evaluate the performance of our proposed technique. We employ a Multi2sim (Ubal et al., 2007) simulator for modelling a 64-core multiprocessor with CPU cores, protocol used and cache hierarchy with enough precision and characteristic. All of the cores contain a dedicated 64 KB L1-cache and 512 KB shared L2-cache. L1 and L2 are 4-way and 16-way set

associative respectively. Each core is assigned with an application to run on it. Based on the *misses per kilo instructions* (MPKI) values are calculated by running SPEC 2006 CPU benchmarking strategies on all cores. Based upon the results we can categorise the benchmarks into low, medium and high. Taking into consideration the network load, we compartmentalised workload into mixes from W1 to W7. After progressing the execution in an ambient manner, we take into account the cache miss details which is then inculcated to the Booksim to obtain the behaviour similar to the real environment. Table 1 shows the benchmark mixes for measuring the percentage of various network injections. Some of the standards under considerations are *leslie3d*, *bwaves*, *calculix*, etc.

Table 1 Benchmark mixes for measuring the percentage of intensity of various network injections

<i>Workload mix</i>	<i>W1</i>	<i>W2</i>	<i>W3</i>	<i>W4</i>	<i>W5</i>	<i>W6</i>	<i>W7</i>
% of Low-MPKI	100	0	0	50	0	50	31
% of Medium-MPKI	0	100	0	50	50	0	31
% of High-MPKI	0	0	100	0	50	50	38

7 Experimental analysis

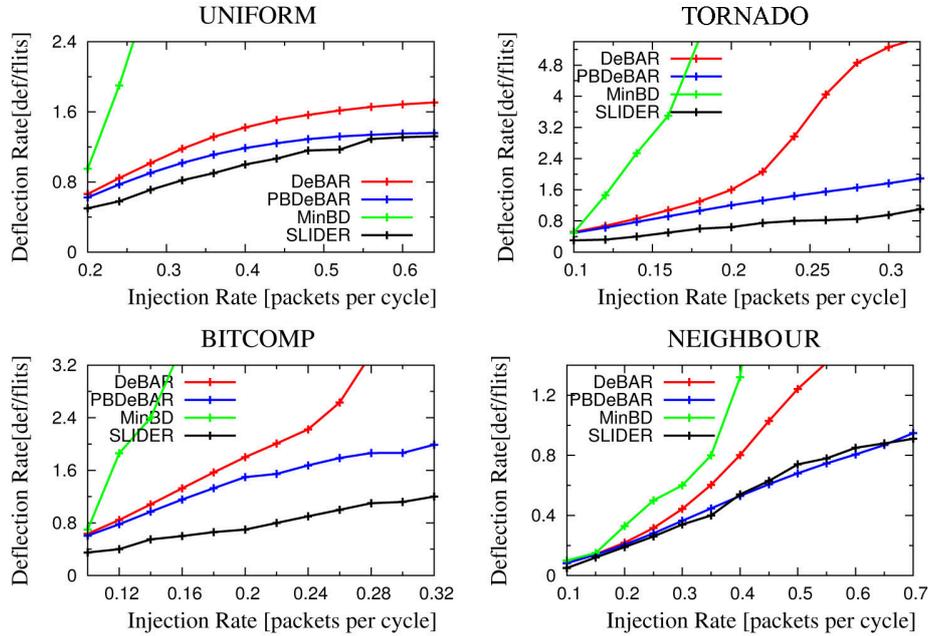
We compare the performance of PBDeBAR with MinBD, DeBAR and SLIDER. PBDeBAR in conformity with the DeBAR is examined to study the influence of upgraded priority scheme on standard network parameters. PBDeBAR is also compared with the SLIDER.

7.1 Effect on deflection rate

Average count of deflections per flit can be explained as the deflection rate. DeBAR prevents the deflection of flits once they are near to the destination. In addition to this priority scheme used in the DeBAR, PBDeBAR helps the heavily deflected flits in reaching their destination faster. This also contributes to reduced deflection rate in PBDeBAR.

