

DVCR: Diagonal Virtual Channel NoC Router Architecture for Multiprocessors

¹E. Lakshmi Prasad, ²A.R. Reddy and ²M.N. Giri Prasad

¹Department of ECE, Jawaharlal Nehru Technological University (JNTUA),
515005 Ananatapuramu, India

²Department of ECE, MadanPalle Institute of Technology and Science (MITS),
517325 Madanapalle, India

Abstract: Network on chip is a modern architecture for multiprocessor. Due to the complex routing in Network on Chip (NoC), it is obstructed with the issue of latency, deadlock and traffic congestion. The problem of deadlock and traffic congestion can be managed by the proposed method called as Diagonal Virtual Channel Router (DVCR) design. Low latency XY routing algorithm can reduce latency to reach the critical path destination node in NoC. Therefore, DVCR and XYD routing algorithm can manage the congestion and latency can be reduced by 50% when compared to existing methods. These methods are examined for 4×4 2-D MESH and 2-D TORUS. Experimental research carried out by using Xilinx 14.7 and targeted on the Vertex-7 FPGA. As per the synthesis report, the minimum amount of time period to reach the critical path node in 2D-mesh is 5.548 nsec and for 2D-Torus is 5.507 nsec, so, each router can execute in four clock cycles, therefore, overall critical path distance node in 2D-mesh is 16.644 nsec and in 2D torus is 11.014 nsec. Low latency applications for NoC based MPSoCs.

Key words: System on chip, network on chip, multiprocessors, router architecture, multiprocessor, destination

INTRODUCTION

Network on Chip (NoC) is widely used for multi-core applications. In the traditional system on chip has a problem with enabling a parallel operation and avoiding the congestion.

So, network on chip is a solution for multi-core system applications. For example of 2×2 NoC based multi-core as shown in Fig. 1.

NoC plays a prominent role and it has internally associated with many hardware logic blocks such as Routing algorithm, crossbar network, buffers, routing computation block, input and output process state identifier, network interface and topology selection.

NoC router able to control the traffic when the network is busy and also can do the parallel operation. Routing algorithm can estimate the shortest path and able to reduce the communication latency. Cross-bar network allows to transverse the packets from one network router to another network router. Buffers are used to store the packets temporarily.

Routing computation block is to allow the packets by buffer channels based on the grants released by the state identifier. Input and output ports receive and transmit the packets. Network interface is used to establish the path between the NoC routers. Topology is selected based on

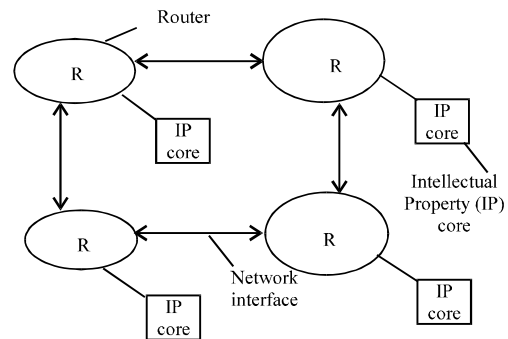


Fig. 1: 2×2 NoC based multi-core

the complexity of the application. Most of the NoC based multi-core applications are implemented by using MESH topology. The mentioned hardware logic blocks are commonly used for traditional NoC Wormhole router (Modarressi *et al.*, 2010). The Wormhole router has a problem with deadlock and starvation.

So, the main objective of this study is to avoid the problem of deadlock and starvation by using traditional XYD routing algorithm and DVCS. Bousamra *et al.* (2012) proposed a centralized controller algorithm to adjust the time intervals between the NoC routers. This algorithm can control the number of iteration to reach the

