K2Router: A Low-Power and High-Performance Router Design for Networks-On-Chip

Hananeh Aliee    Hamid R. Zarandi    Pooria M. Yaghini

Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran

Abstract

This paper proposes a new methodology to design of coarse-grained routers in order to improve performance of Networks-on-Chip (NOC). In this approach, several cluster routers in region area of $K \times K$ are grouped into one new router called $K2Router$ with similar cost of area and power, however, the performance of whole network is improved. The proposed routers are general in terms of constructions and can be designed and implemented to associate 2x2, 3x3 and 4x4 PEs in a traditional mesh called D2Router, T2Router, and Q2Router, respectively. Based on connections available of such routers, several topologies can be configured. Some of these topologies called DLMesh, DMesh, TLMesh, and QLMesh. Several set of experiments were done on D2Router ($k=2$), Q2Router ($k=4$) and O2Router ($k=8$) with several topologies. The results show that as number of grouped PEs in a $K2Router$ increases, network performance increase dramatically. However, as $K$ increases in $K2Router$, implementation of the router becomes more complex to use in the network. This is because of wire length in two-dimensional circuit layout, and that would be solved in 3D circuit implementations. Moreover, in the case of using D2Router, the performance, cost and power overhead of DMesh with help of congestion-aware routing algorithm, would be better than those in traditional mesh.

Keywords: Network-on-Chip (NoC), Chip Multiprocessor (CMP), Network Topology, Router, Congestion-Aware Routing Algorithm.

1. Introduction

With technology scaling down into deep-submicron (DSM), it will be possible to integrate billions of transistors in a single chip [1]. Chip multiprocessors (CMPs) have emerged as a solution to achieve both high performance and reasonable power consumption within a packaged chip [2]. As the number of cores increases in such architecture, the complexity of designing efficient, scalable on-chip communication mechanism will continue to increase; furthermore, power consumption and wire delay will become significant limiting design constrains [3].

Network-on-Chip (NoC) has been introduced as a method to handle the on-chip communication and overcome the limitations. NoC has attracted research attentions because the interconnection architecture based on shared busses will not provide support for the communication requirements of future ICs [4-5].

To minimize the communication cost, these systems should be designed in a way that related cores are allocated closer to each other, due to the fact that an on-chip interconnection design that benefits from communication locality is likely to provide performance better than traditional designs [6-7]. In addition, to improve the packet latency and power consumption, two other critical metrics in designing CMPs, are widely depend on the underlying network [8, 9]. To this end, we were motivated to focus on an effective network topology and new router architecture to address these concerns.

Selecting network topology is a critical decision in designing on-chip networks for its impact on several performance metrics such as channel bandwidth, channel length, router micro architecture, routing algorithm, overall
power consumption and area [8]. Several works have studied network topologies. Although most of the tiled CMPs have utilized mesh topology for its low complexity, it has the disadvantage of poor scalability for larger conFigurations as a result of increased number of routers and average hop count [10-11].

The area and power consumed by channels are significant parts of aggregate area and power consumption of an on-chip network; thus, reducing the channel count reduces the overall area and power consumption. The number of channels can be decreased either by reducing the network diameter or the use of concentration, where the efficiency of the network is increased by sharing the network resources among different processing elements (PEs) [12]. An example of concentration is aggregating traffic from different PEs into a single router which can reduce hop counts. Concentration is practical where the probability that a single router attempts to access the network is relatively low; therefore, sharing the network resources among some PEs can improve network efficiency.

This work investigates the design of area- and power-efficient network-on-chip with low latency for tiled CMP architectures which benefits from concentration to reduce the number of channel as well as buffers. Our model incorporates various aspects of design including: topology, router architecture, and routing algorithm to fit wide range of CMPs. In the proposed architecture, $K \times K$ PEs share a single router called $K2Router$. The $K2Router$ routes flits, received from its neighbors and $K^2$ local PEs, to $4 \times K$ output ports. Local PEs can then communicate with each other via the router. Hereafter we called $K2Router$ with $k$ equal to 2, 3, 4, 8; $D2Router$, $T2Router$, $Q2Router$ and $O2Router$, respectively. Several different topologies are introduced in this paper based on the $K2Router$ for each value of $k$; for example, in the case of $k=2$, Double-Link mesh ($DLMesh$) topology and Diagonally mesh ($DMesh$) topology are introduced. In the first topology, each $K2Router$ connects to its four neighbors just like traditional mesh topology however with $k$ times more links. We called these topologies with $DLMesh$, $TLMesh$, $QLMesh$ and $OLMesh$ for $k$ equal to 2, 3, 4, 8, respectively. This implementation makes it possible to easily replace the $K2Router$ with a group of $K \times K$ connected routers in any mesh-based NoC. The later topology joins a $K2Router$ to $4 \times K$ neighbors in different directions. Explicitly speaking for $K2Router$ when $k$ is 2 (i.e., $D2Router$), direction can be north, east, west, south, southeast, southwest, northeast and northwest.

