

معماری شبکه روی تراشه ۲,۵ بعدی برای بهبود تاخیر و توان مصرفی

ساجد داداشی

گروه مهندسی کامپیوتر، واحد رودسر و املش، دانشگاه آزاد اسلامی، رودسر، ایران

چکیده

پشته سازی سه بعدی، تراشه های سیلیکونی را بر روی یکدیگر قرار می دهد. پشته سازی ۵.۲ بعدی، چندین تراشه سیلیکونی را در کنار یکدیگر و بر روی یک لایه میان گذر سیلیکونی قرار می دهد و می تواند بسیاری از مشکلات پشته سازی سه بعدی، از قبیل مسائل حرارتی را حل کند. تراشه حافظه، یکی از مهمترین عناصری است که با یک تراشه بیش هسته ای، جمع می شود. چندین راه حل از قبیل همبندی و الگوریتم مسیریابی، برای کاهش تاخیر و توان مصرفی در سیستم های شبکه روی تراشه وجود دارد. در فناوری پشته سازی ۲,۵ بعدی، باید یک همبندی شبکه روی تراشه و همچنین، یک مسیریابی کارآمد وجود داشته باشد تا عملیات ارتباطی به شیوه بهتری انجام شود، بنابراین، تمرکز اصلی این تحقیق، داشتن یک همبندی کارآمد برای لایه بیش هسته ای و لایه میان گذر و همچنین، یک الگوریتم مسیریابی برای ارتباط بین آنها می باشد تا تاخیر و توان مصرفی را کاهش دهد. در این حالت، هر هسته در لایه بیش هسته ای، می تواند به طرز بسیار موثری با بقیه هسته ها ارتباط برقرار کند و ارتباط بین این هسته ها و بقیه تراشه ها هم به نحو مطلوبی برقرار شود.

واژگان کلیدی: پشته سازی ۲,۵ بعدی - لایه میان گذر سیلیکونی - شبکه روی تراشه - همبندی

- مسیریابی

2.5D Network-on-chip Architecture to Enhance Delay and Power Consumption

Sajed Dadashi

Department of Computer Engineering, Roudsar and Amlash Branch,
Islamic Azad University, Roudsar, Iran

Abstract

3D stacking places silicon chips on top of each other. 2.5D stacks multiple silicon dies side-by-side on a silicon interposer layer and can solve many of 3D stacking problems, such as thermal issues. Memory chip is one of the most important elements integrated with a many-core chip. There are various solutions such as topology and routing algorithm to reduce the delay and power consumption in network-on-chip systems. In 2.5D stacking technology, there must be a network topology on the chip as well as an efficient routing algorithm in order to perform communication operations in a better way, so the main focus of this research is on having an efficient topology for many-core layer and the interposer layer and also a routing algorithm for this connection to reduce delay and power consumption. In this case, each core in the many-core layer can effectively communicate with another cores, and the communication between these cores and other chips is established in a favorable way.

Keywords: 2.5D stacking, silicon interposer layer, network-on-chip, topology, routing

Title of the article: Analysis of crista bone resorption around conical implants and its comparison with cylindrical implants.
Volume 5 / Number 1 / Spring 1400 / pp. 63-64

1. Introduction

In many of today's systems, there is a need for powerful chips to perform calculations and communication operations, and by combining these chips, high processing power can be obtained. The set of these cores together with the connections between them form a processing communication network on a chip, which is called network-on-chip. Now, if we want to connect several sets of chips to each other, or especially if we want to connect this many-core set to a memory stack, one way is to connect them through a bus and at a distance from each other. This method has the problem of distance and communication delay. The second method is stacking, which is based on two types of 3D and 2.5D technologies. In the 3D stacking structure, several chips are stacked on top of each other using communication layers. What is the structure of the network in each chip and how the communication network between different levels of the stack should be established, will require a detailed analysis of the structures and consideration of efficiency, speed, temperature, etc. One of the most important problems of 3D stacking is the generated heat. 2.5D stacking enables the integration of multiple chips similar to 3D stacking. The 2.5D system consists of a base silicon interposer layer with several other chips that are stacked on top of this layer. One of the most common chips integrated with the processing chip is the memory element. The simplest case is that only the edge nodes in the multicore layer access the memory stack through the interposer layer. This method does not have the necessary efficiency and other nodes have to access the interposer layer and from there to the memory stack through the edge nodes. In addition to this limited routing, the vast majority of the routing area and routing resources are inefficient. Therefore, another way is to connect all the nodes in the many-core layer to the interposer layer. The network in the interposer layer can be selected depending on the need. In this case, the network-on-chip is extended to the interposer layer, and both the network between the cores and the connecting network of the interposer are considered as network-on-chip. In fact, in this case, the network-on-chip consists of two parts, one part that is on the upper layer and provides communication between cores, and the other part is the interposer (lower layer) that provides communication between cores and the memory stack. Anyway, one of the most important issues in network-on-chip systems, including 2.5D stacking technology, is to create a fast communication platform with very low latency and high efficiency so that the chips can communicate with each other in the shortest possible time and achieve the goal. Two very important things to have these goals are topology and routing algorithm. A suitable topology in the many-core layer which, according to the number of communication links, can acceptably establish communication between the cores with each. Also, a suitable alignment should be considered for the communication structure of the interposer layer, which can establish the communication between the cores of the multi-core layer and the memory stack in a favorable way. A deadlock-free routing algorithm should be included to reduce the communication delay between the cores with each other, as well as between the many-core layer and the memory stack, as much as possible.