Effect of varying injection rate on the deflections can be deduced from the graph shown in Figure 5. This figure shows a set of deflection rate plots for MinBD, DeBAR, SLIDER and PBDeBAR. From the graphs we can infer that PBDeBAR achieves lower deflection rate related to DeBAR and MinBD and is very close to that of SLIDER. At low injection rate the deflection rate is more or less same for DeBAR and PBDeBAR. This is because the count of heavily deflected flit is very less and our proposed priority enhancement will not affect the common case performance. We observe that at low injection rate the count of its that are going to side buffer is very less (low injection rate, less port contention, less non-productive its, fewer number of its to side buffer). Thus the case of priority level-0 rarely happens at low injection rate.

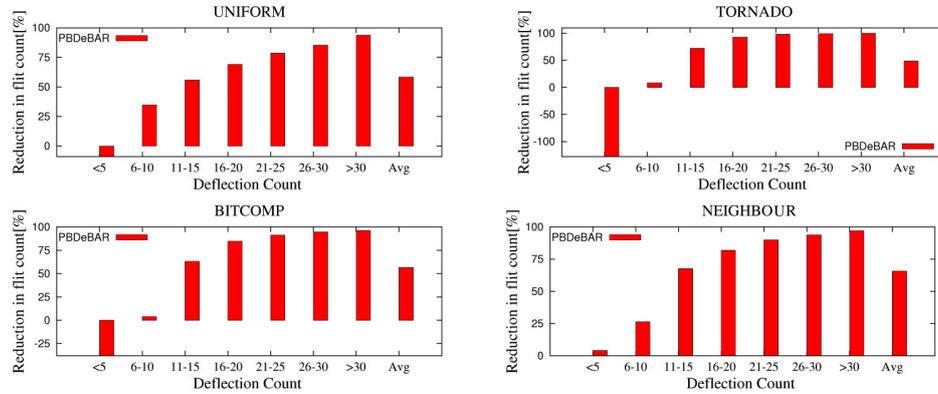
But at high injection rate, significant number of flits enter side buffer and hence cases with level-0 priority increases substantially and it will leads to reduction in deflection rate. We experimentally observe this phenomena and confirmed that our enhanced priority mechanisms are playing a critical role in the reduction of average flit deflection rate.

Figure 5 Comparative analysis of deflection rate verses injection rate for various synthetic traffic patterns in 8×8 mesh network (see online version for colours)

7.2 Effect on deflection count

From Figure 5, we have already seen a reduction in deflection rate. To get more intuition about the reduction in the deflection rate, we analyse each flit and categorised the flit into one of the deflection count class. We formed seven classes of flits based on DC. *Class-1* ($DC \leq 5$), *Class-2* ($6 \leq DC \leq 10$), *Class-3* ($11 \leq DC \leq 15$), *Class-4* ($16 \leq DC \leq 20$), *Class-5* ($21 \leq DC \leq 25$), *Class-6* ($26 \leq DC \leq 30$) and *Class-7* ($DC \geq 30$). For a flit traffic of two lakhs, for both DeBAR and PBDeAR we estimated the count of flits belong to each of the above mentioned class. We find the percentage reduction of flit count in each class with respect to DeBAR on an 8×8 mesh network using the synthetic traffic patterns and plot the same in Figure 6.

A flit reaches the destination after passing through many routers in its path. It can get deflected in many intermediate routers. The number of times the flit gets deflected is recorded in DC field of the flit header. In DeBAR, it is observed that, at high network load the deflection occurs more frequently, thus reducing the performance and efficiency of the system. In PBDeBAR since we employ the enhanced priority scheme, it reduces the deflections of flits. It is observed that the priority scheme we employed reduces the number of deflections in the higher classes at the expense of increased deflection count in *Class-1*. Hence the average reduction was negative in *class-1*, there by emphasising that by using PBDeBAR the number of flits in *Class-1* has increased substantially whereas we see a reduction in all other classes.

Figure 6 Percentage reduction in the DC for various deflection count class w.r.t. DeBAR for various synthetic traffic patterns in 8×8 mesh network (see online version for colours)

7.3 Effect on average buffer occupancy

Buffer occupancy of a flit is defined as the total number of cycles spend in the core buffer of the source router and the side buffer of all intermediate routers. Average buffer occupancy B_o is given by,

$$B_o = \frac{\sum_{i=1}^N (sb_i + cb_i)}{N}, \quad (1)$$

where sb_i is the total number of cycles a flit stayed in side-buffer before it reaches its destination, cb_i is the total number of cycles a flit stayed in the core buffer before it is injected into the network and N is the total number of flits injected into the network.