Deterministic routing may not be able to react to congestion, and consequently results in an increase in network delay. Adaptive routing algorithm is a solution to avoid congestion, so it is more suitable for NoC implementations [13]. It this paper we inspect buffer status of immediate neighbors to find the most appropriate output port for $DMesh$ topology because it is more probable to suffer from congestion. The $DMesh$ topology which uses congestion-aware routing is called CA-$DMesh$.

To evaluate the usefulness of these topologies consider following situation. The longest path in an $N \times N$ mesh network is through the diameter of the network, so a flit in the worst case, has to travel through minimum $2 \times N$ distinct routers if using XY routing algorithm in mesh topology. While, that flit may pass $N$, $2 \times N/3$, $2 \times N/4$, $2 \times N/8$ hops in $DLMesh$, $TLMesh$, $QLMesh$ and $OLMesh$ respectively and $N/2$ in $DMesh$ topology leading to low packet latency. Besides, in regular mesh topology, each router requires five separate buffers to supply the routing [14] which is a total of $5 \times K \times K$ for $K \times K$ routers, which reduces to $(4+K) \times K$ for the $K2Router$. Therefore, it can be observed that the recommended router declines the power and area overhead of buffers and channels by of $(5 \times K \times K - (4+K) \times K)/ (5 \times K \times K) = 0.8 \times (K-1)/K$.

The studies on various traffic patterns on a network of $16 \times 16$ nodes prove that $OLMesh$, $QLMesh$, $DLMesh$ outperform traditional mesh router significantly. However, due to long wire length limitation and signal strength, it shows that using $D2Router$ is reasonable to use compare to other variations of $K2Router$. In this case, the experimental results show that $DLMesh$, $DMesh$ and CA-$DMesh$ outperform mesh in communication latency, energy and throughput. Experimental results reveal that these approaches have better enhancement on power delay product (PDP) as well as throughput than previous architectures. On average, the PDP is enhanced 64%, 65% and 82% for $DLMesh$, $DMesh$ and CA-$DMesh$, respectively. The evaluation on average saturation load proves that 10%, 25% and 38% enhancement is achieved in $DLMesh$, $DMesh$ and CA-$DMesh$, respectively, compared to the regular mesh. Results of implementations show that hardware cost of Macrouter for $DLMesh$ and $DMesh$ topology is 22% less than equivalent base routers in a mesh, with same energy consumption.

The rest of this paper is organized as follow: section 2 discusses related work. Section 3 gives the detailed information about interconnection network architectures based on $K2Router$. Section 4 presents structure and the routing algorithm of $K2Router$. The experimental results are presented in section 5 followed by the concluding remarks in section 6.

2. Related Work

Prior researches have employed variety of interconnects such as shared-bus [15], ring [16], and mesh/touri [17-19]. For several decades, bus has been used as overlay network for communication between nodes. The main advantages of the bus architecture are simplicity and cost-effectiveness. However, for a relatively long bus lines, the intrinsic parasitic resistance and capacitance can be quite significant. Moreover, each IP node connected to the bus adds to this parasitic capacitance, in turn causing increased propagation delay [20]. This fact limits the number of nodes that can be connected to the bus, thus affects the scalability [21].

Hierarchical bus structure [15] introduces an interconnection architecture which reduces local communication overheads, at the expense of cross-chip latencies. This work exploits shared bus, point-to-point link and crossbar interconnection which are efficient for small scale CMPs. It would results in high area and power consumption when the number of nodes increases. Although most of the tiled CMPs have recently utilized 2D mesh topology for its low complexity and short wire length, it has disadvantage of poor scalability for larger conFigurations as a result of increased number of routers and average hop count. TRIPS architecture [11] is an example that used mesh as interconnection network.

Prior researches have described more efficient network architectures than mesh for NoCs. Flattened butterfly [12] uses high radix routers to overcome the scalability
limitations of a mesh. This design reduces the number of channels and routers in the network which results in a more efficient network with lower latency and lower energy consumption. However, when the concentration factor increases, increase in serialization latency becomes a problematic.

