A 5.8 GBPS UNIFORM MAPPING DATA CENTER SWITCH

Wei-Chih Lai and Ching-Te Chiu

Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

ABSTRACT

With the growing of cloud computing, the need of computing power no longer can be satisfied with a few powerful servers or small scale parallel computer systems. More and more servers are connected together as a data center network. Then, fault tolerance becomes an import issue when building a massive data center network. Currently, many researches focus on building fat-tree data center networks. In this paper, we propose a data center switch with uniform mapping connection patterns to provide higher fault tolerant capability for heavy traffic load fat-tree data center networks. A 4×4 banyan type switch IC is demonstrated as the commodity switch for building the fault tolerant fat-tree data center networks. The 4×4 banyan type switch IC is fabricated in 90 nm CMOS technology, and the maximum operation rate of the IC is 5.8 Gbps with only 23 ps peak-to-peak jitter.

Index Terms— Data Center Network, Scalability, Load Balance, Fault Tolerance, Fat-Tree, Load Balanced Birkhoff-von Neumann Switch.

1. INTRODUCTION

Recently data center networks have become popular since the cloud computing and data center services were announced. There are various cloud services and applications nowadays, such as Amazon Elastic Compute Cloud, Google web 2.0 applications, many on-line anti-virus applications, *etc.* With the growing of cloud computing, the need of computing power no longer can be satisfied with a few powerful servers or small scale parallel computer systems. One possible solution is to bring hundreds of thousands of servers together and connect them into a data center network.

According to Cisco Data Center Infrastructure 2.5 Design Guide [1], the core-aggregation-access three-tier tree architecture is adopted since tree topology is simple and easy to build. However, the bandwidth of interconnections near the core tier is huge so that the tree architecture needs numerous high-end switches and routers. The fat-tree architecture is recommended by M. Al-Fares, *et al.* [2] to solve the interconnection capacity issue, and they also build a 27,648-node fat-tree network by using only commodity switches to support their idea.

The fat-tree topology is one of the most common structure to build data center networks. However, the scalability, load balance, and fault tolerance issues in the fat-tree need to be improved to meet the growth of data center networks. With the advance of semiconductor technologies nowadays, the interconnections and switches in the fat-tree structure can be realized by silicon chips. The interconnections and switches of fat-tree networks become chip interconnection wires and switch IPs [3]. Then, the scalability issue can be mitigated. H. S. Chueh, *et al.* [4] introduce one set of special connection patterns with uniform mapping property to balance traffic loading in the networks. Because traffic in the fat-tree is separated evenly with the uniform mapping patterns, some of the interconnections near the root switches can be reduced. This method can provide a load balanced solution with better scalability for fat-tree data center networks. However, without alternative paths from the source to the destination, the fault tolerant capability in the architecture is greatly decreased.

To increase fault tolerant capability of the fat-tree data center networks under heavy traffic load, we propose special connection patterns in which every connection pattern does not consume all of its bandwidth even under heavy traffic. When there are broken links or faulty switches in the network even in heavy traffic load situations, there are available bandwidth in every connection pattern so alternative paths are available to re-route the traffic. We also propose a new fat-tree architecture for the fat-tree data center networks for supporting the uniform mapping connection patterns since this kinds of connection patterns is designed only for independent and identically distributed (i.i.d.) uniform traffic. We adopt the load balanced Birkhoff-von Neumann switch [5] in our architecture to ensure high throughput in the fat-tree data center networks, and we also provide feedback path [6] at the same time for packet reordering. Finally, we design and implement a scalable switch IC with the proposed load balanced connection patterns that can be used as the commodity switch for building the fault tolerant fat-tree data center networks.

The rest of this work is organized as the follows. In Section II, we present a set of connection patterns with uniform mapping property to achieve better load balance and higher fault tolerance. Then, the load balanced fat-tree architecture is presented. In Section III, measurement results of the 4×4 banyan type switch with uniform mapping connection patterns are given. Finally, a brief conclusion is given in Section IV.

