International Journal of Future Generation Communication and Networking
Vol. 8, No. 1 (2015), pp. 19-26
http://dx.doi.org/10.14257/ijfgcn.2015.8.1.03

Heterogeneous Honeycomb-like NoC Topology and Routing based on Communication Division

Pengfei Yang and Quan Wang

School of computer Science and Technology, Xidian University
pfyang@stu.xidian.edu.cn, qwang@xidian.edu.cn

Abstract

Though the mostly used topologies in network-on-chip (NoC) based systems, mesh and torus can lead to a waste of resource and bandwidth when the processing elements(PEs) demand less communication, and fail to meet the requirements of PEs demanding mass communication since they connect all the PEs by the paths of the same type. In this paper, a configurable network and an efficient routing algorithm is designed so that the topology is regular and easy to expend while the adjustable density of transmission roads could satisfy the varied requirements of different PEs. Experimental results have shown that our design is more desirable in terms of network delay, power consumption and area cost among other important performance parameters of a network.

Keywords: *network on chip; hexagonal topology; heterogeneous division*

1. Introduction

As more and more general-purpose/application-specific processing elements (PEs) have been introduced into systems-on-chip (SoC) with the emergence of diverse compute-intensive applications over the past few years[1, 2], it has been increasingly more complex to achieve the communication between such different types of PEs with the limited resources on a chip. In such circumstances, with the advantage of easy expandability, small delay and high communicating efficiency, multiprocessors based on network-on-chip(NoC) have been replacing the sharing bus-based traditional design [3-6] since the conception of NoC was raised by researchers of Royal Institute of Technology(KTH). NoC borrows the concepts and techniques from the well established computer networks, and it uses routing and switching technology to implement the communication among PEs.

Among the various NoC topologies today, mesh [7-9] and torus [10-13] are the mainstream for their high regularity, symmetry and scalability. But with the development of IC technology, they are challenged by the increasing heterogeneity of applicable new PEs on a SoC such as hardware accelerators for specific Digital Signal Processing kernels, high-performance DSP cores, low-power application processors as well as other logic blocks and memory being implemented on a single chip. In a mesh or torus topology, less or no considerations are given to the heterogeneity in the communication demand of different PEs, so all the PEs is connected by the paths of the same type. On one hand, the connection leads to a waste of network bandwidth and on-chip resources in areas where the PEs actually demand less communication. On the other, it blocks the network operation when some PEs demand more communication but are not supplied with sufficient bandwidth and on-chip resources. Both of these non-equilibriums of communication demand and supply will restrict the performance of the SoC.

A new network topology is proposed in this paper, which uses regular hexagons as its basic components. In areas with large communication demand an exchange node will be placed at the hexagon's center to connect the six vertexes and provide more data

ISSN: 2233-7857 IJFGCN

transmission paths. On the basis of the topology, we then design a routing algorithm to find out the best transmission path between a source node and a destination node in a short time with a small amount of system resources. With high regularity, symmetry and scalability, the whole network proves able to improve the communication efficiency and to reduce the on-chip resource consumption.

The rest of this paper is organized as follows. Section II discusses the topology and its coordinate representation. Section III presents the routing algorithm for the topology. A comparative experiment result is shown in Section IV. Section V concludes the paper.

2. Topology of Honeycomb-like Network

The topology of the network is shown in Figure 1. A hexagon whose every vertex presents a switching node is the basic component of the network. For the 2D NoC, generally a switching node only connects to its three neighbor nodes and local PEs. But in the area with busy traffic, to meet the large communication demand of intensive communicating units, an additional switching node is placed in the hexagon's center which only connects to the hexagon's six vertexes to provide more data transmission paths. We call the area containing this kind of nodes "Communication-intensive Region", while the rest area is called "Communication-sparse Region".