2. 2.5D Stacking

There is a need for powerful chips to perform calculations and communication operations, and by combining these chips, high processing power can be obtained. The set of these cores together with the connections between them form a processing communication network on a chip, which is called network-on-chip. If we use a bus to connect multiple chips to each other and also to connect this many-core to a set of memories, distance and communication delay will be a serious

problem. The second method is stacking, which is based on two types of 3D and 2.5D [1]. Fig.1 shows these two types.

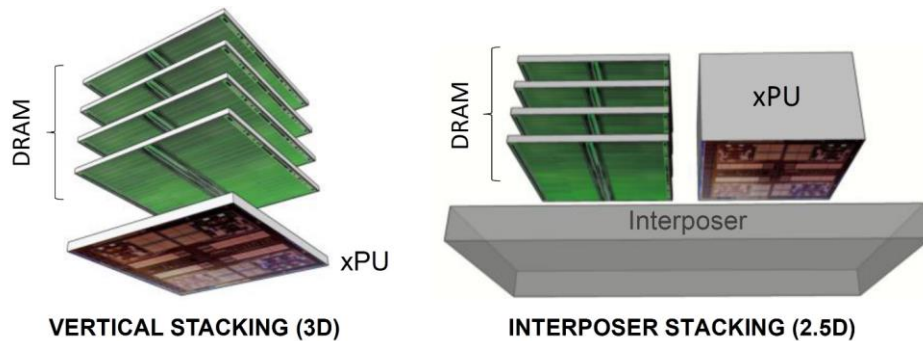


Fig.1. Two types of stacking technologies

In the 3D stacking structure, several chips are stacked on top of each other using communication layers. One of the most important problems of 3D stacking is the problem of generated heat [2]. Horizontal or 2.5D stacking enables the integration of multiple chips similar to 3D stacking. The 2.5D system consists of a base silicon interposer layer with several other chips that are stacked on top of this layer [3]. One of the most common chips integrated with the processing chip is the memory element [4].

The simplest case is that only the edge nodes in the multicore layer access the memory stack through the interposer. This method does not have the necessary efficiency and other nodes have to access the interposer and from there to the memory stack through the edge nodes. In addition to this limited routing, the vast majority of the routing area and routing resources are inefficient. Therefore, another way is to connect all the nodes in the many-core layer to the interposer layer. The network in the interposer layer can be selected depending on the need. In this case, the network-on-chip is extended to the interposer layer, and both the network between the cores and the connecting network of the interposer are considered as network-on-chip [5]. In fact, in this case, the network-on-chip consists of two parts, one part that is on the upper layer and provides communication between cores, and the other part is the interposer that provides communication between cores and the memory stack [6]. Fig.2 shows a 64-core processor chip with four 3D stacks of DRAM.