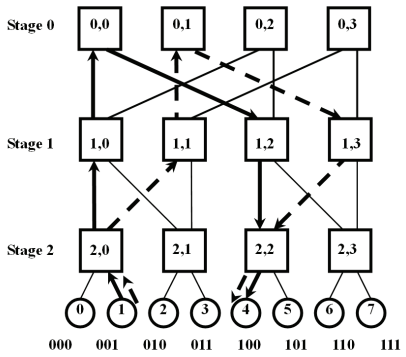


Fig. 1. A 2-ary 3-tree example for sending packets from source node 1 to destination node 4.

2. UNIFORM MAPPING CONNECTION PATTERNS AND LOAD BALANCED FAT-TREE ARCHITECTURE

The fat-tree (k-ary n-tree) is a special type of tree that number of interconnections increases from leaves to root switches. The extra bandwidth provides alternative paths for packets routing from the source to the destination. Taking a 2-ary 3-tree (shown in Fig. 1) as an example, we can find more than one alternative path in the fat-tree to send the packet from source node 1 to destination node 4 (bold lines and dashed lines). However, this fault tolerance example works under an assumption that the traffic of this fat-tree network is light. If we further consider another heavy traffic example that nodes 0 to 3 send packets to nodes 4 to 7 and nodes 4 to 7 send packets to nodes 0 to 3, we observe that all of the interconnections are occupied. In other words, there is no alternative path that can be found in the 2-ary 3-tree when faulty links or switches occur. This fault tolerance issue can be prevented if the connection patterns of the network switches and the fat-tree architecture are well-designed.

2.1. Proposed Uniform Mapping Connection Patterns

In this subsection, a special set of connection patterns with uniform mapping property is introduced to relieve the fault tolerance issue in the fat-tree networks. First, we discuss the special set of connection patterns with uniform mapping property. Then, we compare them with the symmetric time-division multiplexing (STDM) connection patterns, which are applied in the Mailbox switch [6]. Figure 2 (a) shows the N = 4 STDM connection patterns. The equation of each pattern of STDM connection patterns from the source node i to the destination node j on the t^{th} time slot ($t = Nm + l$ and m, l are nature numbers) is shown as below.

$$(i+j) \mod N = (t+1) \mod N \qquad (1)$$

Connection patterns with uniform mapping property separate the traffic uniformly to all of the sub-trees in the fat-tree

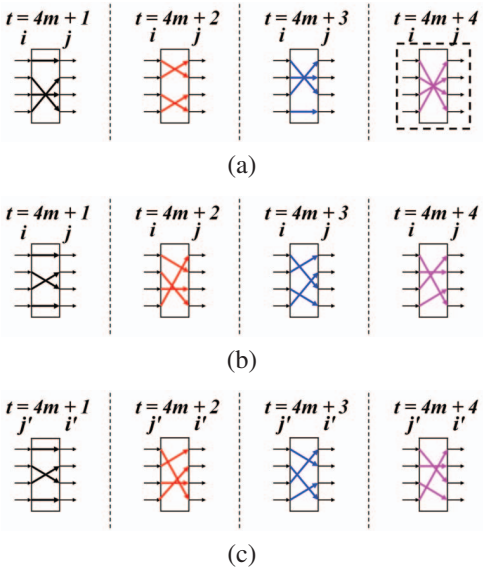


Fig. 2. (a) N = 4 STDM connection patterns. (b) N = 4 bit-reverse connection patterns for 2-ary 2-tree. (c) The mirror patterns of N = 4 bit-reverse connection patterns for load balancing.

