Exploring Topologies for Source-synchronous Ring-based Network-on-Chip

Ayan Mandal, Sunil P. Khatri and Rabi N. Mahapatra
Texas A&M University, College Station TX 77843
{ayan1731, sunilkhatri}@tamu.edu, rabi@cse.tamu.edu

Abstract— The mesh interconnection network has been preferred by the Network-on-Chip (NoC) community due to its simple implementation, high bandwidth and overall scalability. Most existing mesh-based NoC designs operate the mesh at the same or lower clock speed as the processing elements (PEs). Recently, a new source synchronous ring-based NoC architecture has been proposed, which runs significantly faster than the PEs and offers a significantly higher bandwidth and lower communication latency. The authors implement the NoC topology as a mesh of rings, which occupies the same area as that of a mesh. In this work, we evaluate two alternate source synchronous ring-based NoC topologies called the ring of stars (ROS) and the spine with rings (SWR), which occupy a much lower area, and are able to provide better performance in terms of communication latency compared to a state of the art mesh. In our proposed topologies, the clock and the data NoC are routed in parallel, yielding a fast, synchronous, robust design. Our design allows the PEs to extract a low jitter clock from the high speed ring clock by division. The area and performance of these ring-based NoC topologies is quantified. Experimental results on synthetic traffic show that the new ring-based NoC designs can provide significantly lower latency (upto $4.6\times$) compared to a state of the art mesh. The proposed floorplan-friendly topologies use fewer buffers (upto 50% less) and lower wire length (upto 64.3% lower) compared to the mesh. Depending on the performance and the area desired, a NoC designer can select among the topologies presented.

I. Introduction

With the increasing number of PEs in chip multi-processors (CMPs), the communication between the PEs has become a major performance bottleneck. A packet switched paradigm called the Network-on-Chip (NoC) has emerged as the communication subsystem of choice. Researchers have focused their attention on key design aspects of NoCs such as topology [1], [2], [3], global wire management [4], power optimization [5], flow control [6], [7] and routing [8], [9]. Regular topologies like the mesh and torus are common due to their simple implementation, high bandwidth and overall scalability. Other popular topologies include the Ring [10], Fat Tree, 2D Flattened Butterfly [11] and Octagon [12]. In all the above NoC architectures, design decisions are made assuming that the NoC operates at the same or lower clock speed as the PEs, which slows down the communication subsystem.

Recently, resonant clocking techniques [13], [14], [15], [16] have been demonstrated, providing ultra high-speed, low power, stable on-chip clock generators. In [17], [18], the authors utilize such a clock to develop a high-speed, source-synchronous ring-based NoC architecture. The authors implement the NoC topology as a Mesh of Rings (MOR). They report a NoC which runs $9\times$ faster than the PEs [17]. Architectural results [18] obtained on synthetic traffic demonstrate that the ring-based NoC has up to $3.5\times$ lower latency and up to $2.9\times$ higher maximum sustained

injection rate compared with a state of the art mesh-based NoC. However, the MOR consumes the same wiring and buffer area as that of a state of the art mesh. In addition, the PEs operate on a different clock than the NoC, and hence synchronizers are required when the PEs and the ring-based NoC communicate with each other.

In this work, we explore two alternate source synchronous ring-based topologies called the ring of stars (ROS) and the spine with rings (SWR) which consume much lower area than the MOR or a state of the art mesh, and are able to provide better performance in terms of communication latency compared to a state of the art mesh. The ROS topology is constructed by connecting multiple star networks in form of a ring. Each star network is again constructed by connecting smaller star networks. The smallest star network directly connects the PEs. The SWR topology consists of concentric rings with a horizontal and a vertical spine connecting all the rings. In both of the proposed topologies, the data NoC is being routed parallel to a clock ring, which is driven by a fast resonant clock. We use standing wave resonant oscillators (SWOs) to implement the clock rings. Any SWO ring is injection locked with other SWO rings that intersect it using the technique described in [14], and hence the entire NoC is synchronous. Our design allows the PEs to extract a low jitter clock from the high speed ring clock by division, thus making the PEs synchronous with the ring clock. Hence we eliminate the need of synchronizers between the PEs and the NoC (unlike [17], [18]). We evaluate the area and performance of the two new ring-based NoC topologies and compare them with a state of the art mesh and the MOR [18].

The key contributions of this paper are:

- We present two floorplan-friendly topologies for use in a source-synchronous ring-based NoC, wherein the clock is generated by a fast resonant SWO.
- The clock and data NoC are routed in parallel in our proposed architectures yielding a fast, robust design.
- Our design allows the PEs to extract a low jitter clock directly from the high speed ring clock by division. As a result, the PEs are synchronous with the ring clock and hence we avoid the need of synchronizers when the PEs and the NoC communicate.
- The proposed topologies use fewer buffers (upto 50% less) and lower wire length (upto 64.3% lower) compared to a state of the art mesh, as well as the MOR [18].
- Architectural simulations on synthetic traffic show that our proposed ring-based NoC topologies can provide significantly lower latency (upto $4.6\times$) compared to a state of the art mesh.