In [23], the author has proposed a hybrid topology that breaks a large mesh into smaller meshes connected by a hierarchical ring interconnect for routing global traffic. Their motivation was to reduce average hop counts and latencies for global traffic while still maintaining the high throughput that meshes exhibit for local traffic. The topology benefits from the simplicity of the rings to reduce the complexity at each node which results in reduced buffer, area and energy requirements. However, a problem arises by increasing the number of nodes; therefore, there are some optimal sizes for the sub-meshes before performance degrades to unacceptable levels due to congestion.

The hierarchical network presented in [8] is a two-tiered network topology uses a high bandwidth bus for the local communication and a 2D mesh network for the global communication. Hierarchical design can outperform the other networks in terms of latency, power consumption and energy-delay product specifically with localized communication. The main limitation of the hierarchical NoC is that it saturates faster than other topologies, so is applicable for applications with low injection rates and communication localities.

The diagonally-linked mesh introduced in [24], adds diagonal links to 2D mesh which reduce the distance between a source node and a destination node, therefore it improves the latency. However, this design has high area overhead and energy consumption of near 100% more than regular mesh, which are the main constraints for scalability in NoCs.

Concentrated Mesh (CMesh) network [22] improves the mesh to radix-4 mesh in which each router services four processing nodes. CMesh reduces hop count and improves load balance without increasing wiring complexity. However, there are some differences between their work and this paper as follows:

1) CMesh topology has some express channels along the perimeter of the network to decrease the bisection channel count in the radix of the mesh, while in our proposed technique, DLMesh do not have such express channels and the number of links per each dimension is doubled.

2) In CMesh topology, number of physical channels in each direction is not changed compared to that of traditional mesh topology, while in DLMesh, the number of physical channels in each direction is doubled.

3) In [22], only the performance evaluation of CMesh vs. CMeshX2 is experimented and discussed, while in our paper not only DLMesh is experimented and discussed but also DMesh is experimented and analyzed. Moreover, the effectiveness of K2Router is evaluated for these proposed topologies.

4) In this paper, the effectiveness of K2Router for each of DMesh and DLMesh networks are experimented and evaluated while in [22] CMesh is considered with the traditional router.

![Network Topologies for 64-nodes NoCs](image)
3. Interconnection Network

In a large CMP system, to overcome performance and power overhead, the on-chip network needs to minimize communication cost. Clustered communication approach is a solution to address the problem by limiting most of the communication within a small set of nodes in a cluster. This motivates us to design an interconnection network which benefits from communication locality.

3.1. Network Topology

This part is devoted to analyze four topologies namely mesh, DLMesh, DMesh and QLMesh to understand their impact on area, power and performance. Figure 1 shows possible layouts for mesh, DLMesh, DMesh and QLMesh, respectively, with 64 nodes.

Mesh topology, shown in Figure 1(a), has been the most popular topology so far for NoC [10, 17-18]. In this topology each PE is directly linked to a distinct router, and the router is connected to four neighbor routers in a structured manner. The main disadvantage of mesh is power and performance degradation by increasing the network diameter which makes it inefficient for larger networks.

Figure 1(b) illustrates DLMesh. Each router in this topology manages up to four PEs. The router then connects to other clusters of nodes by double links for each direction north, south, west and east. Each PE in a cluster can connect to other PEs in that cluster directly via router. Selection between double links in each direction is random, so the traffic is distributed fairly between them. DLMesh reduces number of routers resulting in reduced hop count and thus reaching excellent latency saving over mesh. As it has 12 radix (number of ports), it provide wider channels, hence on average, a PE will not wait for reserving a channel more than a PE in custom mesh. It also solves the scalability problem by sharing a router between multiple injecting nodes.

The third shown topology, called DMesh, is shown in Figure 1(c) Router structure in this topology is same as DLMesh. The only difference is in the interconnection. In DMesh, each router connects to 8 other neighbors by a single link for each direction: north, north-east, north-west, south, south-east, south-west, west and east. DMesh reduces hop count by both sharing a router with four PEs and also extra links to four more neighbors with diagonal links in compare to other two topologies. DMesh has 12 radixes like DLMesh but with better performance.

The fourth topology shown in Figure 1(d) is QLMesh. This router is same as DLMesh but connects 16 PEs as locals to each router while number of link channels is doubled compared to DLMesh. Although, in this router structure, we have k channels in each directions and we can labeled them as north, north-east1, north-east2, east, south-east1, south-east2, south, south-west1, south-west2, west, north-west1, and north-west2. However, we labeled them as k \((k=4)\) different channels in each directions.