architecture. In other words, the bandwidth usage of fat-tree architecture is maintained about the same, and there are many unused links that can be found in the fat-tree architecture. The uniform mapping property is first proposed by H. S. Chueh, *et al.* [4] to reduce the bandwidth utilization in tree based architectures. They adopt the bit-reverse connection patterns in their work that reduces computation complexity of routing paths in fat-tree architecture. The bit-reverse connection pattern is one of the uniform mapping connection patterns, and these pattens can be used to provide load balance and fault tolerance. An example of N = 4 the bit-reverse connection patterns for 2-ary 2-tree is shown in Fig. 2 (b), and this pattern belongs to one-cycle permutations. The equation for one-cycle permutations that connects the source node i to the destination node j on the t^{th} time slot is shown below.

$$j = (i+t) \mod N \qquad (2)$$

As shown in Eq. (2), the source nodes are connected to the destination nodes with the identities increasing by time. The first pattern of the bit-reverse connection patterns is constructed by connecting every source node to the node with binary bit-inverted identity as shown in Fig. 3 (a). The other patterns of the bit-reverse connection patterns follows the Eq. (2) that are shown in Fig. 2 (b).

In Fig. 3 (a) and (b), the link bandwidth usage in 2-ary 2-tree are shown. The maximum supporting link bandwidth is eight (the maximum link bandwidth in k-ary n-tree is nk^n). The link bandwidth is four between stage 0 and stage 1, and the link bandwidth is also four between stage 1 and the nodes.

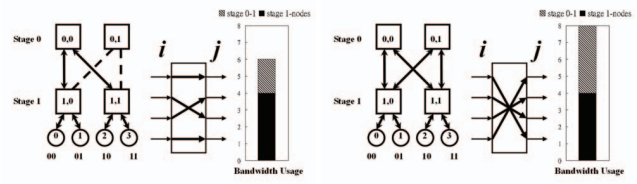


Fig. 3. (a) An example of N = 4 bit-reverse connection patterns applied in 2-ary 2-tree architecture (dashed lines are unused links). (b) A counter example of N = 4 STDM connection patterns applied in 2-ary 2-tree architecture.

The example of bit-reverse connection patterns only requires only six out of eight link bandwidth in total as shown in Fig. 3 (a), and the extra bandwidth can be used for fault tolerance re-routing. On the other hand, the counter example of STDM connection patterns requires all link bandwidth as shown in Fig. 3 (b). The alternative path for packet re-routing when faults occur is impossible to be found.

2.2. The Proposed Load Balanced Fat-Tree Architecture

Fat-tree architectures provide alternative paths from source to destination for routing. With the help of uniform mapping property of bit-reverse connection patterns, the ability of fault tolerance under heavy traffic load can be further improved. However, the bit-reverse connection patterns are designed for i.i.d. uniform traffic because they are periodic and deterministic. High throughput can be achieved only when the traffic load of fat-tree architecture is uniform so that improvement of original fat-tree is required. In order to achieve higher throughput, the proposed load balanced fat-tree architecture combines the k-ary n-tree architecture with an $N \times N$ banyan type switch for load balancing, and the VOQ technique is applied as the interfaces in the proposed architecture.

Figure 4 demonstrates the proposed 2-ary 2-tree architecture, and this architecture can be further scaling up. Taking Fig. 4 as an example, we connect the 2-ary 2-tree with the 4×4 banyan type switch through VOQs (v0, v1, v2, and v3). For the 2-ary 2-tree, these VOQs (v0 to v3) are worked as virtual nodes with uniform traffic load so the network of 2-ary 2-tree can adaptively route packets from these virtual nodes to the destinations according to the N = 4 bit-reverse connection patterns. On the other hand, the 4×4 banyan type switch is applied to balancing the traffic load from nodes (0 to 3) to their virtual nodes (v0 to v3). The connection patterns used in the 4×4 banyan type switch are the mirror patterns of N = 4 bit-reverse connection patterns as shown in Fig 2 (c), which are discussed in next paragraph. The 4×4 banyan type switch (2,0) is without any buffer and consists of four shuffle connected 2×2 crossbar switches (2,0-0,0 to 2,0-1,1). The methods for banyan type switch construction are provided in [6]. In data center networks, it contains hundreds of thousands of server nodes, and it is difficult to connect these modes to the VOQs (virtual nodes). Wiring complexity of an $N \times N$ banyan type switch can be reduced if we implement some of the smaller banyan type switches within a single chip and connect them.

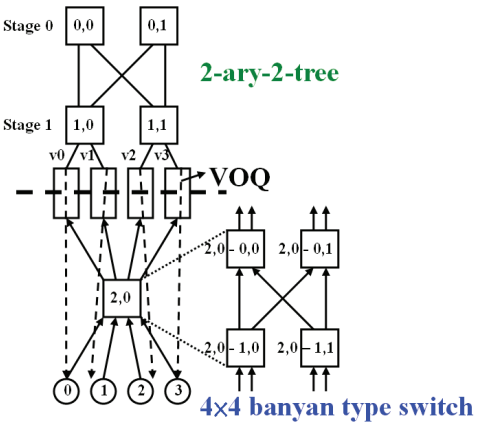


Fig. 4. The proposed load balanced 2-ary 2-tree architecture.

For packet sequence information in the VOQs, we design a new feedback path by combining bit-reverse connection patterns and the mirror connection patterns of them. The mirror connection patterns are for load balancing in the $N \times N$ banyan type switch, and original bit-reverse connection patterns are for sending packets in k-ary n-tree. The first connection pattern of mirror connection patterns is the same as bit-reverse connection patterns. The mirror connection patterns are also one-cycle connection patterns, and they follows the equation shown below.

$$j' = (i' + t) \mod N \tag{3}$$

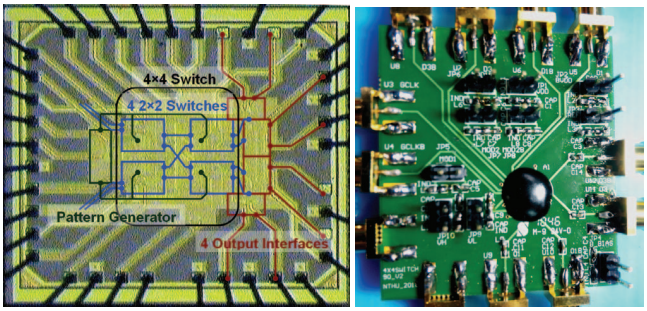
The N = 4 mirror connection patterns for load balancing are shown in Fig. 2 (c). The combination of these two sets of connection patterns provide symmetric property like STDM connection patterns [6]. The VOQ input port index of i' and output port index i are equal so that we have $j = j'$ from Eq. (2) and Eq. (3). The symmetry property of these two set of connection patterns provide feedback path for packet reordering in the proposed load balanced fat-tree architecture.

3. MEASUREMENT OF THE 4×4 SWITCH WITH BIT-REVERSE CONNECTION PATTERNS

We implement the integrated circuit (IC) of 4×4 banyan type switch with the bit-reverse connection patterns and the mirror patterns in 90 nm CMOS technology. The area of this IC is 1.380×1.080 mm^2. Figure 5 (a) shows the IC micro photo, and all function blocks of the 4×4 banyan type switch IC are labelled on the photograph. The function blocks consist

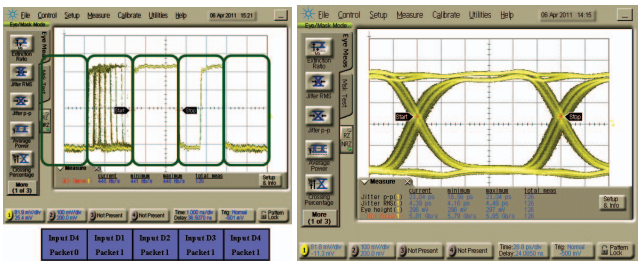
of one pattern generator, four output interfaces, and one 4×4 switch built by four shuffle connected 2×2 switches. Four 2×2 switches are built by current-mode logic multiplexers and D-flip-flops to operate the cross-bar function, and the pattern generator provides select signals for these 2×2 switches to realize the bit-reverse connection patterns and the mirror patterns. Besides, the four output interfaces in the switch IC are for impedance matching of 50 Ω measurement environment load. Figure 5 (b) shows the configuration printed circuit board (PCB) of our switch IC, and this IC is fabricated with Nelco 4000-13.