We can see from the Figure 7 that the PBDeBAR design reduce the average buffer occupancy of the flits significantly for all traffic patterns. The average buffer occupancy is more or less same at lower injection rate. This is because the port conflict is less and hence the count of flits that are going to the side buffer is less. But as injection rate increases, more number of flits are forced to stay in the side buffer. But the improved priority scheme give the highest priority to the re-injected flits and such flits will not go to side buffer of the same router.

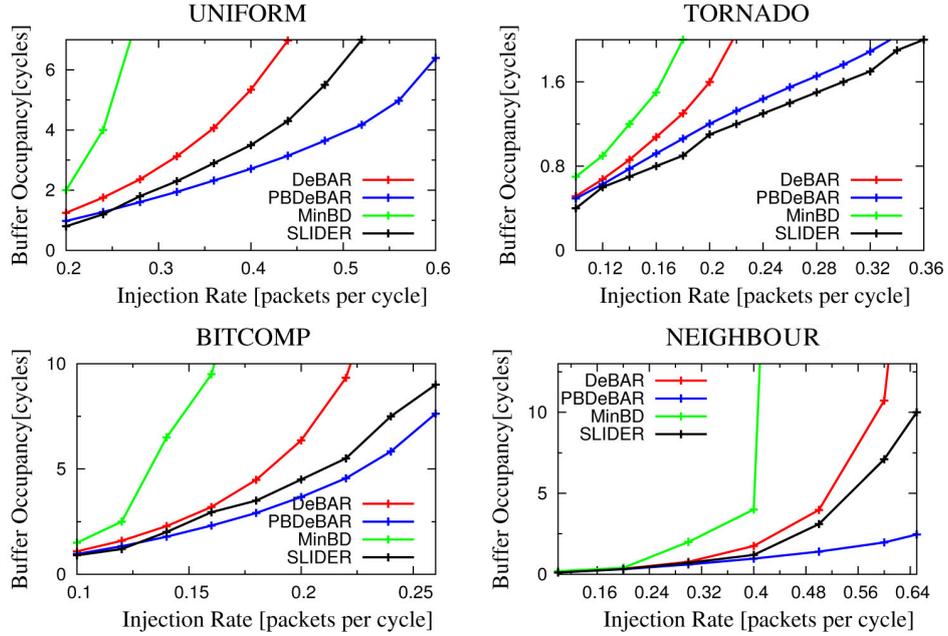
Since SLIDER hold the flits in the buffer till it is assigned productive ports, flits are having minimum deflection rate. But latency of SLIDER and PBDeBAR almost same. This is because SLIDER have more buffer stay occupancy time due to longer stay in buffers till it gets productive port. We are not claiming better performance than SLIDER in all traffics. SLIDER all together uses a different technique called Late Injection, which utilises output channels in a more productive way.

7.4 Effect average flit latency

A flit's latency can be described as the time just needed to travel over the network from its origin to its end. From Figure 8 we can inspect the effect of the injection rate on the

flit latency in an 8×8 mesh network. For linear increase in injection rate the latency also ascent and reaches a breakthrough point and that specific location is called saturation point.

Figure 7 Comparative analysis of average flit buffer occupancy vs. injection rate for various synthetic traffic patterns in 8×8 mesh network (see online version for colours)



It is observed that for every traffic patterns the PBDeBAR saturates at high injection rate than DeBAR and MinBD. This indicate that the PBDeBAR is capable of working at high injection rate. As traffic increases, increased contention causes latency to increase and the packets have to wait in the side buffer. PBDeBAR employs a priority scheme that choose the right flit for the side buffer. Thus average latency of the network gets reduced.