The rest of the paper is organized as follows. Section II de-

scribes previous approaches in this area. Section III presents our approach, while Section IV describes the experimental results which we performed to validate our approach. We conclude in Section V.

II. PREVIOUS WORK

The NoC topology determines the way in which the PEs in a CMP are connected to each other, and affects the bandwidth and latency of the resulting communication network. In terms of topology, the mesh [19] and torus [20] have received greatest attention due to their regular and modular structure, making them a popular choice. However, the mesh and torus topologies suffer from a large communication radius, which results in large amounts of interconnect and a large number of arbiters at the N-S-E-W crossings, as well as high power consumption. The Octagon [12] is a hierarchical network which consists of a basic *octagon unit* having eight nodes and 12 bidirectional links. It has a simpler implementation compared to the mesh, with a higher throughput. Topologies like the 2D flattened butterfly [11] and fat tree [21] provide dense connectivity compared to a mesh, with the downside of the requirement for large-radix routers and increased link area. A topology of concentric rings (similar to a ring road in city) has been proposed in [10], which reduces the risk of congestion in the central parts of the network. Application-specific topologies [22] that can offer superior performance while minimizing area and energy consumption have been also proposed. In [23], the authors implement a simple yet effective reconfigurable source-synchronous NoC, which can can sustain a peak throughput of one word per cycle. There has been also work on asynchronous mesh NoCs [24], which yield a 30-50% gain in speed and a $5\times$ reduction in power, at the cost of $3\times$ more area compared to a synchronous mesh. In [25], the authors have implemented a low-overhead asynchronous NoC, which has significantly lower latency and competitive throughput for mid-range injection rates, but suffers degradation at higher injection rates. In [26], the authors have implemented a power-efficient mesochronous NoC with no area and latency overhead.

In all the above implementations, design decisions were made assuming that the NoC runs at the same or lower clock speed than the PEs. In [17], the authors present a fast source synchronous ring-based NoC architecture which runs significantly faster ($9\times$) than the PEs. We refer to their NoC topology as a Mesh of Rings (MOR). Data is driven in a source-synchronous manner along with a high speed resonant clock. This allows [17] to achieve significantly higher bisection bandwidth with narrower links (yielding a lower area and power for the same bisection bandwidth). The significantly lower latencies allow the NoC architecture of [17] to scale elegantly for larger CMPs. The authors have validated their approach by means of thorough circuit simulations. In [18], the authors focus on the architectural aspects of the ring-based NoC, providing deadlock free routing using link ordering and virtual channels. Architectural results obtained on synthetic traffic demonstrate that the modified ring-based NoC has up to $3.5\times$ lower latency and up to $2.9\times$ higher maximum sustained injection rate compared with a state of the art mesh-based NoC. However, in the above implementations, the authors assume a separate clock distribution for the NoC and the PEs, and focus on standard NoC topologies like the mesh (of rings). In this paper, we propose a NoC architecture with a unified synchronous clock distribution for the NoC and the

PEs. Hence, in our design, the PEs can extract a low jitter clock directly from the high speed ring clock by division. Moreover, since the PEs are synchronous with the ring clock, we avoid the need of synchronizers while PEs and the NoC communicate with each other. In addition, the ring-based NoC in [18] utilizes the same wiring and buffer area as that of a traditional mesh. In contrast, our two alternate source synchronous ring-based topologies consume much lower area than the mesh or MOR, and are able to provide better performance in terms of communication latency compared to a state of the art mesh.

Our approach is based on the use of a very fast resonant clock obtained using standing wave resonant oscillators [13]. Standing wave resonant oscillators (SWO) are a promising technique to generate a high-frequency on-chip clock signal with low power. In [14], the authors present a tiled SWO based resonant grid for high frequency clock distribution. They show how multiple SWOs can be connected such that they oscillate with the same high frequency and phase (by injection locking). The layout of individual SWOs is constructed carefully, to ensure that the electrical environment around each SWO ring is identical. We borrow the injection locking technique [14] to synchronize multiple SWOs which intersect each other. We route the NoC datapath parallel to the SWO ring clock distribution. This enables the NoC routers to derive a high speed, low jitter clock directly from the SWO. In addition, the PEs can extract their clock from the ring by clock division. In this manner, the PE and the NoC clocks are synchronous.

III. OUR APPROACH

In this section, we first discuss our approach towards the resonant clock distribution used in our design. Next, we discuss the router architecture used in our design. Finally, we talk about the two alternate source synchronous ring-based topologies followed by their corresponding deadlock-free routing. In the following discussion, we assume a CMP with 64 PEs, for all NoC topologies.