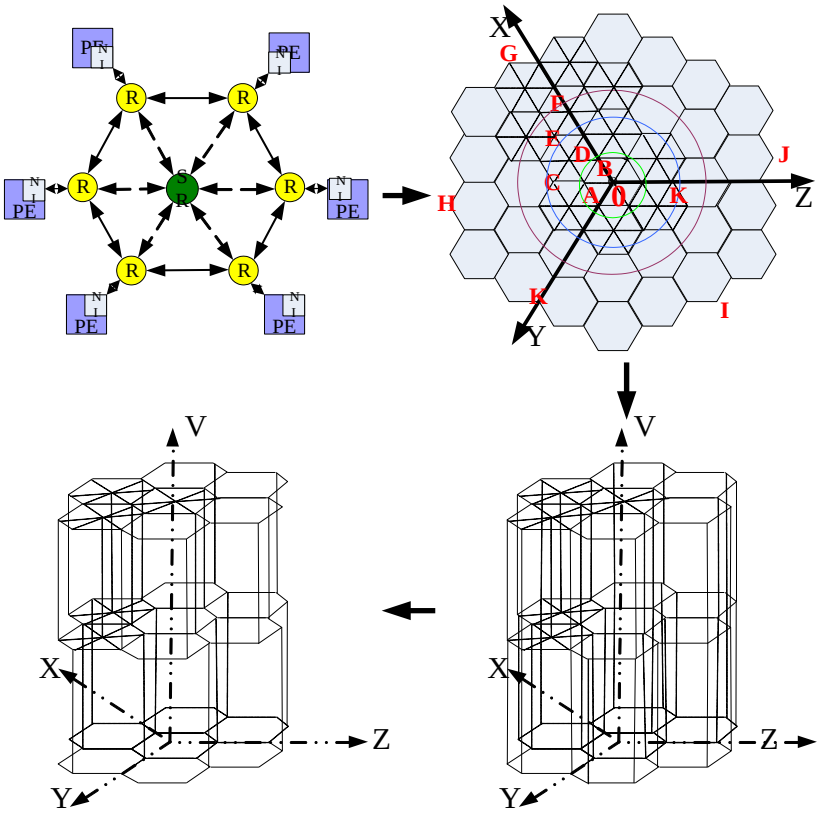


Figure 1. The Topology of NoC

The 3D NoC is formed by vertically stacking a number of 2D NoCs with every switching node except the ones on the bottom and top being connected to its upper and lower immediately adjacent switching nodes.

However, because of the large number of vertical communication links, and in terms of the fabrication feasibility, the vertical links are much more expensive and area consuming when compared with the horizontal links [14]. So we improve the architecture by

removing half of the vertical links in the architecture which has a high implementation cost.

Generally, a topology is evaluated in terms of four parameters.

1) Degree: the number of I/O channels per node;
2) Network Diameter: the maximum value of all shortest paths between any two nodes in the network. Generally, a small degree means a large network diameter, so we define:
3) Network Cost: the product of the degree and the network diameter.
4) Bisection Width: the minimum number of wires that must be cut when the network is divided into two equal sets of nodes.

Table 1 is a comparison of the 2D honeycomb-like topology and the common 2D topologies. It can be seen that the honeycomb-like topology has obvious advantages in terms of the cost.

Table 1. Comparison of Topology (Each with n Nodes)

Types of topology		Degree	Diameter	Cost	Bisection width
Honeycomb-like	Sparse region	3	$1.63\sqrt{n}$	$4.90\sqrt{n}$	$0.82\sqrt{n}$
dense region	Intensive region	6	$1.16\sqrt{n}$	$6.93\sqrt{n}$	$2.31\sqrt{n}$
Mesh		4	$2\sqrt{n}$	$8\sqrt{n}$	$\sqrt{n}$
Torus		4	$\sqrt{n}$	$4\sqrt{n}$	$2\sqrt{n}$
Butterfly		4	O(log n)	O(log n)	O(n/log n)
De Bruijn		4	O(log n)	O(log n)	O(log n)
Hypercube		log n	log n	$\log^2 n$	log n

2.1. Coordinate Representation

As shown in Figure1, the network adopts XYZ coordinate in 2D NoC and XYZV coordinate in 3D to code the nodes. With the Z axis being the horizontal axis, the X, Y, Z axes are 120 degrees angle to each other and are parallel to the three edge directions of a hexagon. It should be noted that for consistency of indication, switching nodes are assumed to exist at the center of every hexagon.

The network is coded as follows. the vertical axis V is used to indicate the layer on which a switching node is located. The X coordinate of a point is the value of the intersection point of the X axis and the straight line which passes the point and parallels to the Y axis. The Y coordinate is the value of the intersection point of the Y axis and the straight line which passes the point and parallels to the Z axis. The Z coordinate is the value of intersection point of the Z axis and the straight line which passes the point and parallels to the X axis. Here blow is an illustration of the coordinate representation using points in 2D NoC shown in Figure1.

The origin point 0: O (0, 0, 0);

Points on the one jump circle: A (1, 0, -1), B (1, -1, 0);

Points on the two jump circle: C (2, 0, -2), D (2, -1, -1), K(-2 ,0 , 2);

Points on the three jump circle: E (3, -2, -1), F (3, -3, 0);

A Point on the five jump circle: L (0, 5, -5);

Points on the six jump circle: G (6, -5, -1), H (5, 1, -6), I (-6, 5, 1); J(-5, -1, 6)

By observing the coordinates above we can get the following rules (which can also be proved by simple geometric relationships):

1) The sum of X, Y, Z coordinate values of each point is 0;
2) The largest absolute value of X, Y, Z coordinates of each point indicates the minimum number of jumps from the origin to its very position.

The similar circle composed by points with the same jump number is called an N jump circle. As shown in Figure 1, the green circle is the one jump circle, the blue is the two jump circle, the purple is the three jump circle, and so forth.

2.2. Communication-intensive Region Distribution

According to the ratio of the calculation amount to the communication amount, a task can be classified as a calculation-intensive task or a communication-intensive one that requires higher communication performance of the network. So to meet the transmission requirement, on the honeycomb-like network the center node of each Communication-intensive Region should also be able to transmit data packets to its six neighbor nodes.

Generally, the core graph is shown by a directional graph G = (V, E). Each vertex $v_i \in V$ represents a PE, the directional edge $e_{i,j} \in E$ stands for connection between v_i and v_j, and the weight of $e_{i,j}$ shown as $COM_{i,j}$ represents the data to be transmitted from v_i to v_j. Here, a parameter CCR is defined in Formula 1 to help determine whether a task is communication-intensive or calculation-intensive.

$$CCR = \frac{\frac{1}{e}(\sum_{i=1}^{e} C_i)}{\frac{1}{v}(\sum_{j=1}^{v} U(n_j))} \tag{1}$$

where e equals to |E| and represents the number of connection between any two cores; C_i means the communication time of message exchange; v=|V| is the number of cores in the task; $U(n_j)$ means average runtime of task n_j which is calculated by Formula(2) below.

$$U(n_j) = \frac{1}{v}(\sum_{i=1}^{v} T(n_j, P_i)) \tag{2}$$

Here $T(n_j, P_i)$ means the runtime of P_i to carry out task n_j which is assigned to P_i.

For a core graph, a larger CCR means more intensive communication. But in practice, a threshold value should be set according to the requirement so that any tasks with CCR greater than the threshold will be assigned to the Communication-intensive Region.

3. Routing Algorithm

In this section, we describe the deadlock free routing algorithms on 2D and 3D topology respectively.

For the 2D topology, as we know, the shortest path between two points is the straight line between them. And when communication is implemented along this line, the system latency could be minimized, a situation which however is rare in the honeycomb-like topology(only when the first point's at least one coordinate equals to the corresponding coordinate(s) of the second and then the stitch nodes between the two points exist). Aiming to find the paths mostly approximate to a straight line between any two points, the routing algorithm is divided into two steps:

Firstly, form a region for candidate paths. The region is composed of overlapped parts of three areas that are determined respectively by inequalities Min(Xs, Xd)≤X≤Max(Xs, Xd), Min(Ys, Yd)≤Y≤Max(Ys, Yd) and Min(Zs, Zd)≤Z≤Max(Zs, Zd). Just as shown in Figure 2, point S is the source node, point D is the destination node and the candidate region is the area fenced by black lines.

Secondly, in the candidate region, start from the source node and use the Dijkstra algorithm to find the shortest path between the source and destination nodes. On one hand, as the paths chosen are multifarious, the probability of resource contention could be reduced. On the other, the algorithm could take evasive action to avoid involving the busy nodes into the path.

As shown in Figure 2, in the region, S will arrive at vertex D1 at one side of the region and the path from S to D1 is the shortest. The shortest path from D1 to D also exists, so the shortest path from S to D certainly exists in the candidate region.

For the 3D topology, the routing algorithm first compares the V coordinate of the source node with that of the destination node. If the two V coordinates equal, the algorithm needs no vertical routing and is reduced to the 2D routing algorithm mentioned above. If the two V coordinates differ, the routing algorithm first moves the source node to the layer of the destination mode, and then executes the 2D routing algorithm.

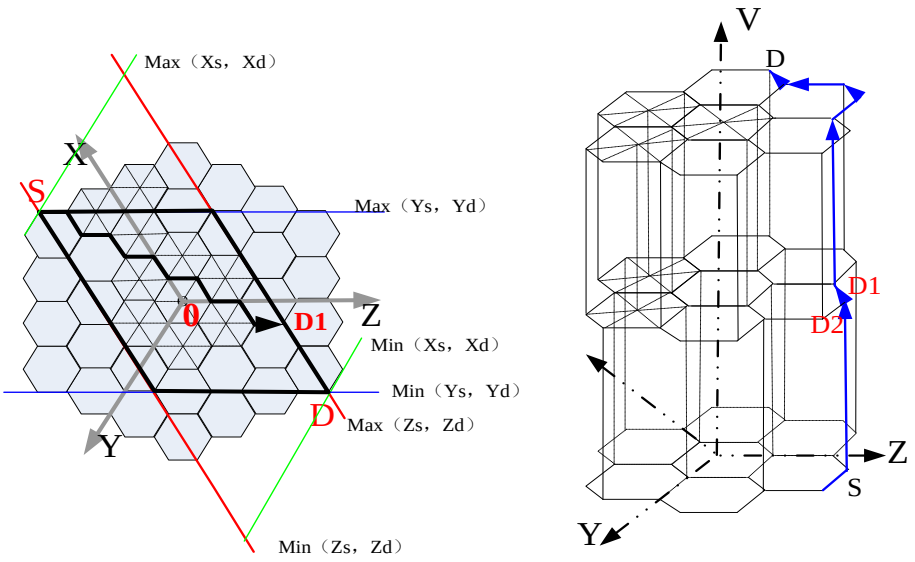


Figure 2. Routing Algorithm

4. Experiment and Testing

Power consumption and packet delay are the most important parameters that reflect network performance. To verify the advantage of the topology and the routing algorithm, we have made a comparative experiment between a honeycomb-like topology with 24 PEs and a regular mesh topology with 25 PEs. We use NIRGAM [15] emulator as the experimental environment which is developed by research and development group of Professor Bashir Alhashimi of Electronics and Computer Science School from Southampton College. As the emulator already supports mesh topology and XY routing algorithm, we just need to additionally implement the honeycomb-like topology and the routing algorithm in it.

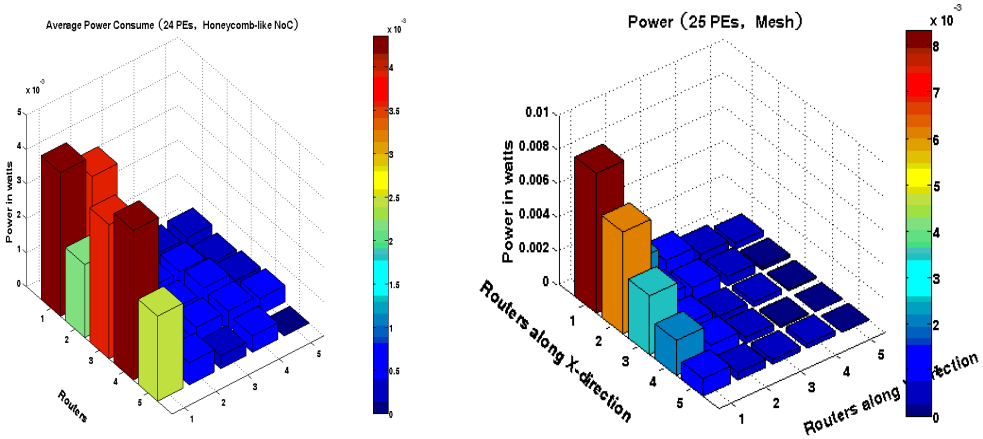


Figure 3. Average Power Consumption

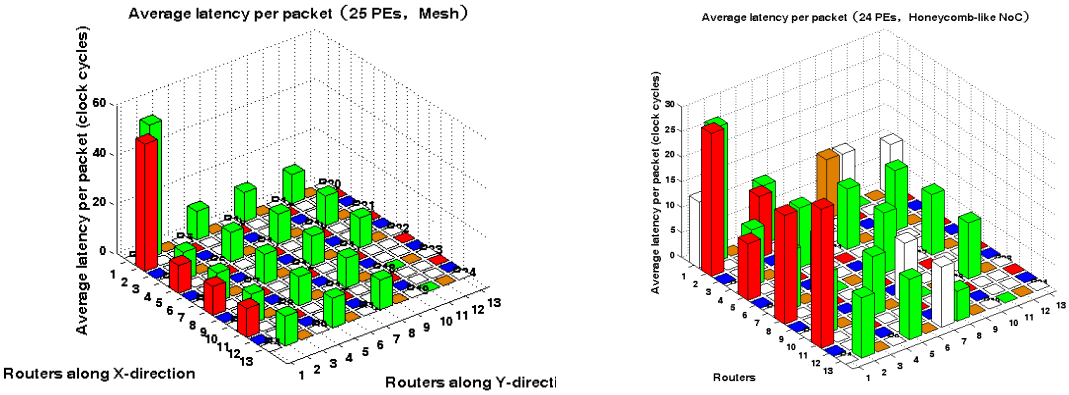


Figure 4. Average Latency per Packet

The results are shown in Figure 3 and Figure 4. It is obvious that the honeycomb-like topology is significantly better than the Mesh structure in terms of both average latency and power consumption. Besides, no router is much busier than others.

5. Conclusion

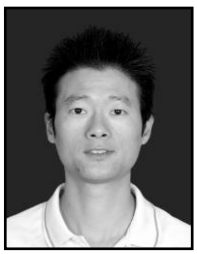
In this thesis, we prove that Honeycomb-like topology is an advantageous alternative for NoC based systems. Its system cost is approximately 20% lower than that of a mesh

and its communication delay is reduced by at least 30% compared with XY routing algorithm in a mesh network.

References

[1] K. Goossens, J. Dielissen and A. Radulescu "Æthereal network on chip: concepts, architectures, and implementations", Design & Test of Computers, IEEE, vol. 5, no. 22, **(2005)**.
[2] N. -S. Woo, "High performance SOC for mobile applications. Solid State Circuits Conference (A-SSCC), 2010 IEEE Asian, **(2010)**; Beijing, China.
[3] S. Kumar, A. Jantsch, J. -P. Soininen, M. Forsell, M. Millberg, J. Oberg, K . and A. Hemani, "A network on chip architecture and design methodology, VLSI, Proceedings, IEEE Computer Society Annual Symposium on **(2002)**; Pittsburgh, PA.
[4] M. Jun, W. W. and E. –Y. Chung, "Exploiting Implementation Diversity and Partial Connection of Routers in Application-Specific Network-on-Chip Topology Synthesis", Computers, IEEE Transactions on vol. 6, no. 63, **(2014)**.
[5] S. Gugulothu, and M. D. Chawhan, "Design and Implementation of Various Topologies for Networks on Chip and Its Performance Evolution", Electronic Systems, Signal Processing and Computing Technologies (ICESC), 2014 International Conference on **(2014)**; Nagpur.
[6] H. J. Mahanta, A. H. Biswas and Md. Anwar, "Networks on Chip: The New Trend of On-Chip Interconnection. Communication Systems and Network Technologies (CSNT)", 2014 Fourth International Conference on **(2014)**; Bhopal, India.
[7] A. Y. Weldezion, M. Grange, D. Pamunuwa, Z. Lu, A. Jantsch, R. Weerasekera and H. Tenhunen, "Scalability of network-on-chip communication architecture for 3-D meshes", Networks-on-Chip, 2009, NoCS 2009, 3rd ACM/IEEE International Symposium on **(2009)**; San Diego, CA.
[8] Y. Ye, J. Xu, B. Huang, X. Wu, W. Zhang, X. Wang, M. Nikdast, Z. Wang, W, Liu and Z. Wang, "3-D Mesh-Based Optical Network-on-Chip for Multiprocessor System-on-Chip", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on vol. 32, no. 4, **(2013)**, pp. 584-596.
[9] C. Marcon, R. Fernandes, R. Cataldo, F. Grando, T. Webber, A. Benso, and L. B. oehls, "Tiny NoC: A 3D Mesh Topology with Router Channel Optimization for Area and Latency Minimization", VLSI Design and 2014 13th International Conference on Embedded Systems, 2014 27th International Conference on **(2014)**; Mumbai.
[10] M. M. Aghatabar, S. Koohi, S. Hessabi and M. Pedram, "An empirical investigation of mesh and torus NoC topologies under different routing algorithms and traffic models", Digital System Design Architectures, Methods and Tools, 2007, DSD 2007, 10th Euromicro Conference on **(2007)**; Lubeck.
[11] J. Jiao, Y. Fu, T. Liu, H. Wang, X. Han and J. Wang, "Performance analysis and optimization for homogenous multi-core system based on 3D Torus Network on Chip", NEWCAS Conference (NEWCAS), 2010 8th IEEE Internationa,l **(2010)**; Montreal, QC.
[12] Y. Ye, J. Xu and X. Wu, "A Torus-Based Hierarchical Optical-Electronic Network-on-Chip for Multiprocessor System-on-Chip", ACM Journal on Emerging Technologies in Computing Systems (JETC), vol. 1, no. 8, **(2012)**.
[13] A. Joshi, P. Venkatesh amd M. Mutyam, "Prevention slot flow-control mechanism for low latency torus network-on-chip", Computers & Digital Techniques, IET, vol. 6, no. 7, **(2013)**.
[14] J. Kim, C. Nicopoulos and D. Park, "A novel dimensionally-decomposed router for on-chip communication in 3D architectures", ACM SIGARCH Computer Architecture News, vol. 2, no. 35. **(2007)**.
[15] NIRGAM：A Simulator for NoC Interconnect Routing and Application Modeling, **(2011)**.

Authors



PengFei Yang. PhD students, was born in 1985. He received the B.Sc., M.Sc.degrees in computer Science and techno logy from Xidian University, Xi'an, China.

His current research interests include embedded system architecture and image processing.

Quan Wang. Was born in 1970. He received the B.Sc., M.Sc., a nd Ph.D. degrees in computer Science and technology from Xidan University, Xi'an, China.

His current interests include input and output technologies and systems, image processing and image understanding.