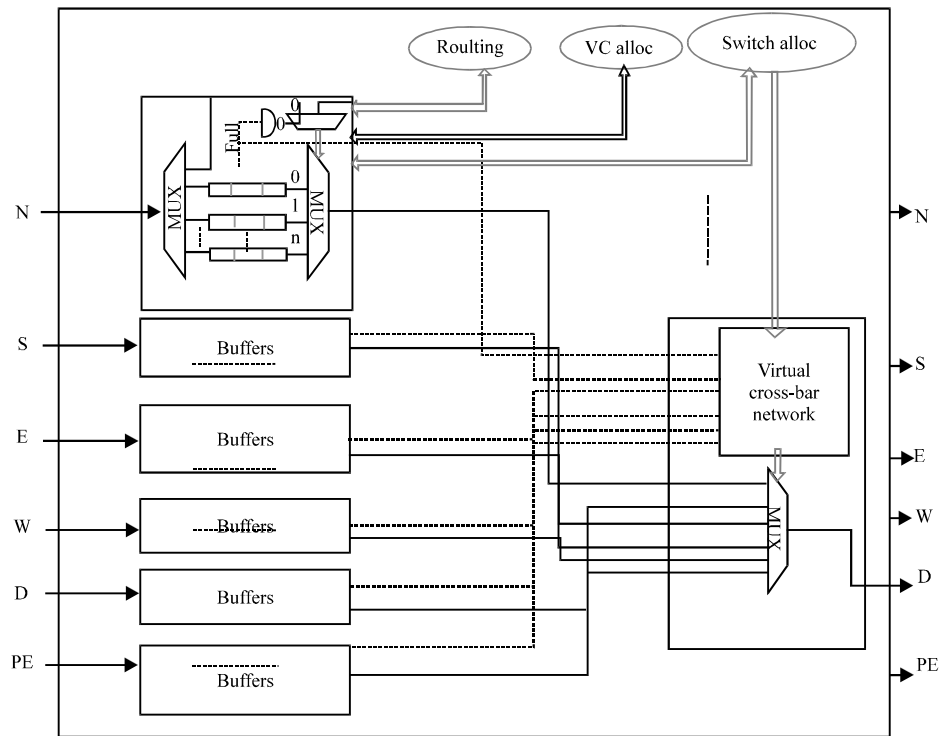


Fig. 2: Diagonal Virtual Channel Router (DVCR) architecture

destination from source by using Bubble sort. Overall flit latency among 16 nodes communication 11% reduced and 9% power consumption is reduced than compared to other traditional methods.

Grot *et al.* (2009) proposed a throttling based congestion control mechanism is used for NoC. With this algorithm, they achieve congestion free and can estimate the shortest way to communicate among the nodes in NoC. They showed a comparison of traditional routing algorithms and proposed routing algorithm, the proposed algorithm showed 28% throughput improvement in smaller NoCs and power consumption reduces by up to 20%.

Kumar *et al.* (2007) Proposed Express virtual channels to control the congestion flow in a busy network. Virtual channels are allowing the packets to bypass intermediate routers. Simulation results showed 85% reduction in packet latency and 23% improvement in throughput when compared to the existing designs.

Diagonal Virtual channel Router (DVCR): Every traditional NoC router has 5 directions such as East, west, north, South and local port. In the proposed DVCR has extended with 4 more directional ports such as North-West, East-North, East-South and South-West diagonal ports. The advantage of adding diagonal ports in the virtual channel router, the distance between the NoC routers communication will reduce the latency. The block diagram of DVCR as shown in Fig. 2.

DVCR is hybrid router architecture and it is having virtual buffers to store the packets temporarily (Prasad *et al.*, 2011). Each input port has contained virtual buffers and it associated with two multiplexers end to end. Routing block contains an information about to route the packets to reach the destination (Postman *et al.*, 2013). Virtual channel allocation block checks the status of availability of space in the buffers and switch allocator releases the grants to the virtual channel allocation block and it allocates the packets in the buffers (Prasad *et al.*, 2016a). Switch allocator is placed in between virtual channel block and crossbar network block. Virtual cross-bar network block is used to receive the packets and send the packets to the corresponding destination router. Here, virtual buffers play a key role because it can manage the traffic congestion and avoid deadlock problems (Prasad *et al.*, 2016b). In addition to that, diagonal directional is another advantage because the communication latency between the routers becomes reduces.