A. Resonant Clocking

Figure 1 shows a standing wave oscillator (SWO) [13]. A long wiring ring is used and oscillations are sustained in this resonant ring by using a cross coupled inverter pair, as shown in Figure 1. The parasitic capacitance C and the parasitic inductance L of ring structure results in oscillation at a frequency $\frac{1}{2\pi\sqrt{LC}}$. A mobius connection at the end of the ring ensures that the clock signal at any point in the ring is sinusoidal and has the *same phase* at all points along the ring. Differential amplifiers are used to recover a square wave clock anywhere along the ring. Hence, we obtain clock signals that have the same phase everywhere along the ring. The reduced ring capacitance due to the use of a single cross-coupled inverter pair increases the operating speed and reduces power consumption as well. There is an AC null (virtual "zero") point in the center of the ring and hence the clock recovery is not performed around the null point.

In the various ring-based topologies that we propose in our current work, we require clock rings of different length, that oscillate at the same frequency. However, for an SWO, as we increase the length of the clock ring, the frequency decreases, since both the parasitic capacitance C and the parasitic inductance L increase. The total phase change is $\lambda/2$ while traversing the ring once for an SWO of perimeter p. If we increase the
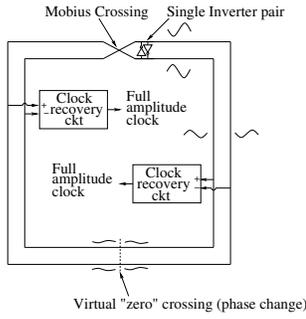
Fig. 1. Standing Wave Oscillator [13]
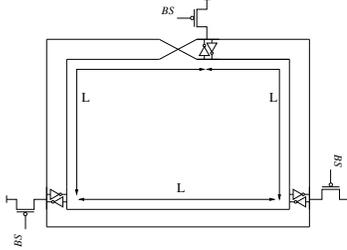
Fig. 2. Circuit Topology of a 3λ/2 SWO Ring

perimeter of the ring to $k \cdot p$ (where k is odd) and introduce a total phase change of $k \cdot \lambda/2$ over a single traversal of the ring, we obtain an SWO ring of larger perimeter, whose frequency is the same as a ring with perimeter p and phase change $\lambda/2$. Such an implementation requires p equally spaced inverter pairs, and an odd number (typically one) of mobius connections. The circuit configuration for a $3 \cdot \lambda/2$ ring is shown in Figure 2. Now in order to ensure that the resonant structure bootstraps in a standing wave configuration, the signals at the inverter pair are initialized using a global bootstrap signal (labeled *BS* in Figure 2). Two SWOs with different perimeters, oscillating at the same frequency use injection locking (at the location where their rings intersect) to ensure identical phase across both the rings. Hence, we achieve a synchronous clock distribution across the die. In our experiments, the ring which distributes the clock is assumed to be laid out on Metal 8, with wires of width $1\mu m$, spacing $1\mu m$ and height $0.9\mu m$. We generate a $14GHz$ clock for the NoC. The required perimeter of the metal wire is $2.75mm$ (corresponding to $\lambda/2$) For rings with perimeter greater than $2.75mm$, we use a $k \cdot \lambda/2$ perimeter (where k is odd). The PEs can extract their clock by division from the SWOs. The above design suffers from a drawback that the clock provided to the PEs are not phase locked to the external reference. However, this is fixed by the observation that only IO PEs need a phase locked loop (PLL) in order to establish off-chip communication. Hence the IO PEs that oscillate at the same frequency can have a separate PLL, and will require a synchronizer while communicating with the NoC. The rest of PEs (which are not IO PEs) can derive their clock from the SWOs by clock division.

B. Basic Router Architecture

The ring-based NoC topologies use two different types of routers called *Insertion Extraction Station* (IES) and *Junction Station* (JS). Each PE is connected to the ring-based NoC by means of an IES, which allows it to insert/extract data into/from the ring. A JS is placed wherever two or more rings intersect, allowing flits to switch rings. The routers are implemented as a 3 stage pipelined virtual-channel router [18], supporting 6 virtual-channels (VCs) per port. In the first pipeline stage (Buffer Write (BW) + Route Compute (RC)), a flit arrives at the input port and the output port is calculated. The arriving flits at the input ports arbitrate for their corresponding output port in the second pipeline stage (Output Port Allocation (OPA)). Flits from the output port are driven out through the output link in the third pipeline stage (Link Traversal (LT)).