3.2. Communication Latency

A common approach to decrease the communication delay is to reduce the average hop count. Figure 1 illustrates how K2Router can improve communication latency compared to traditional router. The longest path for a packet to travel in mesh is from node S to node D when the routing algorithm is XY which is shown with dashed line in the Figure 1(a). The paths which packets travel in the worst case in other topologies are also depicted in Figure 1(b), 1(c) and 1(d). We can find that in this situation the DMesh topology has the best result when D2Router is used. We conclude that in general case for a network with \(N\times N\) nodes, the worst case hop count is \(2\times(N-1)\) for mesh, \((N-2)\) for DLMesh, \((N/2-1)\) for DMesh and \((N/2-2)\) for QLMesh.

3.3. Communication Locality

Tiled CMP architectures may have two important localities: 1) cache-level locality and 2) network-level locality. In the former tiled CMPs where L2 cache is shared, processors communicate with each other through the cache without the external network on chip, while in the latter tiled CMPs, they communicate with each other using message passing through the external network-on-chip. For the former one, the network-on-chip does not play a key role for improving the performance of the system, while in the latter one; the performance of the network significantly impacts the performance of the whole architecture. The proposed router and topologies are suitable for architectures whose locality is defined as network-level. Here, locality is the percentage of injected packets by a node to its immediate neighbors as destination. So, a network topology which supports locality has high local traffic and its nodes have more communication with their nearest neighbors. Hence, it is desired to design a network topology which can support high local bandwidth and low local communication latency for networks with high locality.

A mesh has narrow channels and each flit has to pass through at least one hop to reach out one of its neighbors. In contrast, DLMesh and DMesh designs are better for locality since local PEs can directly communicate with each other through shared K2Router and the local traffic does not distribute through the whole network. This inclines the channel bandwidth.

4. Router Structure

The structure of K2Router with its components is shown in Figure 2 Flits from four local PEs are sent to four Local Buffers and other flits from neighbors reach other buffers depend on their arriving path. Flits wait for routing in Input Buffers. In DLMesh, North and Northeast, East and Southeast, South and Southwest, West and Northwest Input Buffers are connected respectively to north, east, south and west neighbors; whereas in DMesh they are connected to north, northeast, east, southeast, south, southwest, west and northwest neighbors. The connections on DLMesh make the D2Router compatible with regular mesh topology, means that a D2Router can be easily replaced with a group of four routers if they have square structure as depicted in Figure 3 In this condition the reminder part of the network does not need to change.

Routing Unit reads flits from input buffers and after specifying the suitable output port for each input buffer, the Reservation Unit reserves it if the output port is not reserved by any other port. The reservation information is stored in
Figure 2. The K2Router structure consists of Input Buffers, Routing Units, Reservation Units, Crossbar Switch and Output Ports (a) K2Router (general case) (b) D2Router (special case)

reservation table to be accessible by crossbar switch. Crossbar Switch afterward connects input buffers to output ports depend on the reservation feedbacks.

4.1. Routing Algorithm

The dimension-order routing, like XY or YX [25], in which the order of the dimension traversal is constant, can be changed to adapt to DLMesh and DMesh networks. In DLMesh, the direction of a packet is chosen according to the dimension-order routing and the output port is then selected at random between double links in each direction to evenly distribute the load over them. Figure 4 demonstrates how the routing algorithm works in DLMesh.

The dimension-order routing is extended to eight directions instead of four to be used in DMesh network. Figure 5 shows how the routing is implemented in DMesh. In this network the packets can travel through diagonal path (northeast, northwest, southeast and southwest) as well as horizontal (east and west) and vertical (north and south) paths. As, we could conceivably reduce the average number of hops packets travel by preferentially routing over the additional channels because each diagonal link is equivalent to two horizontal and vertical links.
The both routing algorithms are deadlock free because they use dimension order routing and therefore, introduce no additional constraints on the flow control policy. The ordering; however, results in a unique path from source to destination and thus does not allow adaptive routing.

Figure 3. A K2Router can be replaced with 4-grouped and 9-grouped routers for k=2 and k=3, respectively

Figure 4. The extended dimension-order routing algorithm for DLMesh

Figure 5. The extended dimension-order routing algorithm for DMesh

Figure 6. Different turn models in the routing algorithm proposed for DMesh

Figure 7. Inhibited turns in the proposed algorithm to prevent possible cyclic dependencies
4.2. Deadlock Analysis of the Proposed Routing Algorithms

The routing algorithms proposed for DMesh and DLMesh networks in Figures 4 and 5 are deadlock free. Because, in the case of DLMesh network, the algorithm is like XY algorithm, each channel selection is based on next direction of packets in the X and Y axes. Moreover, the transmission of packets in each direction is independent to opposite direction and physically separable.