XYD routing algorithm: XY routing algorithm is generally used for 2D-NoC topology. As we have discussed in the introduction part each NoC router has five directional ports, hence, to direct the packets XY routing algorithm is more sufficient. But in the proposed approach extra diagonal ports are added. Hence, XY routing algorithm changed as XYD routing algorithm. DVCR and XYD

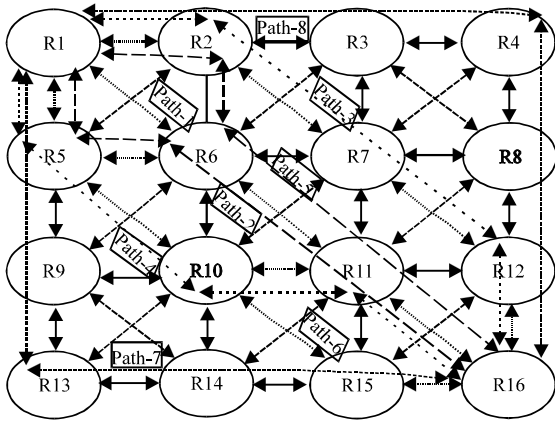


Fig. 3: The 4x4 2D-mesh NoC

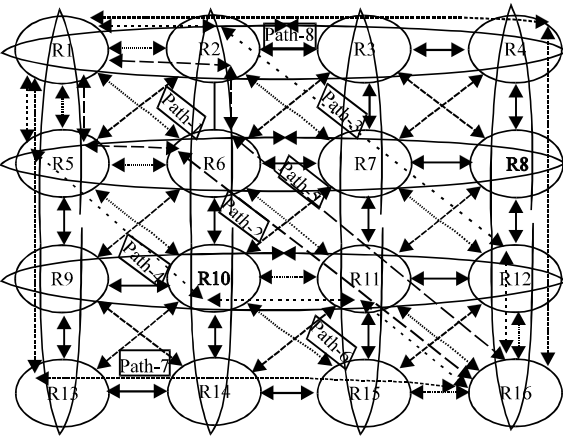


Fig. The 4: 4x4 2D-torus NoC

routing algorithm are applied to two traditional topologies such as 4x4 2D-mesh and 4x4 2D-torus as shown in Fig. 3 and 4. The XYD routing Algorithm 1 and 2 as explained in detail of each step operation.

Algorithm 1: XYD routing algorithm for 4x4 2D-mesh:

X-s = Source, X-t = target, X-plane referred as horizontal direction
Y-s = source, Y-t = target, Y-plane referred as vertical direction
D-s = source, D-t target, D-plane referred as diagonal direction
When $X = 0, Y = 0, D = 0$ // if all directions are zero then it selects the local ports
When $X > 0, Y = 0, D = 0$ // if X plane > 0 then it selects the horizontal forward direction
When $X < 0, Y = 0, D = 0$ // if X plane < 0 then it selects the horizontal backward direction
When $X = 0, Y > 0, D = 0$ // if Y plane > 0 then it selects the vertical forward direction
When $X = 0, Y < 0, D = 0$ // if Y plane < 0 then it selects the vertical backward direction
When $X = 0, Y = 0, D > 0$ // if D is > 0 then it selects the forward direction
When $X = 0, Y = 0, D < 0$ // if D is < 0 then it selects the backward direction

Algorithm 2: XYD routing algorithm for 4x4 2D-Torus:

X-s = Source, X-t = target, X-plane referred as horizontal direction
Y-s = source, Y-t = target, Y-plane referred as vertical direction
D-s = source, D-t = target, D-plane referred as diagonal direction
R-s = source, R-t = target, Ri-plane referred as ring interconnection from one end to another end
When $X = 0, Y = 0, D = 0, Ri = 0$ // if all directions are zero then it selects the local ports
When $X > 0, Y = 0, D = 0, Ri = 0$ // if X plane > 0 then it selects the horizontal forward direction
When $X < 0, Y = 0, D = 0, Ri = 0$ // if X plane < 0 then it selects the horizontal backward direction
When $X = 0, Y > 0, D = 0, Ri = 0$ // if Y plane > 0 then it selects the vertical forward direction
When $X = 0, Y < 0, D = 0, Ri = 0$ // if Y plane < 0 then it selects the vertical backward direction
When $X = 0, Y = 0, D > 0, Ri = 0$ // if D is > 0 then it selects the forward direction
When $X = 0, Y = 0, D < 0, Ri = 0$ // if D is < 0 then it selects the backward direction
When $X = 0, Y = 0, D = 0, Ri > 0$ // if Ri is > 0 then it selects the forward direction
When $X = 0, Y = 0, D = 0, Ri < 0$ // if Ri is < 0 then it selects the backward direction