C. The ROS Topology

1) Architecture: Figure 3 shows our proposed topology of ring of stars (ROS). The smallest star network (level-1) connects 4 PEs and consists of 4 IES' (circles) and 1 JS (square). A level-2 star network is responsible for connecting 4 level-1 star networks. Hence, a level-2 star network consists of 5 JS'. Finally, we have a ring which connects 4 level-2 stars to provide full connectivity for the 64-PE CMP. We highlight a portion of the NoC around the JS in the left of Figure 3. The JS is present at the intersection of two clock rings. The two intersecting clock rings are injection locked and hence oscillate with the same frequency and phase. We also show the five bidirectional links connected to the JS responsible for carrying the data for the NoC. The JS extracts a high speed, low jitter clock directly from these clock rings.
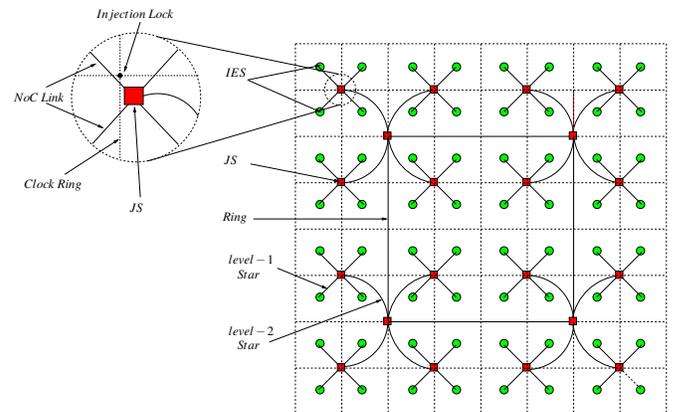
Fig. 3. ROS topology

2) Deadlock-free Routing: In [27], the authors prove that the necessary and sufficient condition for deadlock-free routing is the absence of cycles in the channel dependency graph. A star-topology by construction is acyclic. However, as we introduce a ring to connect multiple stars, we introduce cycles in the channel dependency graph. We visualize the ROS as a 2×2 mesh with a star network rooted at each of the four mesh nodes. The routing in each of the star network is deadlock free (since the star network is acyclic). We establish a deadlock free route in the mesh, using dimension-ordered routing (DOR)-XY. This achieves a deadlock free route for the entire ROS.

D. The SWR Topology

1) Architecture: Figure 4 shows our proposed topology of spine with rings (SWR). The topology consist of 4 concentric rings with a vertical and horizontal spine connecting the rings. The innermost ring is the smallest ring and connects only 4 PEs at the center. The outermost ring is the largest and connects 28 PEs which are present the periphery of the CMP. One vertical and one horizontal spine connects these rings to provide full connectivity for the 64-PE CMP. IES's in Figure 4 are shown

as circles, while JS' are shown as squares. A portion of the NoC around the JS is highlighted in the left of Figure 4. The two intersecting clock rings at the JS are injection locked, and hence oscillate with the same frequency and phase. We also show the four bidirectional links connected to the JS, responsible for carrying the data for the NoC. The JS extracts a high speed, low jitter clock directly from these clock rings.
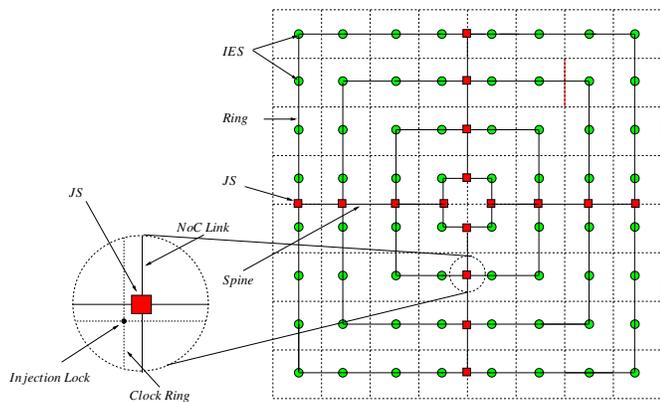
Fig. 4.   SWR topology

2) Deadlock-free Routing: The SWR is composed of 4 concentric rings and a vertical and a horizontal spine connecting the concentric rings. Clearly the SWR has cycles in the channel dependency graph. Each router (IES and JS) in our design supports 6 virtual channels. We construct two virtual networks with the help of these 6 virtual channels. Three of the virtual channels are reserved for flits which travel from an outer ring to an inner ring. The other half of the virtual channels is reserved for flits travelling from an inner ring to outer ring. Hence the flits of these two virtual networks do not share resources and do not create a cycle in the channel dependency graph. This achieves a deadlock free route for SWR.

We compare the above ring-based NoC topologies in terms of area, average communication latency and maximum number of flits delivered. In addition, we analyze the average link utilization (theoretical and experimental) of the links in the ring-based NoCs, which provides an insight to the maximum injection rate sustainable by a NoC structure, and allows us to debug the link(s) that are responsible for congestion.

