

**HIGH BANDWIDTH INTERPOSER SWICH (HBI-S) TOPOLOGY IN  
MODULAR SYSTEM ON CHIP**

An Undergraduate Research Scholars Thesis

by

SUNYOUNG PARK

Submitted to the LAUNCH: Undergraduate Research office at  
Texas A&M University  
in partial fulfillment of requirements for the designation as an

UNDERGRADUATE RESEARCH SCHOLAR

Approved by  
Faculty Research Advisor :

Eun Jung Kim, Ph.D.

May 2021

Major:

Computer Engineering – Computer Science Track

Copyright © 2021. Sunyoung Park.

## **RESEARCH COMPLIANCE CERTIFICATION**

Research activities involving the use of human subjects, vertebrate animals, and/or biohazards must be reviewed and approved by the appropriate Texas A&M University regulatory research committee (i.e., IRB, IACUC, IBC) before the activity can commence. This requirement applies to activities conducted at Texas A&M and to activities conducted at non-Texas A&M facilities or institutions. In both cases, students are responsible for working with the relevant Texas A&M research compliance program to ensure and document that all Texas A&M compliance obligations are met before the study begins.

I, Sunyoung Park, certify that all research compliance requirements related to this Undergraduate Research Scholars thesis have been addressed with my Research Faculty Advisor prior to the collection of any data used in this final thesis submission.

This project did not require approval from the Texas A&M University Research Compliance & Biosafety office.

# TABLE OF CONTENTS

	Page
ABSTRACT.....	1
ACKNOWLEDGEMENTS.....	3
NOMENCLATURE.....	4
SECTIONS	
1. INTRODUCTION.....	5
1.1 Concepts of modular System on Chips.....	5
1.2 Deadlock in modular System on Chips.....	6
1.3 Various topologies with higher bisectional bandwidth.....	8
1.4 High Bandwidth Interposer Switch (HBI-S) Topology.....	8
2. METHODS.....	10
2.1 Implementation of Torus and Flattened Butterfly Topology.....	10
2.2 Implementation of HBI-S Topology.....	11
3. RESULTS.....	12
3.1 Simulation result of mesh, torus, and flattened butterfly Topology.....	12
3.2 Simulation result of HBI-S Topology.....	13
4. CONCLUSIONS.....	15
REFERENCES.....	16

# ABSTRACT

High Bandwidth Interposer Switch (HBI-S) Topology in Modular System on Chip

Sunyoung Park  
Department of Computer Science and Engineering  
Texas A&M University

Research Faculty Advisor: Eun Jung Kim  
Department of Computer Science and Engineering  
Texas A&M University

Systems on Chip (SoCs) are getting bigger and complex to achieve performance improvement. As it is costly to develop one gigantic chip, ideas of modular SoCs are proposed. It divides the SoC into small chiplets while interconnecting them with a silicon interposer. Modular SoCs are cheaper to develop while also providing flexible topologies and routing algorithms. They are also advantageous in terms of power and efficiency compared to the traditional SoCs. However, modular SoC has a deadlock issue when interconnecting the interposer with the chiplets. Remote Control (RC) is one of the techniques that can solve the deadlock problem. It places a buffer into the boundary routers with selective injection controls to avoid the deadlock from happening. RC was proven efficient with a mesh topology. However, the interposer network experiences a bottleneck and limits the potential performance benefit of the RC technique. We propose a High Bandwidth Interposer Switch (HBI-S) topology that improves the interposer network's bisection bandwidth compared to the mesh topology. It also significantly reduces the diameter of the interposer network. HBI-S topology achieved better saturation throughput than the mesh topology when simulating various synthetic traffic patterns through Booksim. Specifically, HBI-S topology

showed about 2 times higher throughput in bit-complement, tornado, shuffle, and asymmetric traffic patterns. On average, HBI-S topology provides 1.73 times better throughput than mesh topology.