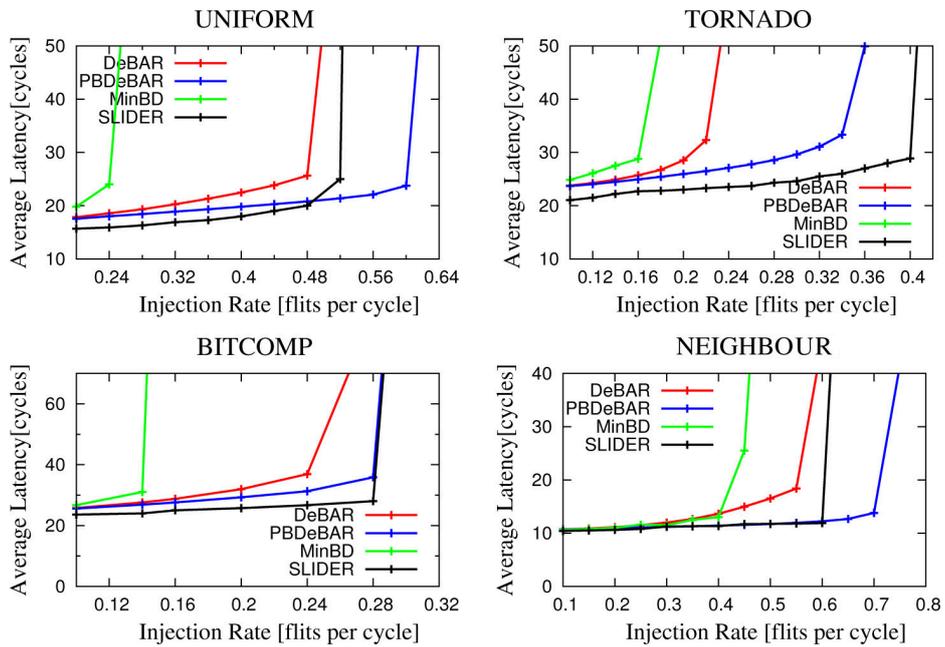
7.5 Effect on critical path

We synthesised the verilog model of DeBAR and PBDeBAR using synopsis design compiler using 45 nm CMOS library to obtain the pipeline delay of the designs. We found that the flit selection module that checks the deflection history and change of priority to '00' at the re-injection point is not changing the critical path. Since there is no change in the critical path with respect to DeBAR, PBDeBAR can operate at the same frequency as DeBAR.

But the additional logical modules consumes 2.4% additional router power and 2.7% area than traditional DeBAR design. The power and area estimates of PBDeBAR with respect to DeBAR is obtained by using Orion 2.0 (Kahnq et al., 2009; Masud et al., 2009). Due to additional number of bits in header for DC 3.5% channel wiring overhead is incurred.

In all the graphs we can see that all the deflection router whether it is MinBD, DeBAR reduces its performance (exponential increase in latency, very high deflection rate, very high buffer occupancy) beyond a specific injection rate called the saturation point. Even Because of this minimally deflection router are suitable for application whose Average Injection Rate is well within the saturation point identified by as in this work But at very low injection rate all the works are having more or less the same performance. Throughout our analysis PBDDeBAR slightly outperforms SLIDER and the conventional DeBAR at rate near to saturation point. Considering the throughput the proposed work show same performance with all other works.

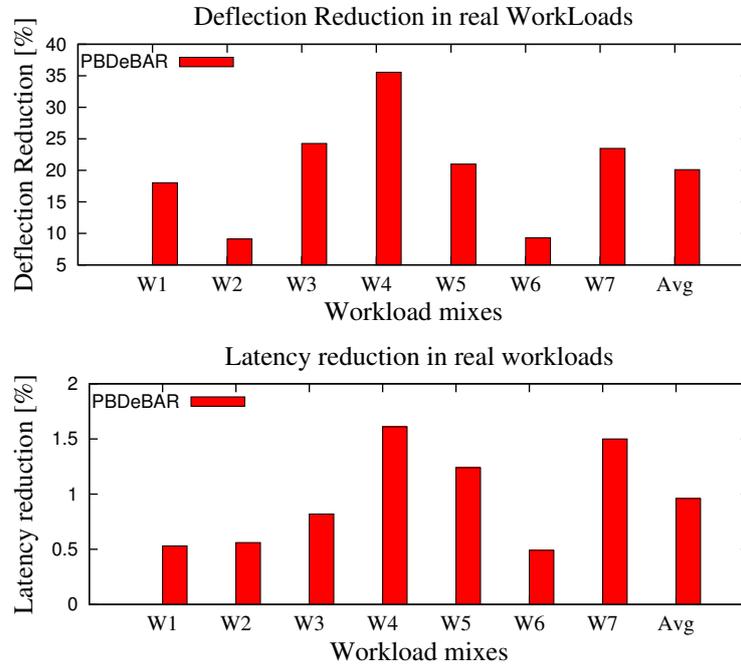
Figure 8 Comparative analysis of average packet latency vs. injection rate for various synthetic traffic patterns in 8×8 mesh network (see online version for colours)