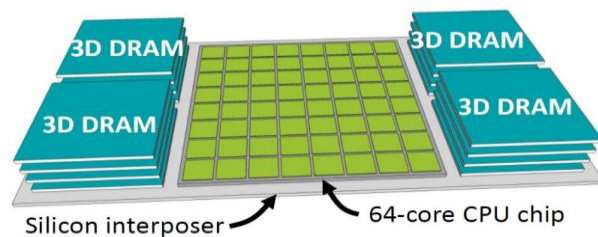


Fig.2. A 64-core processor chip with four 3D stacks of DRAM

To reduce the delay and power consumption, there are various solutions that can be mentioned to improve topology, routing algorithm, arbitration mechanism, etc. Anyway, one of the most important issues in network-on-chip systems, including 2.5D stacking technology, is to create a fast communication platform with very low latency and high efficiency so that the chips can

communicate with each other in the shortest possible time [7]. Two very important things to have these goals are suitable topology and routing algorithm.

By using 2.5D technology and removing interposer packaging steps, the system is made with much less volume and weight, and the length of communication is also reduced. In addition to these, the chips of a system can be manufactured individually, which is best done at the lowest cost. Reusability is another advantage, as a result, systems required for different applications can be realized as different combinations of standard chips [8]. Instead of each chip being designed individually and then placed on the interposer layer (which makes it more complicated to design and route inter-chip connections), considering a silicon interposer layer and multiple chips at the same time based on it, the quality of inter-chip routing can be improved and the design time can be reduced [9].

Among the advantages of the interposer layer, we can mention having a high wiring capacity, significant electrical and thermal efficiency, low cost of active elements due to the partitioning of a large surface and reducing the required power compared to an equivalent chip [10]. Of course, different types of materials can be considered as interposer surfaces, each of which has its own characteristics. One of these materials is silicon and it is chosen because it is a stable base surface and has a very low thermal expansion coefficient. Since the active parts are also made of silicon, thermal-mechanical stresses during manufacturing and processing are minimal and reliability is increased. In addition, silicon offers a very good compromise between thermal conductivity and thickness [11].

The 2.5D inter-silicon path has the ability to focus on the development of the processing node in operational functions with specific applications. Also, the complexity and the number of layers related to the processing node have been reduced, and this will reduce the time required from the start of work to the market. In addition, the resulting wafer has better quality and the cost of working with the wafer is reduced. For this reason, the efficiency, power consumption and the area related to the operation of each application are improved. The silicon interposer layer in 2.5D stacking consists of a silicon chip of the desired size, with metal layers facing up. Different chips can be placed on this layer and combined together. Of course, this layer will be bigger than many-core chips to be able to put different chips on it. The interposer layers can be metallic and passive, and there are no active elements such as transistors on the interposer silicon layer. This layer only performs metal routing between chips and inter-silicon paths for signals entering and exiting the chip. For a passive interposer, the logic required to implement the routers remains on the processor layer, but the wiring is through the interposer [12]. Fig.3 shows the passive interposer layer.

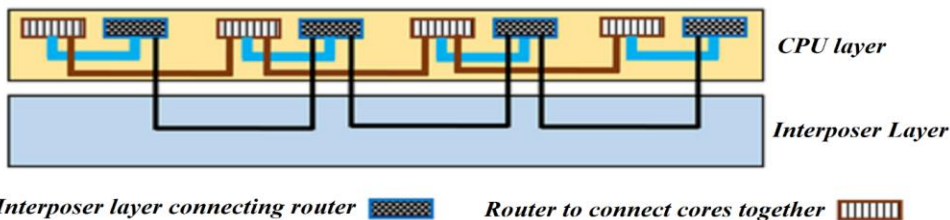
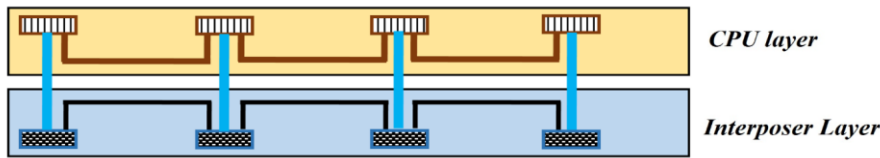


Fig.3. Passive interposer layer

The active interposer consists of transistors on the layer that can act as routers. The active interposer implements both the router logic and the wires for that part of the on-chip network that resides at the interposer layer. Both passive and active interposer network-on-chip are identical

in terms of configuration and operation, but have different physical organization to fulfill the capabilities of their intermediaries. Fig.4 shows the active interposer layer.