IV. EXPERIMENTAL RESULTS

A. Evaluation Platform

We use a modified version of GEM5 [28] for cycle-accurate micro-architectural NoC simulations. For all topologies, we simulate a network with 64 PEs. Each PE tile is assumed to be $1.229mm \times 1.229mm$ (using the estimates of [17]). The link width in any direction is assumed to be 18 bytes. The routers (IES' and JS') operate at 14 GHz and can drive a maximum link length of $0.615mm$. For links greater than $0.615mm$, we insert repeater(s) to ensure that largest link driven is $0.615mm$. We assume a clock frequency of 2 GHz for the PEs and a clock frequency of 14 GHz for the various ring-based NoC designs. We assume a single flit packet of 18 bytes. Each of the routers (IES' and JS') support 6 virtual channels. We compare our architectural results with a state of the art mesh with virtual channel buffered flow control [29]. The mesh operates at 2 GHz.

The routers in the mesh are 3-stage pipelined supporting 6 virtual channels and perform a dimension-ordered (DOR)-XY routing.

Figure 5 shows the cross section of the wiring used in implementing the SWO rings. The clock wires are implemented in METAL9, and are shielded on either side, as well as above and below, by grounded wires.
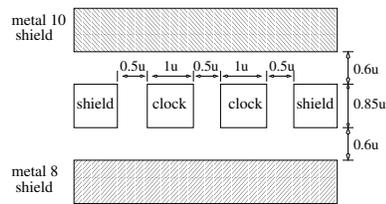
Fig. 5.   Cross section of Wires in any H-tree Segment

B. Area Comparison

Table I reports the number of links, wire length, the total number of input ports and the total number of buffers across all the routers for the MOR [18], ROS, SWR and mesh [29] topologies. The first column is intended to provide a coarse measure of connectivity. Note that the links reported in Column 1 are not all of the same length. Hence, we report the total wire length of all the links in the second column. The buffers were shown to occupy 75% of the total on-chip network area [30] in the TRIPS chip and hence we use them as an indicator of the global logic area requirement. Note that since each router supports 6 virtual channels, the number of buffers is six times the number of ports.

Topology	Number of Links	Wire Length (mm)	Number of ports	Number of buffers
MOR [18]	112	275.30	288	1728
ROS	84	206.10	168	1008
SWR	92	98.32	144	864
Mesh [29]	112	275.30	288	1728

TABLE I
AREA COMPARISON FOR VARIOUS NoC TOPOLOGY

Clearly the MOR and the mesh have the longest wire length and the highest number of buffers. The ROS and SWR have 25.2% and 64.3% lower wire length respectively compared to the mesh. Moreover, the ROS and the SWR have 41.7% and 50% fewer buffers respectively, compared to the mesh. In the rest of this section, we show that our ROS and SWR topologies leverage the benefit of a high speed NoC by reducing the wiring length and the number of buffers and still provide lower communication latency compared to a state of the art mesh.

C. Link Utilization

In this section, we analyze the utilization of the various links for different ring-based NoC designs. Figure 6 shows the histogram of the analytical and experimental link utilization for the ROS and SWR topologies. For analytical link utilization, we assume that each link is able to provide an unlimited bandwidth to the incoming flits. For experimental link utilization, the *PEs* inject traffic uniformly and the injection rate is chosen to ensure that the corresponding network does not saturate. The injection rates used for both ROS and SWR were 0.35. The same injection rate was used for the analytical experiment as well. For the ROS topology, the level-1 star networks have the lowest utilization since each of them serves only 4 PEs. The utilization increases

for the level-2 star network which serves 16 PEs. Finally, the ring which connects four level-2 star networks has the highest utilization. For the SWR topology, all the links present in the concentric rings have same utilization due to symmetry. However, the spines which are responsible for connecting all the concentric rings have the highest link utilization. From Figure 6, we observe that there is a very close resemblance between the analytical and the experimental link utilization. Ideally, we prefer a link utilization distribution that is uniform across all the links. For a ROS topology, the link utilization is non-uniform due to its hierarchical topology. For the SWR topology, the link utilization is uniform across all the links in the concentric rings but is highest for the links in the spine connecting the rings. The above study provides an insight to the network load of each topology and also gives an indication about the maximum sustained injection rate. The results of the above experiment can allow the NoC designer to determine which links need to be widened, and thereby fine-tune the performance verses the area trade off.
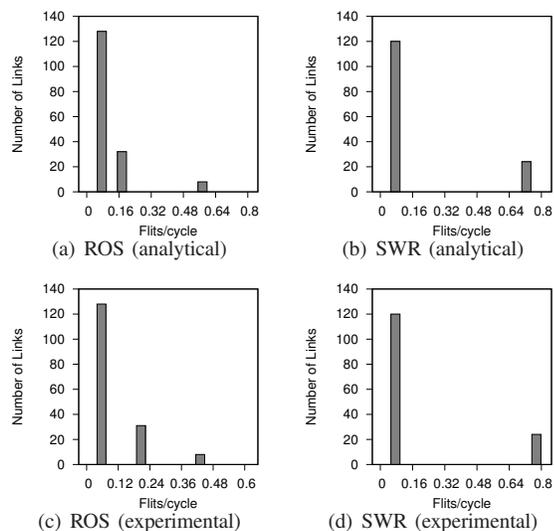
Fig. 6.   Link Utilization