The above two topologies followed the directions based on the values of XYD, due to extra diagonal direction reduces the communication distance from Source to Destination for 2D-mesh and 2D-torus. For example in 2D-mesh and 2D-torus, consider a communication critical path distance from source R1 to destination R16 it takes minimum of 6 hops to reach the destination but in case of diagonal direction it takes just 3 hops in 2 Dmesh and 2 hops in 2D-torus to reach the destination. Hence, the communication latency is reduced because of the diagonal direction and congestion problems are managed by using virtual channels in DVCR.

RESULTS AND DISCUSSION

DVCR and XYD routing algorithm can able to manage the traffic congestion and latency in Network on chip. These two routing algorithms are applied and examined to 4x4 2D-mesh and 2D torus. Simulation and synthesis results are carried out by using Xilinx-14.7 and targeted on the device vertex-7 FPGA⁸. According to the synthesis report, overall minimum cycle period for 2D-mesh is 5.548 ns and for 2D-torus is 5.507 ns, so, each router can execute in four clock cycles, therefore overall critical path distance node in 2D-mesh is 16.644 nsec and in 2D torus is 11.014 nsec. The performance comparison of 2D-mesh and 2D-torus as shown in Fig. 5 and 6.

In synthesis report shows that in terms of area and the power consumption is bit more in 2D-torus than 2D-mesh. The only advantage with 2D-torus is hop count is less when compared to the 2D-mesh. However, because of the diagonal direction hop count is reduced into half when compared to the other traditional router design.