The routing algorithm proposed for DMesh, is also deadlock free since there would no cycle in dependency graph based on turn model. Figure 6 shows all turns that would be happened when this algorithm is used. Based on these turn model, there would no cycle dependencies among routers’ buffers and therefore, this algorithm would not cause deadlock. As shown in Figure 7, there are no turns in the proposed algorithm and it prevents possible cyclic dependencies.

4.3. Congestion-Aware Routing Algorithm

Figure 8 illustrates different possible paths for diagonal links (a) and direct links (b) in DMesh where S is the source node and D is the destination node. In both, path 1 is the minimal path which can be specified by the minimal routing algorithm mentioned in previous subsection. Routers using minimal routing algorithm may encounter congestion in some of their links depends on the traffic load. For example, for transpose traffic, diagonal links may suffer from congestion, while other links have low utilization.

Deterministic routing may not be able to react to congestion due to network bursts, and consequently results in an increase in network delay. Adaptive routing algorithms employed in traditional networks as a solution to congestion avoidance, are more suitable for NoC implementations [13]. Congestion control routing algorithms distribute the traffic among links [13]. In Figure 8, it allows packets to leave the source via path 2 and 3 in the existence of congestion. The packets may pass longer path, however this method helps avoiding packet drop and additional latency caused by congestion. Common congestion control algorithms monitor buffer or link status to avoid resource starvation or congestion [26]. In this paper, congestion control algorithm inspects buffer status of immediate neighbors to find the most appropriate output port. To avoid deadlock, non-minimal paths are selected by probability of P, means even if a non-minimal path is more suitable for controlling the congestion, it is selected by probability of P; otherwise the minimal path is selected.

4.4. Router Implementation Cost

In order to get precise values for area and energy consumption of the router, we used a VHDL model of a traditional router design [32], and then we modified it so that a D2Router was implemented. Next, the results of Synopsys tool chain for 65nm technology were obtained. The router parameters and their corresponding value are shown in Table 1.

Table 1. Router parameter for Synopsys tool

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>65nm</td>
</tr>
<tr>
<td>VDD Voltage</td>
<td>1.21 V</td>
</tr>
<tr>
<td>Wire Model</td>
<td>UMCL65SP_LOWK_M50</td>
</tr>
<tr>
<td>Flit Size</td>
<td>128 bit</td>
</tr>
</tbody>
</table>

Table 2 shows delay, area and power consumptions of D2Router and traditional mesh router according to the Synopsys tool results. Based on values reported in Table 2, D2Router design imposes much times delay, area and power consumption compared to traditional router, however, it plays roles of four grouped routers in traditional network, so for sake of fairness, it should be compared to four connected traditional routers and located in second row of Table 2. It shows that using D2Router, delay, area and power consumption compared to traditional router, however, it plays roles of four grouped routers in traditional network, so for sake of fairness, it should be compared to four connected traditional routers and located in second row of Table 2. It shows that using D2Router, delay, area and power consumption can be reduced about 31%, 24% and 43% compared to four connected traditional routers, respectively. Therefore, using D2Router in the networks is cost-effective and low power overheads compared to having four connected traditional mesh routers.

Figure 9 (a) illustrates the area breakdown and 9 (b) shows the energy consumption breakdown in a regular mesh router. The same Figureures for D2Router are depicted in Figure 10. With growth in deep submicron technology, the power consumption and area overhead of links become noticeable. This Figureure depicts that links’ portion in consumed area and power of overall router is 61% and 35%, respectively. This means, reducing the links in a network will improve the area and power consumption of the network considerably. This can be seen in Figure 10, where link area and power contributions to overall router are mitigated to 54% and 29%, respectively.
Table 2. Comparing traditional router versus D2Router

<table>
<thead>
<tr>
<th>Router design</th>
<th>Type</th>
<th>Delay (ps)</th>
<th>Area (um²)</th>
<th>Power (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional Mesh Router</td>
<td>Simple</td>
<td>781</td>
<td>100</td>
<td>1.75</td>
</tr>
<tr>
<td>4 Connected Traditional Mesh Router</td>
<td>Grouped</td>
<td>1668</td>
<td>214</td>
<td>7.00</td>
</tr>
<tr>
<td>D2Router</td>
<td>Grouped</td>
<td>1149</td>
<td>147</td>
<td>3.95</td>
</tr>
</tbody>
</table>