In Fig. 6 (a), we demonstrate function verification of the first output port in the 4×4 banyan type switch IC. We set the first input port (D1) with 3.6 Gbps pseudo random binary sequence (PRBS) content and the third input port (D3) with 450 MHz clock. For other two input ports, we set the second input port (D2) with logic '1' and the fourth input (D4) with logic '0'. Figure 6 (a) shows the function of the mirror patterns for load balancing in the proposed fat-tree architecture, and the first output port outputs the content from D1, D2, D3, and D4 in sequence as the function of Fig. 2 (c). Furthermore, we test the switch IC with 5.8 Gbps PRBS input content to show the maximum operation speed, and the measurement eye diagram is shown in Fig. 6 (b) with only 23 ps peak-to-peak jitter. The power consumption of this IC is 265 mW.



(a) (b)

Fig. 5. (a) The chip micro photo of the 4×4 banyan type switch IC. (b) The printed circuit board of the 4×4 banyan type switch IC.



(a) (b)

Fig. 6. (a) The function verification of the first output port in the 4×4 banyan type switch IC. (b) The PCB measured eye diagram of first output port at 5.8 Gbps with only 23 ps peak-to-peak jitter.

4. CONCLUSION

In this paper, we adopt the idea of load balanced Birkhoff-von Neumann switch with bit-reverse connection patterns in fat-tree network to solve the source-destination matching problem. With the help of uniform mapping property provided by bit-reverse connection patterns, we can prevent the fat-tree network from improper source-destination matching pairs. Consequently, the fat-tree network can provide higher fault tolerant capability even under heavy traffic load. The load balanced fat-tree architecture is proposed to support bit-reverse connection patterns and provides feedback path for packet reordering. Finally, a 5.8 Gbps 4×4 banyan type switch IC with only 23 ps peak-to-peak jitter is demonstrated as the commodity switch for building the fault tolerant fat-tree networks. To sum up, this paper presents an overall solution for facing the challenges of building massive data center networks.

5. REFERENCES

[1] Cisco Data Center Infrastructure 2.5 Design Guide: [On-line]. Available: http://www.cisco.com/application/pdf/en/us/guest/netsol/ns107/c649/ccmigration_09186a008073377d.pdf, Dec. 2007.

[2] M. Al-Fares, A. Loukissas, and A. Vahdat, "A Scalable, Commodity, Data Center Network Architecture," in *Proceedings of ACM SIGCOMM*, Aug. 2008, pp. 63-74.

[3] N. Farrington, E. Rubow, and A. Vahdat, "Data Center Switch Architecture in the Age of Merchant Silicon," in *Proceedings of 17th IEEE Symposium on High Performance Interconnects*, Aug. 2009, pp. 93-102.

[4] H. S. Chueh, C. M. Lien, C. S. Chang, J. Cheng, and D. S. Lee, "Implementing Load-Balanced Switches with Fat-Tree Networks," technical report NTHU, 2011.

[5] C. S. Chang, D. S. Lee and Y. S. Jou, "Load Balanced Birkhoff-von Neumann Switches, Part I: One-stage Buffering," *Computer Communications*, vol. 25, pp. 611-622, 2002.

[6] C. S. Chang, D. S. Lee, Y. J. Shih, and C. L. Yu, "Mailbox Switch: A Scalable Two-Stage Switch Architecture for Conflict Resolution of Ordered Packets," *IEEE Transactions on Communications*, vol. 56, no. 1, pp. 136-149, Jan. 2008.