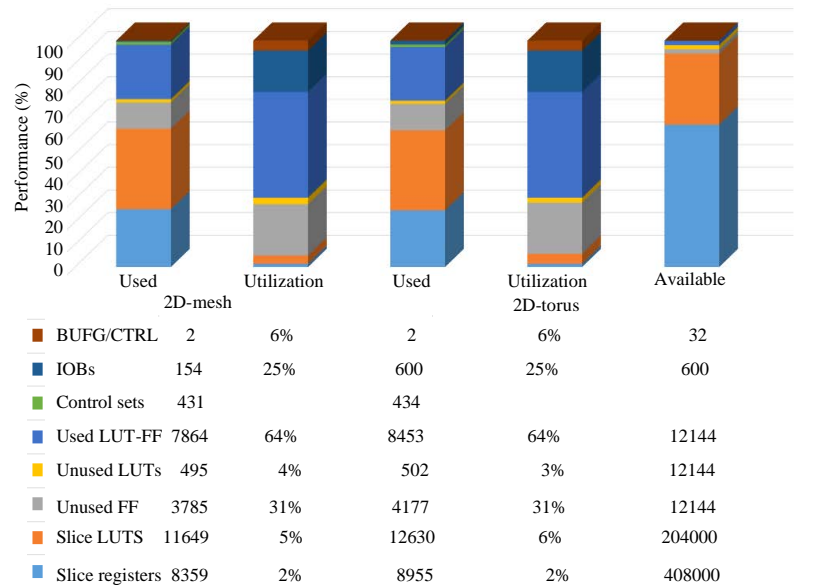


Fig. 5: Synthesis comparison report for 2D-mesh and 2D-torus

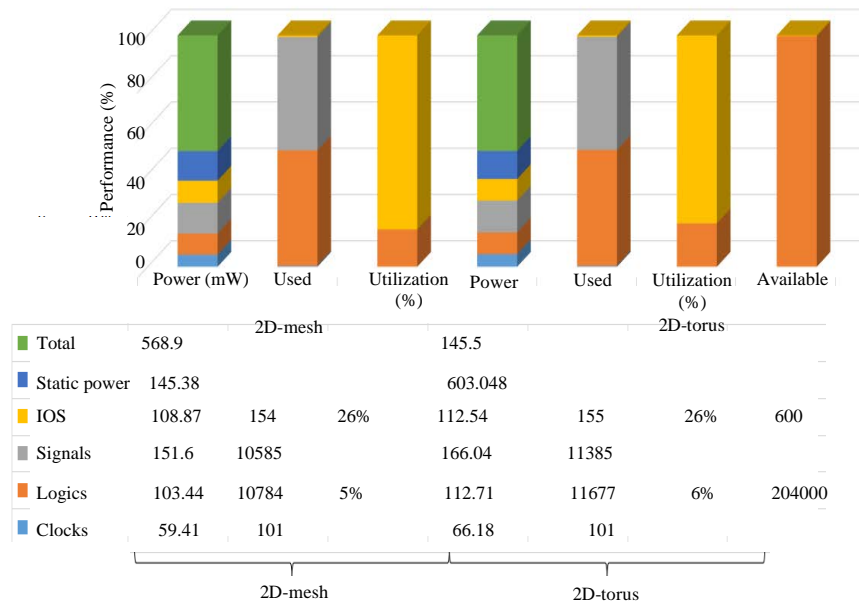


Fig. 6: Power comparison report for 2D-mesh and 2D-torus

CONCLUSION

In this study main objective is to manage the traffic congestion and reduce the latency this is achieved by using DVCR and XYD routing algorithm. Therefore, deadlocks managed by using virtual channels and latency got reduced by diagonal direction. DVCR and XYD examined and applied to the 2D-mesh and 2D-Torus, according to the synthesis report the area is little more in 2D torus about 5% and power consumption nearly equal but the latency is reduced by 29% when compared to the

2D-mesh. Moreover, without diagonal direction to reach the critical path node in 2D-mesh it takes minimum of 6 hops but due to the diagonal direction in DVCR, the hop count got reduced by half.

ACKNOWLEDGEMENT

The researchers would like to thank the Principal and management of MITS madanapalle for their kind support on behalf of TEQIP-2 world bank organization.

REFERENCES

- Bousamra, A., A.K. Jones and R. Melhem, 2012. Codesign of NoC and cache organization for reducing access latency in chip multiprocessors. *IEEE. Trans. Parallel Distrib. Syst.*, 23: 1038-1046.
- Grot, B., J. Hestness, S.W. Keckler and O. Mutlu, 2009. Express cube topologies for on-chip interconnects. *Proceedings of the IEEE 15th International Symposium on High Performance Computer Architecture HPCA 2009*, February 14-18, 2009, IEEE, Raleigh, North Carolina, USA., ISBN:978-1-4244-2932-5, pp: 163-174.
- Kumar, A., L.S. Peh, P. Kundu and N.K. Jha, 2007. Express virtual channels: Towards the ideal interconnection fabric. *ACM. SIGARCH. Comput. Archit. News.*, 35: 150-161.
- Modarressi, M., A. Tavakkol and H. Sarbazi-Azad, 2010. Virtual point-to-point connections for NoCs. *IEEE. Trans. Comput. Aided Des. Integr. Circuits Syst.*, 29: 855-868.
- Postman, J., T. Krishna, C. Edmonds, L.S. Peh and P. Chiang, 2013. Swift: A low-power network-on-chip implementing the token flow control router architecture with swing-reduced interconnects. *IEEE. Trans. Very Large Scale Integr. Syst.*, 21: 1432-1446.
- Prasad, E.L., A.R. Reddy and M.G. Prasad, 2016. EFASBRAN: Error free adaptive shared buffer router architecture for network on chip. *Procedia Comput. Sci.*, 89: 261-270.
- Prasad, E.L., A.R. Reddy and M.G. Prasad, 2016. Performance comparison of network on chip methods. *Proceedings of the Online International Conference on Green Engineering and Technologies (IC-GET)*, November 19, 2016, IEEE, Coimbatore, India, ISBN:978-1-5090-4557-0, pp: 1-8.
- Prasad, E.L., V. Sivasankaran and V. Nagarajan, 2011. Multiple task migration in mesh network on chips over virtual point-to-point connections. *Intl. J. Comput. Intell. Inf.*, 1: 202-207.