D. Architectural Simulations

We compare the performance of different ring-based NoC designs along with a state of the art mesh by running synthetic traffic (uniform, tornado and bit-complement) through these NoCs. Figure 7 presents the latency (in terms of *PE* cycles) as a function of flit injection rate. Figure 8 provides the number of flits delivered over 10K *PE* cycles as a function of flit injection rate for the corresponding traffic patterns. For uniform traffic, we observe that MOR, ROS and SWR provide on average 3.6×, 3.9× and 2.3× lower latency compared to a mesh. Moreover, from Figure 8, we observe that MOR, ROS and SWR can sustain an injection rate of 1.1, 0.42 and 0.36 respectively in comparison to 0.38 for the mesh.

We summarize the data of Figure 7 and Figure 8 (normalized to the mesh [29]) in Table II and Table III respectively. The maximum number of flits delivered over 10K *PE* cycles for MOR, ROS and SWR are 5.5×, 1.2× and 1.9× compared to the mesh as reported in Table III. The fact that the ring-based NoC designs run significantly faster than the mesh (by 7×) contributes to these improvements.

For tornado traffic, we observe that MOR, ROS and SWR provide on average 3×, 3× and 1.7× lower latency compared to a mesh and can sustain an injection rate of 0.85, 0.38 and 0.2 respectively in comparison to 0.35 for the mesh. The maximum number of flits delivered over 10K *PE* cycles for MOR, ROS and SWR are 6.6×, 1.8× and 1.6× compared to the mesh as reported in Table III. Finally, for bit-complement traffic, we observe that MOR, ROS and SWR provide on average 3.7×, 4.6× and 3× lower latency compared to a mesh and can sustain an injection rate of 0.66, 0.23 and 0.2 respectively in comparison to 0.23 for the mesh. The maximum number of flits delivered over 10K *PE* cycles for MOR, ROS and SWR are 3.8×, 1.1× and 1.9× compared to the mesh as reported in Table III.

Clearly, the MOR has the best performance in terms of the latency and the maximum sustained injection rate. However, we also observe that the ROS and the SWR provide a better latency (with a minimum of 1.7× better) for all synthetic patterns in comparison to a state of the art mesh. ROS is able to sustain the same or better injection rate as that of the mesh across all synthetic traffic patterns. However, SWR sustains a lower injection rate than that of the mesh across all synthetic traffic patterns. In the SWR, as suggested by the link utilization discussion, the spine becomes the throughput bottleneck at higher injection rate. Both the SWR and ROS topologies use fewer buffers (upto 50% less) and lower wire length (upto 64.3% lower) compared to the mesh, and therefore are significant improvements over the state of the art in this respect. Figure 8 indicates that both MOR and ROS are able to deliver a larger number of flits as the mesh, but with a significantly lower latency and area utilization.

Topology	Uniform	Tornado	Bit-Complement
Mesh [29]	1	1	1
MOR [18]	1/3.6	1/3	1/3.7
ROS	1/3.9	1/3	1/4.6
SWR	1/2.3	1/1.7	1/3

TABLE II

LATENCY COMPARISON

Topology	Uniform	Tornado	Bit-Complement
Mesh [29]	1	1	1
MOR [18]	5.5	6.6	3.8
ROS	1.2	1.8	1.1
SWR	1.9	1.6	1.9