Table 3. Occupied area of internal parts of routers (um²)

<table>
<thead>
<tr>
<th></th>
<th>D2Router</th>
<th>Traditional mesh router</th>
<th>#</th>
<th>%</th>
<th>#</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer</td>
<td>11119</td>
<td>16</td>
<td>4633</td>
<td>21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crossbar</td>
<td>16023</td>
<td>23</td>
<td>2264</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual Channel</td>
<td>3483</td>
<td>5</td>
<td>1383</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>1161</td>
<td>2</td>
<td>462</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Link</td>
<td>37962</td>
<td>54</td>
<td>13780</td>
<td>61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>69748</td>
<td>100</td>
<td>22522</td>
<td>100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Power consumption of internal parts of routers (mW)

<table>
<thead>
<tr>
<th></th>
<th>D2Router</th>
<th>Traditional mesh router</th>
<th>#</th>
<th>%</th>
<th>#</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer</td>
<td>1.36</td>
<td>35</td>
<td>0.82</td>
<td>47</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Crossbar</td>
<td>1.13</td>
<td>29</td>
<td>0.29</td>
<td>16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual Channel</td>
<td>0.07</td>
<td>2</td>
<td>0.005</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>0.23</td>
<td>6</td>
<td>0.015</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Link</td>
<td>1.15</td>
<td>29</td>
<td>0.62</td>
<td>35</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>3.95</td>
<td>100</td>
<td>1.75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 9. (a) Area and (b) power consumption breakdown for a traditional router

Figure 10. (a) Area and (b) power consumption breakdown for a D2Router

Tables 3 compares area of D2Router with that of traditional mesh router. It should be noted that D2Router is responsible for four regular mesh routers; therefore, the number of buffers in D2Router is twelve compared to twenty buffers in four regular mesh routers. Moreover, relative contributions of buffer area to crossbar area in traditional router is 21% by 10% or about 2 times, however this is 16% by 23% or about 0.7 times in D2Router. This is because of the fact that crossbar is one of the main component of area in D2Router when number of ports increased. In contrast, crossbar increases the router area by 13%. Growth in crossbar area shows that the main constrain in this design is the complexity of crossbar when the number of PEs increases. It means that this implementation may have area
overhead and more complexity for more than four PEs. The D2Router structure lessens the area of the router 22\% compared to four traditional mesh routers.

Table 3 evaluates the power consumption of D2Router with traditional mesh routers. Similar to area evaluation, power of buffer and link in D2Router is less than regular mesh which gratifies the increased power by crossbar.

5. Performance Analysis

In this section we analyze the extracted results of experiments to find out the behavior of the proposed network-on-chip architectures under variant traffic loads.

5.1 Selection of Experiment Parameters

We have performed experiments using wormhole switching with a constant packet size of 10 flits. The maximum bandwidth at each link was set to 1 flit per cycle. Switches had an input buffer size of 4 flits and random selection policy at the arbiter for adaptive routing algorithms. Buffer level selection policy was also considered for DMesh. We have used XY routing algorithm for mesh, and extended XY for DLMesh and DMesh. The packet injection rate was varied between 0.001 and 0.03 (packet/cycle/node). Different traffic pattern were used consist of uniform (random), bit-reverse, transpose1 (matrix transpose), transpose2, butterfly and shuffle to measure the performance. Under the pattern transpose1, a PE at (i, j) (i, j C [0, E]) only sends messages to node (E-1-i, E-1-j); under the pattern transpose2, a node (i, j) only sends messages to node (j, i) [27]. Each simulation was executed for 2000 clock cycles for warm-up and then continued for 10,000 clocks for extracting results. In our experiments, we have continued each experiment so that simulation results become stable; the empirical results showed that 10,000 cycles after a warm-up session 2,000 cycles, are enough to get stable results from simulations. In the experimental results, the 95 percent confidence intervals are mostly within 2 percent of the means.

We have evaluated the networks based on their area, communication latency and energy consumption using a cycle-accurate mesh-based NoC simulator called Noxim [28-29]. We have modified Noxim to implement the two suggested topologies. For each network, we also evaluate the impact of varying the locality percentage to determine the conFigureuration providing the most competitive area, latency and energy efficiency.

Performance and communication power of a NoC is influenced by the packet injection rate and traffic pattern. We have analyzed latency, communication power and throughput under plenty of traffic patterns. The effect of congestion-aware routing algorithm (CA-DMesh) on DMesh performance is also analyzed.