Interposer layer connecting router  *Router to connect cores together* 

Fig.4. Active interposer layer

3. Using interposer layer

2.5D stacking typically uses an interposer layer for edge-to-edge communications between adjacent chips. In this case, if we have a multi-core chip and a memory stack, only the processing nodes on the edge of the multi-core layer are connected to the memory stack through the interposer layer, and as a result, the rest of the processing nodes have to use the edge nodes with the interposer layer and communicate with the memory stack from there, and as a result, routing is limited. In addition, the vast majority of interposer level and routing resources are not optimally utilized. In fact, only the interposer layer is used for chip-to-chip routing, and the vertical connections to the underlying surface are for conducting voltage, ground, and I/O [13]. Fig.5 shows this method.

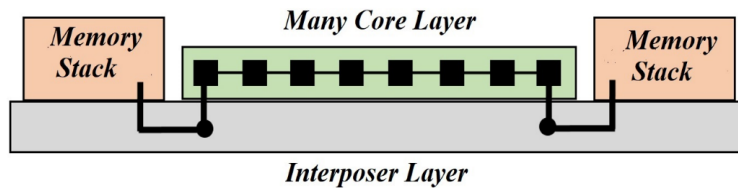


Fig.5. The simplest mode and the least use of the interposer layer

Another method is an interlayer-based network-on-chip architecture that expands both the many-core layer and the interposer layer to make much better use of the interposer space. In this case, the network-on-chip extends to the interposer layer. Core-to-core traffic will be transferred through the network-on-chip in the many-core layer, and memory traffic will be carried out through the network-on-chip in the interposer layer. In fact, the interposer layer will have its own on-chip network, although the type of work depends on whether the interposer layer is passive or active [14]. Fig.6 shows the network-on-chip development method at the interposer layer.

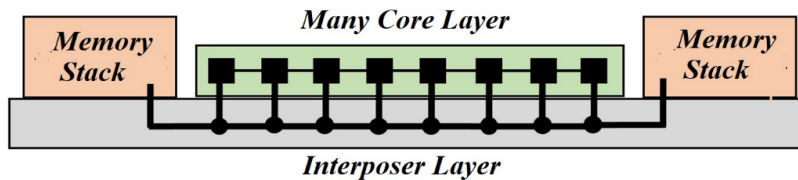


Fig.6. Network-on-chip developed to interposer layer and better efficiency

If the interposer layer is inactive, the router implementation logic remains on the processor layer and the wiring is through the interposer layer, but if the interposer layer is active, it implements both router logic and wires. This implementation is for the part of the network-on-chip that is located on the interposer layer. In fact, both on-chip network links (wires) and routers (transistors) will be placed on the interposer layer. This solution enables the use of interposer metal layers for on-chip network routing. The cost of this work is a part of the processor chip area that creates the network logic elements on the chip. Chips are mounted on the interposer and down by an array of tiny bumps. Focusing can be used to reduce the overhead of the fine bump area. The usual way to create centralization is to have all four nodes in the processor chip layer be centralized to a single node in the network on the interposer layer chip. This reduces the overhead of fine bumps by a factor of four. In this case, part of the overall organization of the system is centralized, which is related to the interposer layer, and the other part is decentralized, which is in the processor layer. This difference is useful when operational separation of integrity traffic and main memory traffic is used. Centralization results in a smaller diameter for the interposer network and reduces the average number of steps for memory range requests to reach their destinations.

4. Separation of different traffic

If the different traffics in the network are distinct from each other, it has the advantage that the traffic is divided among several virtual or physical networks. When designing a multi-core processor with communication networks, it is often helpful to consider several separate logical networks, as it allows prioritization, traffic isolation, and independent flow control [15]. This prevents protocol-level deadlocks. There are different types of messages in cache integrity protocols. As a result, protocol-level cyclic dependencies and deadlocks may occur as messages share network resources. Network-on-chip architectures often use the partitioning of physical resources available at each router input port among multiple virtual channels to perform different routing for different messages. Another method is to use several physical networks, each network is assigned to a class of messages. In both cases, deadlocks are avoided by routing different messages on the dedicated network (virtual or physical) [16]. Core-to-core traffic is from one core in the processor layer to another core in this layer, but in core-to-memory traffic, requests always originate from cores and are always destined for memory nodes, and no other core will be the destination. Therefore, it is clear that core-to-core cache integrity traffic has different characteristics compared to core-to-memory traffic. It is better to use class-based deterministic routing in order to balance different traffic workloads. Many methods for partitioning a network on a chip have been proposed based on request types [17]. Of course, in these methods, there is no interposer layer and different requests related to cores and caches are classified. An interposer-based system can implement a separate physical part of the network-on-chip. As a result, this operation is done without the need for more metal layers in the space. With memory at the edge of the system, the average step count for network traffic on a grid is larger than the average step count for cache traffic. Additionally, cache integrity traffic can often interfere with main memory traffic, and main memory traffic can also be in the path of integrity traffic. If we have a network-on-chip interconnect that is distributed across both the many-core chip and the interposer layer, it is better to operationally partition the network-on-chip so that core-to-core integrity traffic flows through the network-on-chip in the many-core layer and the main memory traffic to be transferred through the network in the interposer layer. In general, we can conclude that by separating core-to-core traffic and core-to-memory traffic, many problems can be avoided. Each core in the processor layer has a link to each of its routers. In fact, the part

of the network-on-chip that is on the processor layer implements a direct network, because each network node can be the source or destination of traffic entering or leaving the cores. The situation regarding the interposer part of the network on the chip is different and it is an indirect network. The reason is that only the endpoints on the left and right edges are connected to the stacked memory channels, and the rest of the middle nodes must pass the packets to the interposer layer or return them to the processor layer. Therefore, it is possible to consider an indirect network for the interposer layer (such as a concentric and indirect network connection).

5. Topology of CPU and Interposer layer

One of the most important issues in network-on-chip systems is to create a fast communication platform, with very low delay and high efficiency. In fact, in the many-core layer, there is a need for an efficient deadlock-free routing algorithm so that the cores can communicate with each other at the right time and the delay and power consumption can be reduced. In the interposer layer, the situation is the same, and a deadlock-free routing algorithm must be included in order to establish the connection between the many-core layer and the interposer layer, and through it, with memory chips. One of the most common connections available is the mesh, which is of interest due to the simplicity of the connection. On the other hand, this connection has fewer communication links compared to many connections, and therefore, the construction cost is lower, but the nodes located in the end areas are far apart, which increases the average number of steps and diameter. The network also increases latency. Fig.7 shows the mesh configuration for the interposer layer.

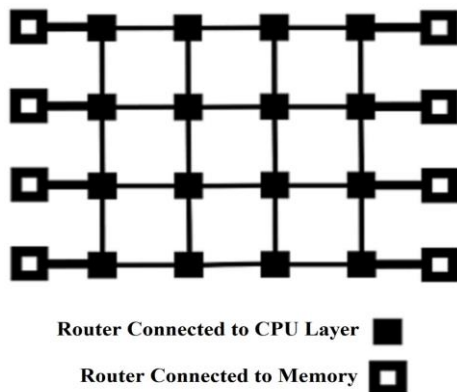
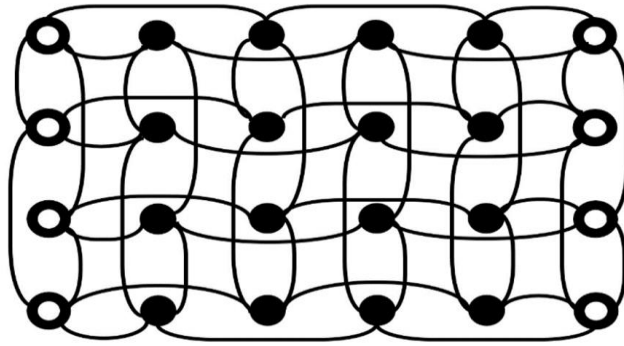


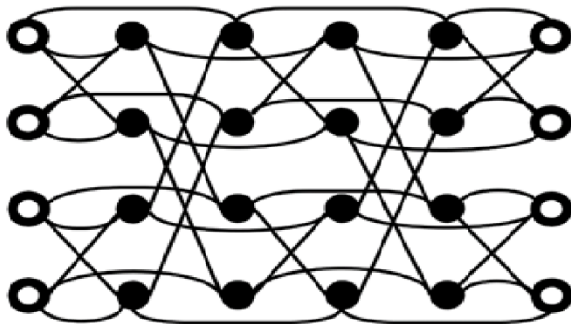
Fig.7. Mesh network

Considering 64 processing nodes in the processor layer and using centralization, there will be 16 routers in the interposer layer, because every four nodes in the processor layer are connected to one router in the interposer layer. These routers establish communication between the processor layer and other chips. 8 routers are included to connect to other chips (memory segments). Regardless of the communication link with other chips, the degree of the router connected to other chips is one. Many of the existing interconnections for the interposer network require increasing the degree of interposer routers in the interposer layer or routers connected to other chips. Many of them cannot be easily used in the processor layer to have a uniform system. Two examples of integrations to improve network-on-chip in the interposer layer are Folded Torus and ButterDonut [6]. Fig.8 shows the Folded Torus topology and Fig.9 shows the ButterDonut topology.



Router Connected to CPU Layer ■
Router Connected to Memory □

Fig.8. Folded Torus Topology



Router Connected to CPU Layer ■
Router Connected to Memory □

Fig.9. ButterDonut Topology

Regardless of the communication link with other chips, the degree of routers connected to other chips is 4 in the Folded Torus configuration and 3 in the ButterDonut configuration. ButterDonut topology is not scalable for networks of larger size and therefore cannot be used for the interposer layer with more nodes and for the processor layer. On the other hand, for a network of this size, it has significant wiring congestion. Folded Torus topology can be used for networks with larger size. The 8x8 Folded Torus configuration for the processor layer is shown in Fig.10.

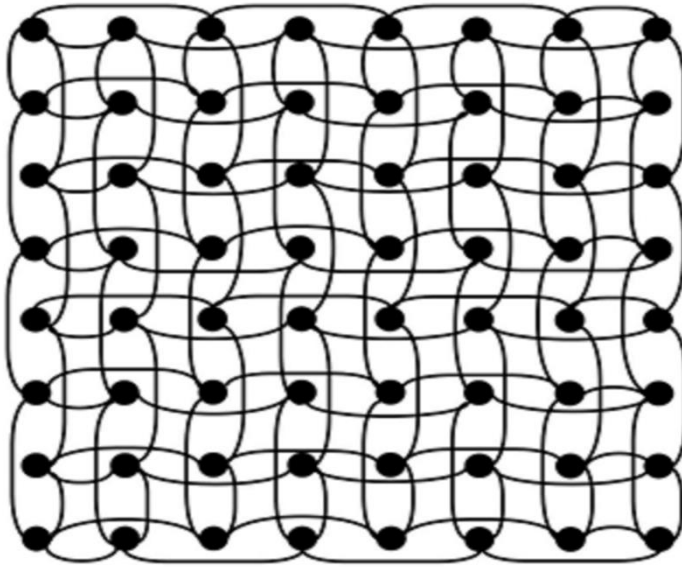


Fig.10. 8×8 Folded Torus Topology

As it turns out, this combination also has high wiring congestion. Especially when the size of the network increases, the number of cross communication channels is very significant. Many network-on-chip architectures use round-robin routing to avoid deadlocks. Next order routing is one of the most common methods in this type of routing. For example, to avoid routing cycles, rotation from Y dimension to X dimension is prohibited. You can use the same method for routing in the layer structure interposer used. Although there are sophisticated routing algorithms that can improve routing, they require a much more complex implementation of next-order routing [9]. For this reason, many network-on-chip architectures use next-order routing. Designers also try to improve the algorithm mapping between tasks and cores, instead of optimizing the routing algorithm for the best use of on-chip network structures [18]. In order to have a deadlock-free routing algorithm, all channels can be divided into several unrelated segments. Unrelated segments include different motion directions or different virtual channel numbers. There is no limit to the selection of channels from each section, but the transition between sections must be in a sequential form and no return is allowed, that is, once transferred to a section, it cannot return to the previous sections [19].

6. Conclusion

The silicon interposer layer in 2.5D stacking consists of a silicon chip of the desired size, with metal layers facing up. Different chips can be placed on this layer and combined together. The interposer layer can be passive or active. There are no active elements such as transistors on the passive interposer silicon layer. This layer only performs metal routing between chips and inter-silicon paths for signals entering and exiting the chip. For a passive interposer, the logic required to implement the routers remains at the processor layer, but the wiring is through the interposer. The active interposer consists of transistors on the layer that can act as routers. The active interposer implements both the router logic and the wires for that part of the network-on-chip that resides at the interposer layer. Both passive and active interposer network-on-chip are identical in terms of configuration and operation, but have different physical organization to fulfill the capabilities of their intermediaries. One of the most important issues in network-on-chip systems is to create a fast communication platform, with very low delay and high efficiency,

so that the cores can communicate with each other in the shortest possible time and achieve the overall goal. In fact, in the many-core layer, there is a need for an efficient deadlock-free routing algorithm so that the cores can communicate with each other at the right time and the delay and power consumption can be reduced.

References:

1. Black, B. (2013). *Die stacking is happening*. Paper presented at the Intl. Symp. on Microarchitecture, Davis, CA.
2. Deng, Y., & Maly, W. P. (2001). *Interconnect characteristics of 2.5-D system integration scheme*. Paper presented at the Proceedings of the 2001 international symposium on Physical design.
3. Abts, D., Enright Jerger, N. D., Kim, J., Gibson, D., & Lipasti, M. H. J. A. S. C. A. N. (2009). Achieving predictable performance through better memory controller placement in many-core CMPs. *37(3)*, 451-461.
4. Saban, K. J. X., White Paper. (2011). Xilinx stacked silicon interconnect technology delivers breakthrough FPGA capacity, bandwidth, and power efficiency. *1*, 1-10.
5. Jerger, N. E., Kannan, A., Li, Z., & Loh, G. H. (2014). *Noc architectures for silicon interposer systems: Why pay for more wires when you can get them (from your interposer) for free?* Paper presented at the Proceedings of the 47th Annual IEEE/ACM International Symposium on Microarchitecture.
6. Kannan, A., Jerger, N. E., & Loh, G. H. (2015). *Enabling interposer-based disintegration of multi-core processors*. Paper presented at the Microarchitecture (MICRO), 2015 48th Annual IEEE/ACM International Symposium on.
7. Balfour, J., & Dally, W. J. (2014). *Design tradeoffs for tiled CMP on-chip networks*. Paper presented at the ACM International Conference on Supercomputing 25th Anniversary Volume.
8. Deng, Y. S., & Maly, W. (2004). *2.5 D system integration: a design driven system implementation schema*. Paper presented at the Proceedings of the 2004 Asia and South Pacific Design Automation Conference.
9. Ho, Y.-K., & Chang, Y.-W. (2013). *Multiple chip planning for chip-interposer codesign*. Paper presented at the Proceedings of the 50th Annual Design Automation Conference.
10. Lenihan, T. G., Matthew, L., & Vardaman, E. J. (2013). *Developments in 2.5 D: The role of silicon interposers*. Paper presented at the Electronics Packaging Technology Conference (EPTC 2013), 2013 IEEE 15th.
11. Bellenger, S., Omnès, L., & Tenailleau, J.-R. J. C., France, IPDiA White Paper Silicon Interposers_260214. (2014). Silicon interposers with integrated passive devices: Ultra-miniaturized solution using 2.5 D packaging platform.
12. Loh, G. H., Jerger, N. E., Kannan, A., & Eckert, Y. (2015). *Interconnect-memory challenges for multi-chip, silicon interposer systems*. Paper presented at the Proceedings of the 2015 International Symposium on Memory Systems.
13. Santarini, M. J. X. J. (2011). Stacked and Loaded: Xilinx SSI, 28-Gbps I/O Yield Amazing FPGAs. *74*, 8-13.
14. Thonnart, Y., & Zid, M. (2014). *Technology assessment of silicon interposers for manycore SoCs: Active, passive, or optical?* Paper presented at the NOCS.
15. Wentzlaff, D., Griffin, P., Hoffmann, H., Bao, L., Edwards, B., Ramey, C., . . . Agarwal, A. J. I. m. (2007). On-chip interconnection architecture of the tile processor. (5), 15-31.

16. Volos, S., Seiculescu, C., Grot, B., Pour, N. K., Falsafi, B., & De Micheli, G. (2012). *CCNoC: Specializing on-chip interconnects for energy efficiency in cache-coherent servers*. Paper presented at the Networks on Chip (NoCS), 2012 Sixth IEEE/ACM International Symposium on.
17. Lotfi-Kamran, P., Grot, B., & Falsafi, B. (2012). *NOC-Out: Microarchitecting a scale-out processor*. Paper presented at the Proceedings of the 2012 45th Annual IEEE/ACM International Symposium on Microarchitecture.
18. Montañana, J. M., Koibuchi, M., Matsutani, H., & Amano, H. (2009). *Balanced Dimension-Order Routing for k-ary n-cubes*. Paper presented at the Parallel Processing Workshops, 2009. ICPPW'09. International Conference on.
19. Ebrahimi, M., & Daneshtalab, M. J. A. S. C. A. N. (2017). EbDa: A new theory on design and verification of deadlock-free interconnection networks. *45(2)*, 703-715.