TABLE III

MAXIMUM FLITS DELIVERED COMPARISON

V. CONCLUSIONS

In this paper, we evaluate two new floorplan-friendly source synchronous ring-based NoC topologies which consume much lower area and are able to provide better performance in terms of communication latency compared to a state of the art mesh. In our proposed topologies, the clock and the data NoC are routed in parallel, yielding a fast, robust design. Our design allows the PEs to extract a low jitter clock from the high speed ring clock by clock division. The area and performance of these ring-based NoC topologies is quantified. Experimental results on synthetic traffic show that the new ring-based NoC designs can provide significantly lower latency (upto 4.6×) compared to a state of the art mesh. The proposed topologies use fewer buffers (upto 50% less) and lower wire length (upto 64.3% lower) compared to a
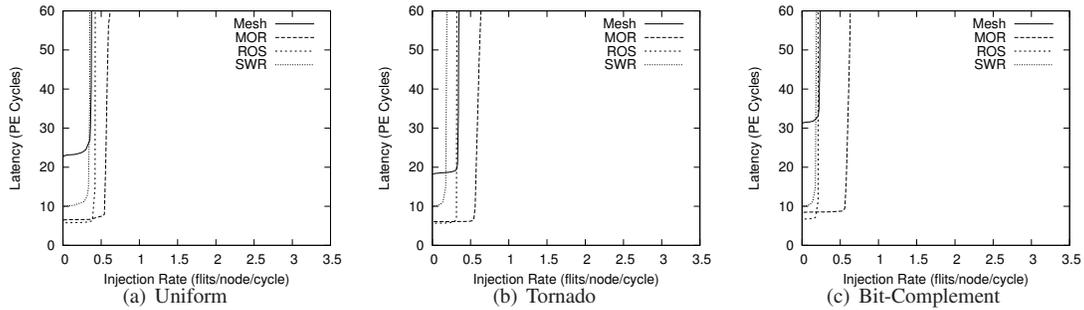
Fig. 7.   Latency Comparison
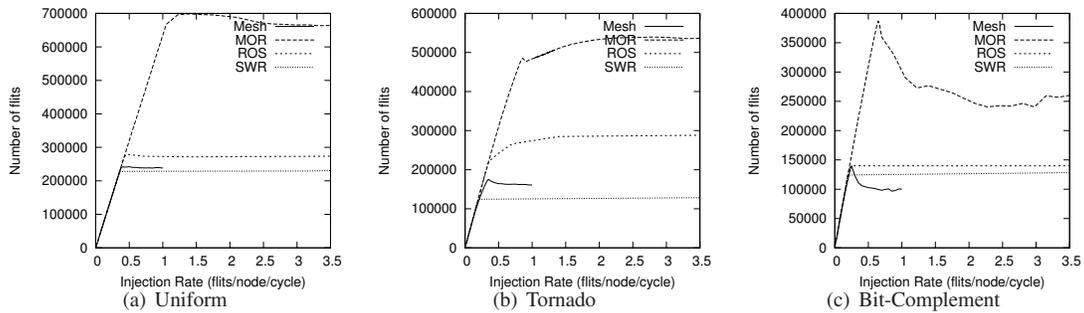
Fig. 8.   Number of Flits Delivered

mesh. The proposed ROS topology is able to sustain the same or better injection rate as that of the mesh across all synthetic traffic patterns. However, the SWR topology sustains a lower injection rate for all synthetic traffic patterns compared to the mesh, but has a significantly lower latency.

REFERENCES

[1] L. Bononi, N. Concer, M. Grammatikakis, M. Coppola, and R. Locatelli, "NoC Topologies Exploration based on Mapping and Simulation Models," in *Digital System Design Architectures, Methods and Tools, 2007. DSD 2007. 10th Euromicro Conference on*, pp. 543 –546, 2007.

[2] M. Kim, J. Davis, M. Oskin, and T. Austin, "Polymorphic On-Chip Networks," in *Computer Architecture, 2008. ISCA '08. 35th International Symposium on*, pp. 101 –112, june 2008.

[3] S. Tota, M. R. Casu, and L. Macchiarulo, "Implementation analysis of NoC: a MPSoC trace-driven approach," in *Proceedings of the 16th ACM Great Lakes symposium on VLSI*, GLSVLSI '06, pp. 204–209, ACM, 2006.

[4] R. Balasubramonian, N. Muralimanohar, K. Ramani, and V. Venkatachalapathy, "Microarchitectural Wire Management for Performance and Power in Partitioned Architectures," in *Proceedings of the 11th International Symposium on High-Performance Computer Architecture*, (Washington, DC, USA), pp. 28–39, IEEE Computer Society, 2005.

[5] L. P. H Wang and S. Malik, "Power-driven design of router microarchitectures in on-chip networks," in *Microarchitecture, 2003. MICRO-36. Proceedings. 36th Annual IEEE/ACM International Symposium on*, pp. 105 – 116, dec. 2003.

[6] B. Sethuraman, P. Bhattacharya, J. Khan, and R. Vemuri, "LiPaR: A light-weight parallel router for FPGA-based networks-on-chip," in *Proceedings of the 15th ACM Great Lakes symposium on VLSI*, GLSVLSI '05, (New York, NY, USA), pp. 452–457, ACM, 2005.

[7] P. Kermani and L. Kleinrock, "Virtual cut-through: a new computer communication switching technique," *Computer Networks*, vol. 3, pp. 267–286, 1979.

[8] M. Taylor, M. B. Taylor, W. Lee, S. Amarasinghe, and A. Agarwal, "Scalar Operand Networks: On-chip Interconnect for ILP in Partitioned Architectures," in *In International Symposium on High Performance Computer Architecture*, pp. 341–353, 2002.

[9] D. Lenoski, J. Laudon, K. Gharachorloo, W. dietrich Weber, A. Gupta, J. Hennessy, M. Horowitz, M. S. Lam, and D. T. E. of use, "The Stanford DASH multiprocessor," *IEEE Computer*, vol. 25, pp. 63–79, 1992.

[10] H. Samuelsson and S. Kumar, "Ring Road NoC architecture," in *Norchip*, pp. 16–19, 2004.

[11] J. Kim, J. Balfour, and W. Dally, "Flattened butterfly topology for on-chip networks," *Computer Architecture Letters*, vol. 6, pp. 37 –40, feb. 2007.

[12] F. Karim, A. Nguyen, and S. Dey, "An interconnect architecture for networking systems on chips," *Micro, IEEE*, vol. 22, pp. 36 – 45, sep/oct 2002.

[13] V. H. Cordero and S. P. Khatri, "Clock distribution scheme using coplanar transmission lines," in *Proceedings of the conference on Design, automation and test in Europe*, DATE '08, (New York, NY, USA), pp. 985–990, ACM, 2008.

[14] A. Mandal, V. Karkala, S. Khatri, and R. Mahapatra, "Interconnected Tile Standing Wave Resonant Oscillator Based Clock Distribution Circuits," in *VLSI Design (VLSI Design), 2011 24th International Conference on*, pp. 82 –87, jan. 2011.

[15] J. Wood, T. Edwards, and S. Lipa, "Rotary traveling-wave oscillator arrays: a new clock technology," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 1654–1665, Nov 2001.

[16] J. Wood, T. Edwards, and C. Ziesler, "A 3.5GHz Rotary-Traveling-Wave-Oscillator Clocked Dynamic Logic Family in 0.25 μm CMOS," *Solid-State Circuits Conference, 2006. ISSCC 2006. Digest of Technical Papers. IEEE International*, pp. 1550–1557, Feb. 2006.

[17] A. Mandal, S. P. Khatri, and R. N. Mahapatra, "A fast, source-synchronous ring-based network-on-chip design," in *DATE*, pp. 1489–1494, 2012.

[18] A. Mandal, S. P. Khatri, and R. N. Mahapatra, "Architectural Simulations of a Fast, Source-Synchronous Ring-based Network-on-Chip Design," in *ICCD*, 2012.

[19] W. Dally and B. Towles, "Route packets, not wires: on-chip interconnection networks," in *Design Automation Conference, 2001. Proceedings*, pp. 684 – 689, 2001.

[20] J. Duato, S. Yalamanchili, and N. Lionel, *Interconnection Networks: An Engineering Approach*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002.

[21] C. E. Leiserson, "Fat-trees: universal networks for hardware-efficient supercomputing," *IEEE Trans. Comput.*, vol. 34, pp. 892–901, October 1985.

[22] J. Hu, Y. Deng, and R. Marculescu, "System-level point-to-point communication synthesis using floorplanning information," in *Proceedings of the 2002 Asia and South Pacific Design Automation Conference*, ASP-DAC '02, (Washington, DC, USA), pp. 573–, IEEE Computer Society, 2002.

[23] A. T. Tran, D. N. Truong, and B. Baas, "A Reconfigurable Source-Synchronous On-Chip Network for GALS Many-Core Platforms," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 29, pp. 897 –910, june 2010.

[24] Y. Thonnart, P. Vivet, and F. Clermidy, "A fully-asynchronous low-power framework for GALS NoC integration," in *Proceedings of the Conference on Design, Automation and Test in Europe*, DATE '10, (3001 Leuven, Belgium, Belgium), pp. 33–38, European Design and Automation Association, 2010.

[25] M. Horak, S. Nowick, M. Carlberg, and U. Vishkin, "A Low-Overhead Asynchronous Interconnection Network for GALS Chip Multiprocessors," in *Networks-on-Chip (NOCS), 2010 Fourth ACM/IEEE International Symposium on*, pp. 43 –50, may 2010.

[26] D. Ludovici, A. Strano, G. N. Gaydadjiev, and D. Bertozzi, "Mesochronous NoC technology for power-efficient GALS MPSoCs," in *Proceedings of the Fifth International Workshop on Interconnection Network Architecture: On-Chip, Multi-Chip*, INA-OCMC '11, (New York, NY, USA), pp. 27–30, ACM, 2011.

[27] W. J. Dally and C. L. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," *IEEE Trans. Comput.*, vol. 36, pp. 547–553, May 1987.

[28] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, "The GEM5 simulator," *SIGARCH Comput. Archit. News*, vol. 39, pp. 1–7, Aug. 2011.

[29] J. D. Balfour and W. J. Dally, "Design tradeoffs for tiled CMP on-chip networks," in *International Conference on Supercomputing*, pp. 187–198, 2006.

[30] P. Gratz, C. Kim, R. McDonald, S. Keckler, and D. Burger, "Implementation and Evaluation of On-Chip Network Architectures," in *Computer Design, 2006. ICCD 2006. International Conference on*, pp. 477 –484, oct. 2006.