7.6 Effect on real workloads

In the deflection rate analysis of the real workloads in the Figure 9, we can see that PBDDeBAR outperforms DeBAR for all mixes. In workloads with High MPKI applications (W3, W5, and W7), the reduction in deflection rate is around 20–25% whereas the Low MPKI application W4 shows around 35% reduction in deflection rate. High deflection count has high impact on the network latency. We rerouted the flits so as to distribute the load among the lower deflection count flits. With reference to DeBAR the Figure 9 shows the percentage degradation of PBDDeBAR for all diversified workloads. For all mixes, we can see the reduction in the average flit latency using PBDDeBAR design. The latency reduction is more for mixes W4, W5 and W7.

Figure 9 Percentage reduction with respect to DeBAR for real traffic (see online version for colours)



8 Conclusion

Through this paper we emphasise the use of side buffers for the buffer-less routing. We also proposed few logical addition to the baseline DeBAR architecture. Injection of newly generated flits and re-injection of buffered and preempted flits are coordinated in an effective manner with better priority metrics. PBDDeBAR architecture stands above the DeBAR architecture in terms of better overall average flit latency, deflection count, average buffer occupancy, deflection rate, link activity and throughput. All these design optimisations make PBDDeBAR an excellent choice for minimally buffered NoC routers. We conclude that PBDDeBAR is an optimal solution to work on high network load and can bring about bountiful betterment in the network behaviour.

References

- Azampanah, S., Khademzadeh, A., Bagherzadeh, N., Janidarmian, M. and Shojaei, R. (2013) 'Contention-aware selection strategy for application-specific network-on-chip', *IET Computers and Digital Techniques*, Vol. 7, No. 3, pp.105–114.
- Bhawna, N., John, J. and Madhu, M. (2013) 'SLIDER: smart late injection deflection router for mesh NoCs', *Proceedings of International Conference on Intelligent Computing, Communication and Devices*, North Carolina, pp.377–383.