5.2. Network Latency

In order to find the effect of K in K2Router on the performance of a network with size of 16×16, various simulations were done and the effect of K in the network is demonstrated in Figure 11 In this figure, K1Mesh, K2Mesh, K4Mesh and K8Mesh, denotes to traditional mesh, DLMesh, QLMesh and OLMesh respectively. These names are chosen so that it can be easily to compare the results based on value of K. For all traffics, packet latency K2Router-based networks decrease when value of K increase. Moreover, saturation load of network increase when value of K increase. It should be noted that, although having more value of K is desirable for improving performance of K2Router, however, because of some limitations in wire length and signal strength, it may not be possible to layout networks with K2Router for large value of K. This limitation would be relaxed when these topologies used in 3D layouts.

Without loss of generality, we have used K2Router with K equal 2, i.e., D2Router, for coming analysis. It is notable to say that for larger value of K, the performance of K2Router would be better than D2Router. In the case of using D2Router, Figure 12 illustrates packet latency for different topologies. On average, before network saturation, the average packet latency in DLMesh and DMesh has decreased in compare to regular mesh. Reduction in average delay is because the dimensions in the proposed topologies are half of the mesh dimension with equal PEs. Also, packet latency is cause of queuing time, transitions time and blocking time. Queuing time is the main constraint for DMesh to outperform DLMesh as it is expected. Each node in DLMesh is connected to its neighbors by two links which reduces the queuing time and packet latency consequently. Except random traffic, for other traffics, the CA-DMesh has improved the latency much more than other topologies. For random traffic pattern, the enhancement is less because it is more probable of congestion.

5.3. Network Throughput

Figure 13 depicts the comparison of throughput among mentioned topologies. Throughput means average number of received packet per each cycle in each router. Since each D2Router is responsible of four tiled nodes, the results of traditional mesh router are multiplied by four for sake of fair comparison. Results show that throughput has been same as mesh and somewhere outperformed in DMesh and DMesh.

The saturation load can also be extracted from throughput graph which is a point where throughput no longer grows linearly with packet injection rate. Figure 14 shows how saturation load is modified in different topologies. It shows that for all traffics the saturation load of DMesh, DLMesh and CA-Mesh are increased, while for bit-reverse, DMesh has negative impact on saturation load. On average, the saturation load is improved 10\%, 24\% and 38\% in DLMesh, DMesh and CA-DMesh, respectively, compared to regular mesh.

5.4. Communication Energy

Figure 15 compares the communication energy in the topologies. In all traffic patterns, the average communication energy is declined for DMesh, DLMesh and CA-Mesh. This is essentially because they reduce the average hop count. The CA-DMesh degrades the consumed power the most. The figure shows that the communication power of DLMesh topology is more than DMesh because the diagonal links in DMesh shortens average packet path, then less energy is consumed in links and buffers compared to others.
Figure 11. Average packet latency for different traffic patterns
Figure 12. Average packet latency for different traffic patterns
Figure 13. Throughput of different traffic patterns
Figure 14. Saturation load

Figure 15. Consumed energy for different traffic patterns
Power delay product (PDP) in Figure 16 shows the trend across 4 topologies and the values are averaged for all injection rates just below saturation point. All the values are normalized with respect to mesh. CA-DMesh topology has the least PDP for all traffic patterns and DLMesh and DMesh decrease PDP for all traffics. Under random traffic, DLMesh improves PDP by 67%, DMesh by 78% and CA-DMesh by 78% over mesh. For bit-reverse traffic, DLMesh, DMesh and CA-DMesh do respectively 65%, 72% and 78% better than mesh. For transpose1 traffic, they improve PDP by 70%, 78% and 92%, for transpose2 by 68%, 76% and 86%, for butterfly by 45%, 44% and 70%, and finally for shuffle traffic, PDP is enhanced by 65%, 44% and 70%, for CA-DMesh by 78% over mesh. For transpose1 traffic, they improve PDP by 70%, 78% and 92%, for transpose2 by 68%, 76% and 86%, for butterfly by 45%, 44% and 70%, and finally for shuffle traffic, PDP is enhanced by 65%, 44% and 70%, for CA-DMesh by 78% over mesh. For transpose1 traffic, they improve PDP by 70%, 78% and 92%, for transpose2 by 68%, 76% and 86%, for butterfly by 45%, 44% and 70%, and finally for shuffle traffic, PDP is enhanced by 65%, 44% and 70%, for CA-DMesh by 78% over mesh. For transpose1 traffic, they improve PDP by 70%, 78% and 92%, for transpose2 by 68%, 76% and 86%, for butterfly by 45%, 44% and 70%, and finally for shuffle traffic, PDP is enhanced by 65%, 44% and 70%, for CA-DMesh by 78% over mesh.