## **ACKNOWLEDGEMENTS**

### **Contributors**

Thank you for Dr Kim and Ph.D. student, Pritam Majumdar, for helping me in conducting research and writing a thesis paper. The modified Bookism that contains the Remote Control deadlock avoidance technique was provided by Pritam Majumdar and used for the simulation. Additional modifications are done to the Bookism by myself to run a simulation to obtain improved performance.

### **Funding Sources**

This research did not receive any funding.

## NOMENCLATURE

MTR	Modular Turn Restriction
RC	Remote Control
SoC	System on Chip
VC	Virtual Channel

# 1. INTRODUCTION

## 1.1 Concepts of modular System on Chips

Conventional Systems on Chip hold multiple components inside one chip. However, as it becomes larger, SoCs become highly complex, and the manufacture's cost increases. Those shortcomings can be overcome by dividing SoC into multiple chiplets connected through silicon interposers. Chiplets are cheaper to manufacture, with higher yield rates. However, modular SoC has a critical problem of having a deadlock when networks of each chiplets are connected, although the chiplet itself is proven to be deadlock-free [1].

In this research, a chiplet structure with GPUs is used, as shown in Figure 1. It contains four GPU chiplets that are connected in a 4x4 mesh. Routers in a chiplet are represented in yellow circles, and routers located on first row second and third columns and last row second and third columns (colored in orange) are connected with the interposer network through interposer routers, which are represented in a green square.

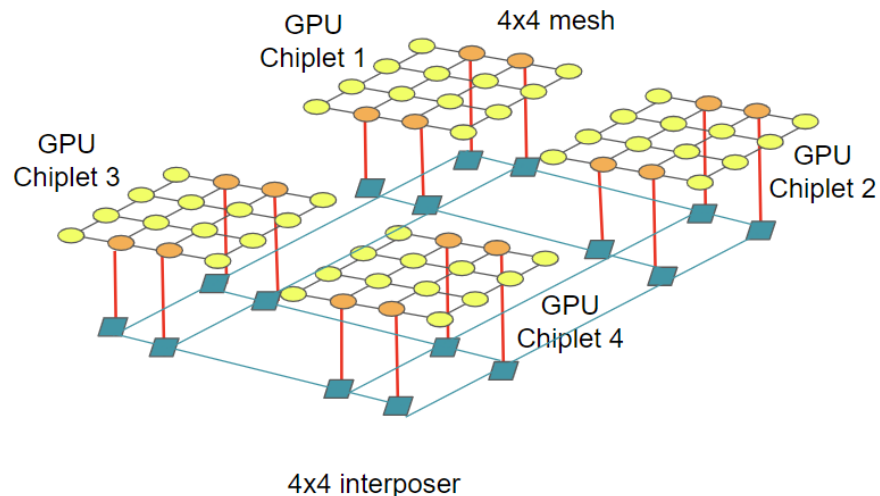


Figure 1: Chiplet structure used in the research



Those routers inside the chiplet that are connected to the interposer network with the interposer routers are called boundary routers from now on. Similar to the chiplet structure, the interposer is also connected with a 4x4 mesh topology. It is possible to attach additional chiplets like CPUs to the interposer routers.

## 1.2 Deadlock in modular System on Chip

One problem of modular System on Chips is that it is possible to have deadlock while connecting different chiplets together, although each chiplet has a deadlock-free routing algorithm. For example, consider a case where packet 1 in chiplet 1 tries to reach router 10 in the neighboring chiplet, and packet 2 in chiplet 2 attempts to reach router 5 on chiplet 1, as shown in figure 2 (a). Assume each packet has 8 flits and each router has one virtual channel. Therefore, each router can store at most one flit at a time. Those packets 1 and 2 are moving in the y-direction to reach their desired destination. However, when the head flit of Packet 1 reaches router 6 on the destination chiplet, it cannot move furthermore to reach its destination at router 10 because the body flit in Packet 2 is still stored in router 10 and blocks the way. It is also not possible for this flit to move forward since at chiplet on the left side, the head of Packet 2 is stuck on router 9 due to Packet 1 stored in router 5, as shown in figure 2 (b). Therefore, this creates a cyclic dependency

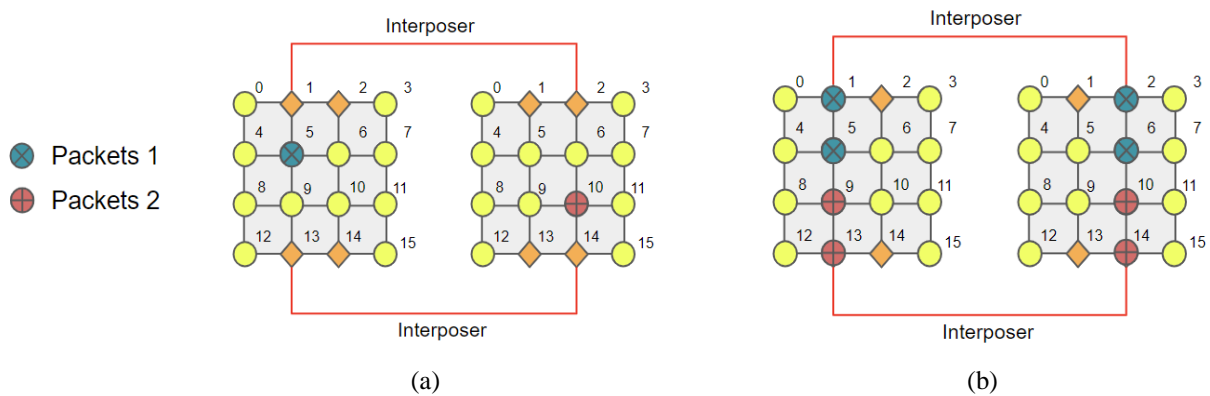


Figure 2. (a) shows the initial location of packet 1 and 2. (b) shows a deadlock situation, where both packets cannot proceed to their destination due to preceding packets.

and result in a deadlock.

### *1.2.1 Modular Turn Restriction*

There are several deadlock avoidance techniques proposed to solve the deadlock among chiplets and interposer networks. Modular Turn Restriction (MTR) is one of the techniques. It places turn restrictions on the boundary routers so that the cyclic dependency shown in the above situation cannot happen, resulting in the deadlock prevented [3]. A packet moving out of the chiplet to reach its destination on another chiplet is called an outbound packet. In contrast to this, a packet that is moving into the chiplet is called an inbound packet. Using the deadlock scenario suggested in the previous section as an example, MTR prohibits outbound packets on boundary router 1,2, and 13 to move into the interposer network. It also prohibits inbound packets from moving into boundary routers 1,2. Therefore, it is obvious that cyclic dependency shown in figure 2 (b) cannot happen from restricting their path to reach their destination. However, it would require more hops for a packet to reach its destination. Although MTR allows deadlock-free multi chiplet structure with an interposer, it has a drawback in terms of efficiency. Due to the restrictions, packets are not moving in an optimal path with minimum numbers of hops. Restrictions also lead to unbalanced load distribution among the boundary routers and lead to slower traffic [4].

### *1.2.2 Remote Control*

Remote Buffer (RC) is another deadlock avoidance technique that places a buffer in boundary routers instead of restricting the turn happening in the boundary router. In this way, it is possible to store outbound packets inside the buffer that could block inbound packets' paths into their destination. After all inbound packets reach their destination, outbound packages are moved out of the buffer and proceed to their destination without creating cyclic dependencies. RC technique showed performance improvement compared to MTR using mesh topology. It also allows complete flexibility in topology and routing. However, RC experiences a slowdown of

traffic inside the interposer network, resulting in a bottleneck of the overall system. Bottleneck happens because a small number of interposer routers should handle a large number of loads coming from all boundary routers from the chiplet network [4].

### **1.3 Various topologies with higher bisectional bandwidth**

It is possible to increase the network's bandwidth by adding more links so that more packets can be transferred one at a time. 2D torus is similar to mesh topology, but it has additional links that connect routers in the edges. Therefore, it could reduce the number of hops between the routers if the source and destination of the packet are placed in the networks' edges.

A flattened butterfly is another high bandwidth topology where each router is connected to all the other routers that are located in the same columns and rows. Therefore, it is a high radix router with 8 links. This results in lower latency and higher bisectional bandwidth. Flattened butterfly topology also significantly reduces the number of hops compared to the mesh, using minimal routing, like XY routing algorithms. It provides 2 hops separation from source to destination routers. Indeed, flattened butterflies are proven to provide 50% higher throughput saturation than concentrated mesh [2].

### **1.4 High Bandwidth Interposer Switch (HBI-S) Topology**

In addition to the topology proposed above, we develop the High Bandwidth Interposer Switch (HBI-S) topology that enhances an interposer network's bisectional bandwidth. HBI-S topology is mixture of ideas on NVSwitch (a switch that allows to connects multiple GPUs for communication through their ports) and BiGraph network topology (proposed to evenly connect switches that are divided into top and bottom directly) [2]. HBI-S topology is a switch like extended interposer topology that provides higher bandwidth and also significantly reduces number hops to reach the destination inside the interposer. Instead of connecting the network in a

4x4 structure like previous mesh, torus, or flattened butterfly topology, the 2x8 structure is used, as shown in figure 3 below. The first top 8 interposer routers, which are indicated in blue and green circles, are connected to the 4 boundary routers of chiplet 1 and 2. The lower part of the interposer routers (represented in orange and gray color) is connected to the boundary routers of chiplet 3 and 4. Then, each router on top is directly connected to all the routers placed on the lower side. Therefore, it is possible to reach the destination within one or two hops at the maximum. For example, moving from boundary routers of chiplet 1 to any other boundary routers in chiplet 2 and 3 will be in 1 hop from direct connections. If it needs to reach chiplet 2, it is possible to move to the router located on the opposite side and move upward to reach its destination in two hops. Therefore, HBI-S provides higher bandwidth with a minimum number of hops and expected to show better performance compared to mesh topology.

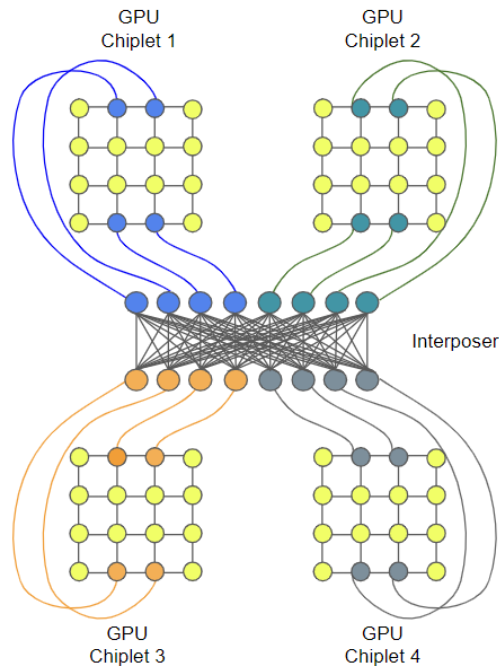


Figure 3. High Bandwidth Interposer Switch Topology

## 2. METHODS

### 2.1 Implementation of torus and flattened butterfly topology

To achieve performance improvement and higher saturated throughput, we implemented 2D torus and flattened butterfly topologies in four 4x4 GPU chiplets, connected with one 2x2 CPU through a 4x4 interposer. Additional links are added to the routers located in the edges to connect edge routers located on the opposite side. Boundary routers have one more link compared to other non-boundary routers due to connection with interposer routers. Those connections are appropriately handled with the correct output port to prevent potential deadlock. For the routing algorithm, we implemented both XY and YX dimensional routing algorithms. For simulation, we used XY routing for both chiplets and interposer networks.

Flattened butterfly topology can be implemented by adding additional links to connect each router with other routers located in the same rows and columns. For the routing algorithm, we implemented the XY routing algorithm. It is considered a minimal routing algorithm and prevents deadlocks as it restricts packets from moving X-direction first and then in Y directions [5].

After implementing 2D torus and flattened butterfly topologies and their routing algorithm, they are simulated through Booksim with RC techniques to evaluate their performances using different synthetic traffic patterns. From obtaining throughput graphs that show the saturation throughput, it is possible to compare their performances, as higher saturation throughput results in better overall performance.

### 2.2 Implementation of HBI-S topology

For simplicity, HBI-S topology is implemented in four 4x4 GPU structures connected by the interposer without 2x2 CPU. The interposer network is created by implementing 2x8 mesh

topology and connecting additional links from top routers to all the routers at the bottom to create a network shown in figure 3 above. For the routing algorithm, it first determines whether the packet is currently located. Then, it determines the column of the destination router and moves to the router on the opposite row with the same column number as the destination router. If the router is a destination router in the interposer, then the packet exits the interposer router by moving into the boundary routers. If not, it will move into the interposer routers in the opposite direction, so that it is possible to reach the destination router. To understand the improvement obtained from the HBI-S topology, it is simulated on Booksim with different synthetic traffic patterns, and compared the saturation throughput with simulation results obtained by using mesh topology.

### 3. RESULTS

#### 3.1 Simulation result of mesh, torus, and flattened butterfly topology

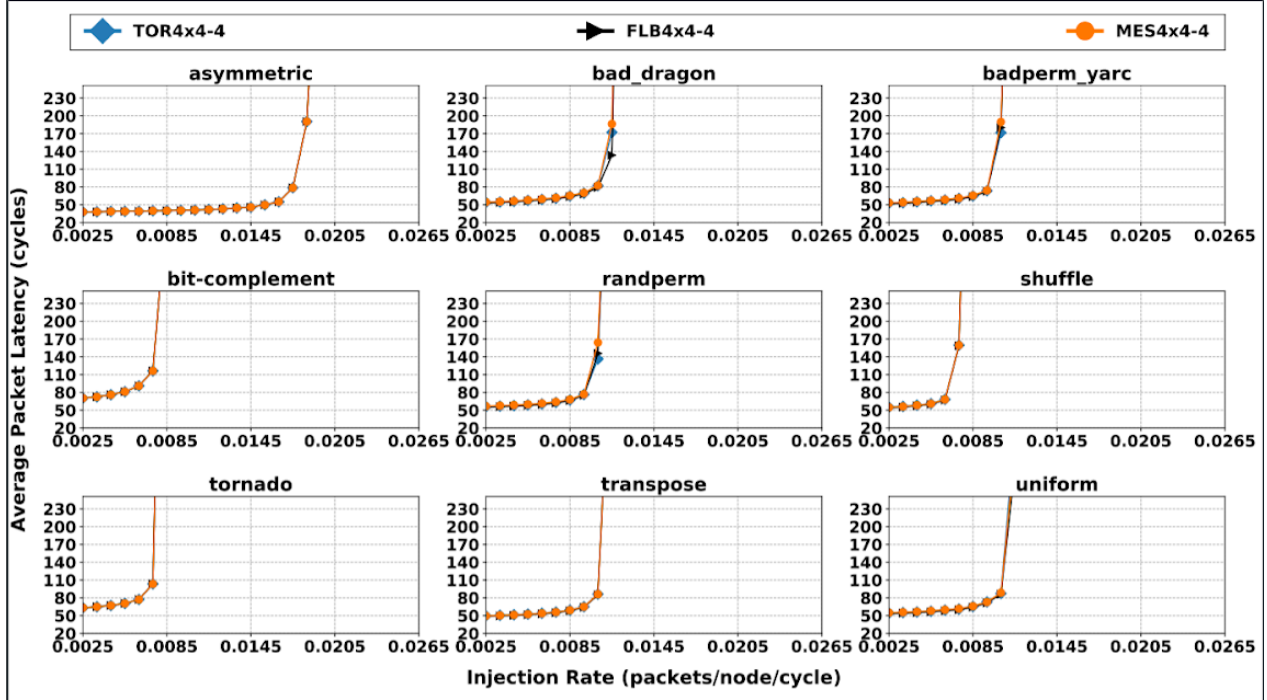


Figure 4: Simulation result of mesh, flattened butterfly, and torus with RC buffer spaces of 4 with 1 virtual channel

Figure 4 shows the throughput obtained from simulating 2D torus, mesh, and flattened butterfly topologies with different synthetic traffic patterns, such as asymmetric, bad\_dragon, badperm\_yarc, bit\_complement, randperm, shuffle, tornado, and uniform. Those topologies are simulated with 4 RC buffers in boundary routers and 1 virtual channel in each router. The saturation throughputs for mesh, 2D torus, and flattened butterfly are similar for most traffic patterns. One notable thing from the graph is that in bad\_dragon traffic, flattened butterfly topology showed the least average packet latency compared to torus and mesh. This suggests that flattened butterfly has the least number of hops while moving inside the chiplets, compared to

torus or mesh. It is reasonable that the mesh has a higher average packet latency since it requires more hops to traverse through the network compared to the torus.

It is expected for the flattened butterfly to show higher saturation throughput compared to other topologies since it has higher bisectional bandwidth compared to mesh and torus. However, the result did not match the expectation. It may happen because the interposer network handles a significantly larger amount of traffic than the chiplet network, so improvement in bandwidth in the chiplet does not benefit the performances. Therefore, it is needed to increase the bisection bandwidth of the interposer network rather than a chiplet network to enhance the overall performance and reduce the potential interposer bottleneck.

### **3.2 Simulation result of HBI-S topology**

HBI-S topology provides higher bisectional bandwidth to the interposer network compared to mesh topology. Figure 5 below shows the saturation throughput result obtained from mesh interposer topology, represented in blue line and HBI-S topology, represented in black. It is obvious from the graphs that for all different traffic patterns, the HBI-S topology handles more packet injections compared to the mesh and has significantly better saturation throughput. Table 1 below summarizes the saturation throughput enhancement achieved by the HBI-S topology compared to the mesh topology. The HBI-S topology achieves 1.58 times higher saturation throughput compared to the mesh topology for uniform traffic patterns. It performed the best with a bit-complement traffic pattern, which shows 2 times improvement in saturation throughput. HBI-S achieves nearly twice the improvement in tornado, shuffle, and asymmetric traffic patterns.



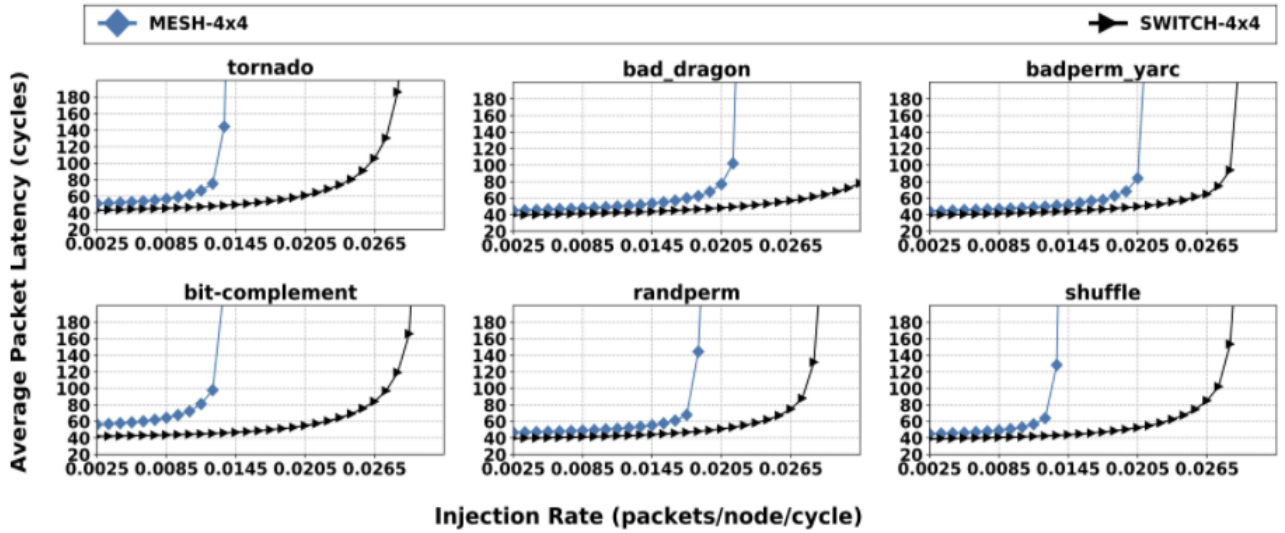


Figure 5. Saturation Throughput graph obtained from simulating HBI-S topology

Table 1. Throughput improvement for each traffic pattern

Traffic pattern	Saturation Throughput of mesh topology	Saturation Throughput of HBI-S topology	Throughput Improvement
Bit-complement	.0135	.0285	2.1
Tornado	.0145	.0285	1.97
Shuffle	.0145	.0280	1.93
Uniform	.0225	.0355	1.58
Bad dragon	.0225	.0355	1.58
Randperm	.0195	.0295	1.51
Bad Perm	.0215	.0285	1.32
Asymmetric	.0315	.0605	1.92

## 4. CONCLUSIONS

RC prevents deadlock efficiently while allowing chip designers to implement routing algorithms and topologies for each chiplet freely. One downside of RC is that the interposer network serves as a bottleneck of the overall system. 2D torus or flattened butterfly topology, which has better bisectional bandwidth than mesh topology, did not improve saturation throughput and performances. HBI-S topology, an interposer topology that provides higher bisectional bandwidths, reduces the bottleneck experience in the interposer network. Compared to the mesh topology, HBI-S achieved an average of 1.73 times higher saturation throughput using various synthetic traffic patterns. HBI-S performed exceptionally with Bit-complement, Tornado, Shuffle, and Asymmetric traffic patterns, where it improved throughput up to 2 times. Therefore, HBI-S topology enhances the benefits of RC by supplementing the shortcoming of it. Furthermore, HBI-S topology can be a simple solution to resolve bottleneck issues that exist in modular SoC.

## REFERENCES

- [1] A. Kannan, N. E. Jerger and G. H. Loh, "Exploiting Interposer Technologies to Disintegrate and Reintegrate Multicore Processors," in *IEEE Micro*, vol. 36, no. 3, pp. 84-93, May-June 2016, doi: 10.1109/MM.2016.53.
- [2] J. Dong et al., "EFLOPS: Algorithm and System Co-Design for a High Performance Distributed Training Platform," 2020 IEEE International Symposium on High Performance Computer Architecture (HPCA), San Diego, CA, USA, 2020, pp. 610-622, doi: 10.1109/HPCA47549.2020.00056.
- [3] J. Yin et al., "Modular Routing Design for Chiplet-Based Systems," 2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA), Los Angeles, CA, 2018, pp. 726-738, doi: 10.1109/ISCA.2018.00066.
- [4] P. Majumder, S. Kim, J. Huang, K. H. Yum and E. J. Kim, "Remote Control: A Simple Deadlock Avoidance Scheme for Modular Systems-on-Chip," in *IEEE Transactions on Computers*, doi: 10.1109/TC.2020.3029682.
- [5] J. Kim, J. Balfour and W. Dally, "Flattened Butterfly Topology for On-Chip Networks," 40th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO 2007), Chicago, IL, 2007, pp. 172-182, doi: 10.1109/MICRO.2007.29