- Dahir, N., Mak, T., Al-Dujaily, R. and Yakovlev, A. (2013) 'Highly adaptive and deadlock-free routing for three-dimensional networks-on-chip', *IET Computers and Digital Techniques*, Vol. 7, No. 6, pp.255–263.
- Dally, W. and Towles, B. (2001) 'Route packets, not wires: on chip interconnection networks', *Proceedings of the 38th Conference on Design Automation (DAC '01)*, Las Vegas, NV, USA, pp.684–689.
- Dally, W. and Towles, B. (2003) *Principles and Practices of Interconnection Networks*, Morgan Kaufmann Publishers Inc., USA.
- Dally, W. (1992) 'Virtual-Channel flow control', *IEEE Transactions on Parallel and Distributed Systems*, Vol. 3, No. 2, pp.194–205.
- Fallin, C., Craik, C. and Mutlu, O. (2011) 'CHIPPER: a low complexity bufferless deflection router', *Proceedings of the International Symposium on High Performance Computer Architecture*, San Antonio, TX, USA, pp.144–155.
- Fallin, C., Nazario, G., Yu, X., Chang, K., Ausavarungnirun, R. and Mutlu, O. (2012) 'MinBD: minimally-buffered deflection routing for energy-efficient interconnect', *Proceedings of the International Symposium on Networks-on-Chip*, Copenhagen, Denmark, pp.1–10.
- Gomez, C., Gomez, M.E., Lopez, P. and Duato, J. (2008) 'Reducing packet dropping in a bufferless NoC', *Proceedings of the 14th International Conference on Parallel Processing -Euro-Par*, Las Palmas de Gran, Spain, pp.899–909.
- Gratz, P., Kim, C., McDonald, R., Keckler, W. and Burger, D. (2006) 'Implementation and evaluation of on-chip network architectures', *Proceedings of International Conference on Computer Design*, San Jose, CA, USA, pp.477–484.
- Hanmin, P. and Kiyong, C. (2016) 'Adaptively weighted round-robin arbitration for equality of service in a many-core network-on-chip', *IET Computers and Digital Techniques*, Vol. 10, No. 1, pp.37–44.
- Jafri, S., Hong, Y., Thottethodi, M. and Vijaykumar, T.N. (2010) 'Adaptive flow control for robust performance and energy', *Proceedings International Symposium on Computer Architecture and High Performance Computing*, Atlanta, GA, USA, pp.433–444.
- Jiang, N., Becker, U.D., Michelogiannakis, G., Balfour, J., Towles, B., Kim, J. and Dally, J.W. (2013) 'A detailed and flexible cycle-accurate network-on-chip simulator', *Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software*, Austin pp.86–96.
- Jonna, G.R., Jose, J., Radhakrishnan, R. and Mutyam, M. (2014) 'Minimally buffered single-cycle deflection router', *Proceedings of International Conference on DATE- Design Automation and Test in Europe*, pp.1–4.
- Jose, J., Nayak, B., Kumar, K. and Mutyam, M. (2013) 'DeBAR: deflection based adaptive router with minimal buffering', *Proceedings of ACM International Conference on Design, Automation and Test*, Europe, pp.1583–1588.
- Kahng, A.B., Li, B., Peh, L. and Samadi, K. (2009) 'Orion 2.0: a fast and accurate NoC power and area model for early stage design space exploration', *DATE*, Nice, France, pp.423–429.
- Kim, G., Kim, J. and Yoo, S. (2011) 'FlexiBuffer: reducing leakage power in on-chip network routers', *Proceedings of the Design Automation Conference*, New York, NY, USA, pp.936–941.
- Lu, Z., Zhong, M. and Jantsch, A. (2006) 'Evaluation of on-chip networks using deflection routing', *GLSVLSI-Great Lakes Symposium VLSI06*, Philadelphia, PA, USA, pp.296–301.
- Masud, A.A., Khan, S.U., Loukopoulos, T., Bouvry, P., Li, H. and Li, J. (2010) 'An overview of achieving energy efficiency in on-chip networks', *Int. J. Communication Networks and Distributed Systems*, Vol. 5, No. 4, pp.444–458.
- Moscibroda, T. and Mutlu, O. (2009) 'A case for bufferless routing in on-chip networks', *Proceedings of the Annual International Symposium on Computer Architecture*, Austin, TX, USA, pp.196–207.

- Nychis, G., Fallin, C., Moscibroda, T. and Muthlu, O. (2010) 'Next generation on-chip networks: What kind of congestion control do we need?', *Hotnets-IX Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, Article No. 12, Monterey, California.
- Oxman, G., Weiss, S. and Birk, Y.T. (2012) 'Streamlined network-on-chip for multicore embedded architectures', *Proceedings of the International Conference on Architecture of Computing Systems*, Germany, pp.238–249.
- Smai, A. and Thorelli, L. (1998) 'Global reactive congestion control in multicomputer networks', *Proceedings of the 5th International Conference on High Performance Computing*, Nara, Japan, pp.179–186.
- Ubal, R., Sahuquillo, J., Petiti, S. and Lopez, P. (2007) 'Multi2sim: a simulation framework to evaluate multicore-multithreaded processors', *Proceedings of the International Symposium on Computer Architecture and High Performance Computing*, Brazil, pp.62–68.
- Vangal, S.R., Howard, J., Ruhl, G., Dighe, S., Wilson, H., Tschanz, J., Finan, D., Iyer, P., Singh, A., Jacob, T., Jain, S., Venkataraman, S., Hoskote, Y. and Borkar, N. (2008) 'An 80-tile sub-100-W TeraFLOPS processor in 65-nm CMOS', *IEEE Journal of Solid-State Circuits*, Vol. 43, No. 1, pp.29–41.

Website

SPEC2006 CPU benchmark suite, <http://www.spec.org>