On average, overall traffics, DLMesh, DMesh and CA-DMesh outperform mesh. Results show that DLMesh and DMesh have almost same PDP and CA-DMesh outperforms them.

5.5. Communication Locality

The effect of locality in communication latency and consumed energy under packet injection rate of 0.01 is depicted in Figure 17. Locality favoring topologies like DMesh and DLMesh consistently show latency lower than mesh. This is mainly because the global hop count is reduced in these topologies. In addition, local PEs which is connected by a single D2Router can communicate directly without using the rest of the network. For both energy and latency, CA-DMesh shows better results. The Figure reveals that averaged over all the locality percentage, the mesh latency is declined 44% in DLMesh, 34% in DMesh and 50% in CA-Mesh. The consumed energy is reduced 48% in DLMesh, 49% in DMesh and 60% in CA-DMesh. This level of reduction in delay and consumed energy make these topologies a good choice for applications with high locality.

6. Conclusions

This paper presented two topology architectures, called DLMesh and DMesh, which benefits from communication locality. These topologies use D2Router to communicate with other nodes in the network. Each D2Router is connected to eight neighbors and four local PEs. To avoid congestion, congestion aware routing algorithm is considered for DMesh which is called CA-DMesh. Some experiments have been studied comparing the topologies with a typical 8x8 mesh topology with 64 individual PEs. In this work, the effect of proposed topologies in communication latency, power and area of a network-on-chip has been demonstrated. The experimental studies proved that DLMesh and DMesh outperform mesh in communication latency, energy and throughput. Experimental results reveal that these approaches have better enhancement in power delay product (PDP) as well as throughput than previous architectures. On average, overall traffics, DLMesh, DMesh, CA-DMesh...
performs 5, 5 and 6 times, respectively, better in throughput than mesh before network saturation. The results show that DLMesh has enhanced PDP 64%, DMesh 65% and CA-DMesh 82% compared to traditional mesh. It means that DLMesh and DMesh have almost same PDP and CA-DMesh outperforms them. Also, with increase in locality percentage from 10 to 90, the average mesh latency is declined 44% in DLMesh, 34% in DMesh and 50% in CA-Mesh. This shows that for high locality among nodes using clustered on-chip networks becomes more crucial. Therefore, DLMesh and DMesh gain better results in compare to regular mesh.

References


Hananeh Aliee received both her B.Sc. and M.Sc. degrees in department of computer engineering and information technology at Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran in the years of 2009, and 2011, respectively. Her main fields of interest are Fault-Tolerant System Design, Reliability Analysis, and Computer Architecture. She has joined as a researcher in the chair of Hardware-Software Co-Design at Erlangen-Nürnberg University, Erlangen, Germany in 2011.

Email: h.aliee@aut.ac.ir

Hamid R. Zarandi received his B.Sc., M.Sc., and Ph.D. degrees all in department of computer engineering at Sharif University of Technology (SUT), Tehran, Iran, in 2000, 2002, and 2007, respectively. He is currently an assistant professor in computer engineering and information technology department at Amirkabir University of Technology (AUT) (Tehran Polytechnic), since 2006. His research interests include dependability evaluation using fault injection techniques, fault-tolerant computing, dependable computer architecture, high performance computing, and fault-tolerant embedded systems, on which he has published more than 90 referred conference and journal papers. Dr. Zarandi established the “Design and Analysis of Dependable Systems (DADS)” laboratory at Amirkabir University in 2006, and has chaired the laboratory since then. He is a member of the IEEE Computer Society, and the Computer Society of Iran (CSI).

Email: h_zarandi@aut.ac.ir

Pooria M. Yaghini is a Ph.D. candidate in department of electronic engineering and computer science at University of California, Irvine. He has received his B.Sc. in computer hardware Engineering from Sajjad Institute of Higher Education in 2006 and his M.Sc. in department of computer engineering and information technology at Amirkabir University of Technology (Tehran Polytechnic) in 2009. His research interests include 3D IC design, on-chip networks, low-power design, reliability analysis and verification, and fault-tolerant architecture design.

Email: pooria.yaghini@aut.ac.ir

Paper Handling Data:
Submitted: 18.05.2011
Received in revised form: 17.08.2011
Accepted: 05.10.2011
Corresponding author: Dr. Hamid R. Zarandi, Department of Computer Engineering and Information Technology, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran.