ISSN:0975-9646

Simulation and Synthesis of 16×16 Switch for Feed Back Based Switch Systems

P.Aruna [#1], R. Mahesh Kumar [*2], G.Mahammed Rafi[#3]

[1&2]*Electronics and Communication Engineering,*
Annamacharya Institute of Technology and Sciences, Rajampet, A.P
[3]*Lakki Reddy Balireddy College of Engineering*
Mylavaram, Krishna(dt)

***Abstract*— In this paper, A low propagation delay Birkhoff-von Neumann (LB-BvN) 16 × 16 switch fabric IC is proposed for feedback-based switch systems is designed. The next generation terabit switches are constructed from the 16×16 switch. For a feedback-based switch system, throughput significantly reduces due to the long propagation delay of the switch module of system. To overcome propagation delay problem load balanced switches are used. This switch consists of CML D flipflop and CML MUX which are faster compared to CMOS circuits. In CML buffer PMOS active load and active backend termination are added. The switches are directly implemented in high speed domain without SerDes interfaces.The power is reduced with the removing of SerDes interfaces.**

General Terms: D flipflop,MUX,HOL blocking

Keywords—crossbarswitch,VOQ, throughput, delay

I. Introduction

Now a day's computers and commercial devices communicate with each other either by wired or wireless connections. This revolution has led to increasing data traffic in networks. With the help of high-speed internet, cloud computing services are being provided to corporate the individual users. So these applications, which require high bandwidths, have become more and more popular, and this trend is set to continue. Therefore, to support high-bandwidth traffic, the performance of and switches and internet routers should grow drastically. The input buffered switches.

Traditional telephony networks to establish connections use circuit switching techniques. The circuit switching scheme is usually a centralized processor that determines all the connections between input and output ports. Each connection lasts for an average of 180 seconds. In an N × N switch, there are at most N connections simultaneously and the time required to make each connection is 180 seconds divided by N. For suppose , if N is 1000, the time to make a connection is at most 180 ms, which is quite relaxed for most of switch fabrics using current technology, such as CMOS cross point switch chips. For IP routers, the time needed to configure the input–output connections is much more stringent, and it is normally based on a fixed-length time slot[1]. For instance, it could be as small as 64 bytes to cope with the smallest packet length of 40 bytes. For a 10 Gbit/s line, the slot time is about 50 ns. As the line bit rate increases and the number of the switch ports increases, the time needed for each connection is further reduced. As a result, it is impractical to employ centralized connection processors to establish connections between the inputs and the outputs.

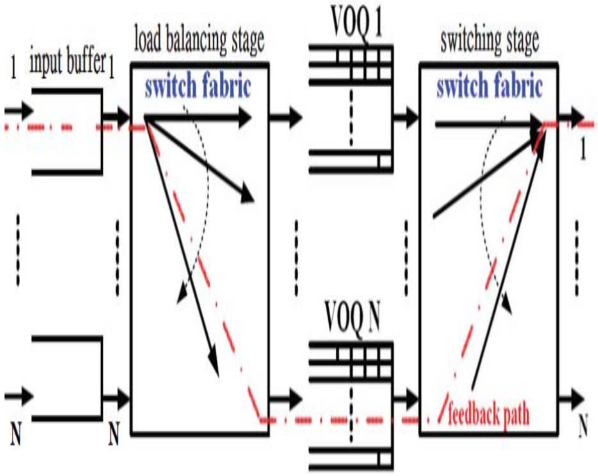


Fig1: Concept of the LB-BvN switch system architecture

The LB-BvN switch consists of two-stage switch fabrics and one-stage parallel buffers (equipped with VOQs) between them. The first stage performs load balancing for the incoming traffic, in order to make traffic arrives at the second stage uniformly. The second stage performs BvN switching on the uniform traffic. Since the connection patterns in both stages are periodic and deterministic, there is no need to find a matching result in every time slot. The design and implementation of a switch fabric IC, a digital signal processing (DSP) switch core with the serializer– deserializer (SerDes) with 8 B/10 B CODEC interfaces is commonly used, as shown in Fig below.. The SerDes interfaces reduce the pin counts of chips. In a feedback based two-stage switch system, a long propagation delay in the feedback path makes the system throughput to decrease significantly. The high-order switch fabric is usually constructed from lower order switches. The effect of propagation delay in the SerDes interface and DSP core becomes worse when a switch fabric scales up.

The motivation for this work is to present an LB-BvN switch architecture for reducing the long propagation delay in a feedback-based LB-BvN switch system . The overall architecture of an LB-BvN 4×4 switch is shown in Fig1. Based on the characteristics of deterministic and periodic connection patterns in an LB-BvN switch, we implement the switch directly in the high-speed domain instead of a DSP core. The current-mode logic D-type flip-flops (CML DFFs) and CML multiplexers (CML MUXes) are adopted to achieve higher operating speed. By operating the switch system directly using high-speed circuits, the SerDes interfaces. which convert high-speed serial stream to low-speed parallel data for DSP core, can be saved in the design .An Internet Protocol (IP) router is a vital network node in today's packet switching network. It consists of multiple input/output ports. Packets from different places arrive at the input ports of the IP router. They are delivered to appropriate output ports by a switch fabric according to a forwarding table, which is updated by routing protocols. In the packet switch network, packets from various input ports may destine for the same output port simultaneously, resulting in output port contention. How to arbitrate and schedule packets when contention arises is an important and challenging issue in designing a high-performance scalable packet switch

II. PROBLEM DESCRIPTION

The RTT of the feedback path includes the packet time, propagation delay from the load balancing stage to the VOQs, delay in the VOQs , and propagation delay from the VOQs to the switching stage . A larger time RTT means that input packets have to wait longer for information from the middle-stage VOQs to keep packets in sequence. With increased scaling , the feedback-based system might degrade the system throughput since the next packet has to wait for the departure information of the previous packets in the VOQs. When the propagation delay in the two-stage switch is longer than the packet time, the system throughput rate decreases as the RTT increases.

In an input-queued switch, with each input maintaining a single first-in first-out (FIFO) queue, may suffer head-of-line(HOL) blocking problem and then result in degradation in throughput down. To solve this problem , the virtual output queuing (VOQ) technique, which maintains a separate queue for each output at each input is used. . Since there are N2 buffers (memories) at the inputs of an N×N switch fabric, the key problem of input-queued switches (equipped with VOQs) is to apply a certain matching algorithm to choose at most N of N 2HOL packets to transmit through the switch fabric. Several algorithms have been proposed to reduce the complexity. The iSLIP has a time complexity of O(log N) to converge with great matching using 2N arbiters . The randomized algorithms computational complexity is O(log N) with the cost of increasing cell delay . The input-queued switch has a longer delay than the load-balanced switch with the traffic arrival rate is above 0.9 . Matching algorithms for conflict resolution require extra computation and communication overheads for every time slot, and these overheads result in another scalability problem. And also matching algorithms cannot guarantee 100% throughput theoretically without a speedup of 2 because the use of a maximal matching algorithm, such as PIM and SLIP [3],will only achieve about 50% throughput Even though heuristic scheduling algorithms require speedup to achieve higher throughput, on-chip speedup could be inexpensive using parallel processing with the advance of semiconductor technologies.

In order to reduce the communication overhead, one approach is to gather the long term statistics of the connection patterns. For an N × N. switch, the computation complexity for the Birkhoff-von Neumann decomposition is O (N4.5) and the number of permutation matrices produced by the decomposition is O(N2). The need for storing the O(N2) number of permutation matrices in the Birkhoff-von Neumann switch makes it difficult to scale for a large N[]. Though there are decomposition methods that reduce the number of permutation matrices, they in general do not have good throughput. For instance, the throughput is O(1/ logN) and it tends to 0 when N is large. Other problem of using long term statistics is that the switch does not adapt well to traffic fluctuation.

III. DESIGN

An LB-BvN N×N switch is constructed with ((N/2)× log2N) 2×2 switches. The STDM connection pattern of each 2 × 2 switch depends on the position in the LB-BvN N × N switch module and the current time slot. The position of a 2 × 2 switch is defined by the column index l and row index m (l,m) as shown in fig below. The column index l of each 2×2 switch is defined from right to left as 1, 2, . . . , log2 N, and the row index m is defined from top to bottom as 1, 2, . . . , N/2[9].

Design and implementation of an LB-BvN N×N switch can be done directly with a specific number N. The strategy is to design a smaller switch to be the basic unit, and then construct an N × N switch with these basic units for flexibility and ease of VLSI design complexity. So a 16×16 switch is constructed by 32 2×2 switches as shown in Fig. 2. In the following section, we will show how to decompose a series of N × N connection patterns into those small 2 × 2 switches. The overall architecture of an LB-BvN 16 ×16 switch is shown in Fig. 2. The pattern generator, the 4 × 4 switch built by four 2 × 2 switches, and the input/output interfaces are the key blocks. The pattern generator provides the deterministic connection patterns for the switches. The 2 × 2 switch realizes the crossbar function which is controlled by the connection pattern.

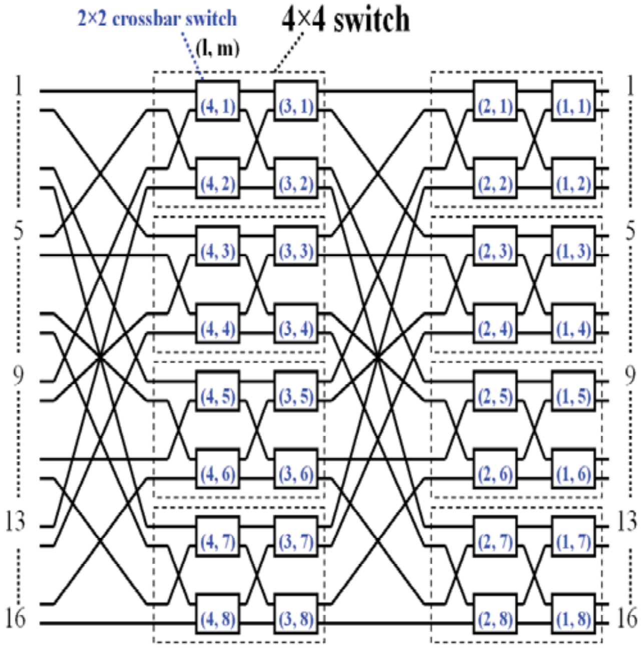


Fig 2:16×16 switch fabric constructed using eight perfect shuffle-connected4 × 4 switch module

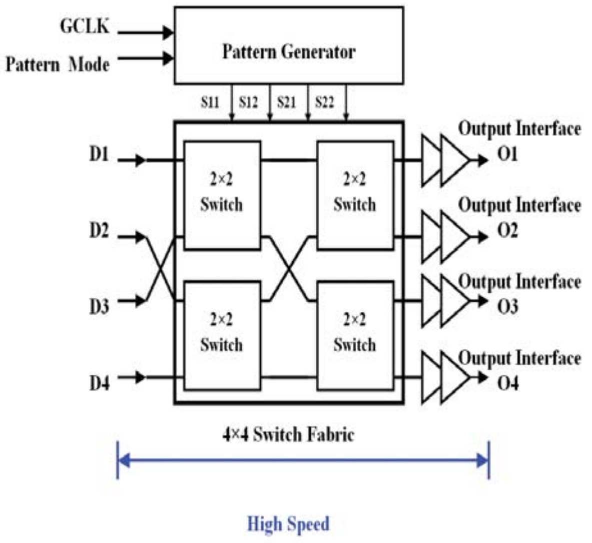


Fig3 :LB- BvN 4x4 switch

For the LB-BvN 4 × 4 switch can be indexed the four 2 × 2 switches with the stage index l and switch index m. For example, the index (l,m) of the upright 2 × 2 switch is (1, 1). The pattern generator can be built by a clock signal (GCLK) and generating different phase signals with a phase shifter. The implementation of 4×4 STDM connection patterns is by using two DFFs as shown in Fig5, for the three MUXes the two-bit reconfigurable selecting signal needs to be set at “00.” The two DFFs can be built by traditional true single-phase clocked DFFs or CML DFFs, depending on the system switching speed. The 4 × 4 staggered symmetric connection patterns also can be generated by the pattern generator circuit as shown in Fig4 below.

The STDM patterns and staggered symmetric patterns are signal inversions. Consider the load-balancing stage for example, S11, S12, and S22 of the staggered symmetric patterns are equal to the STDM ones[1]. On the other hand, the S21 of the staggered symmetric patterns is obtained from inverting the S21 of the STDM pattern. From fig4, the staggered symmetric patterns can be got by setting the pattern mode to “10” for the load-balancing stage and “11” for the switching stage.

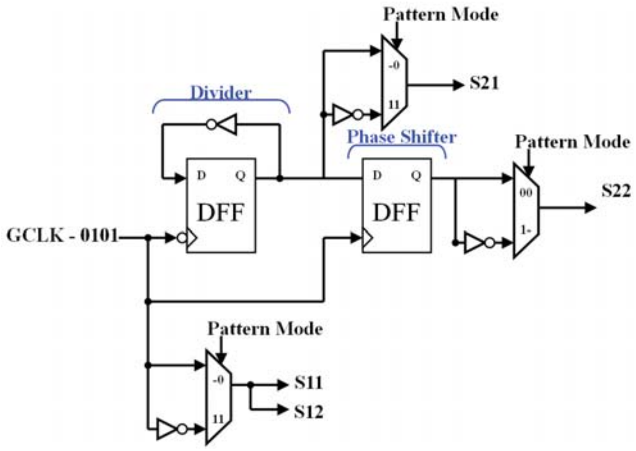


Fig4: Block diagram of pattern generator

In order to construct higher order switches, such as a 64×64 or 128× 128 switch, the same method can be applied[10]. By comparing used pattern generator only to previous work since this pattern generator is used to generate switch connection pattern and not the conventional pseudo-random binary sequence (PRBS) generator. The switch pattern generator is the key device to control the connection between the input and output ports. This is the first and the only ones to realize the load-balanced BvN switch pattern generator in hardware.

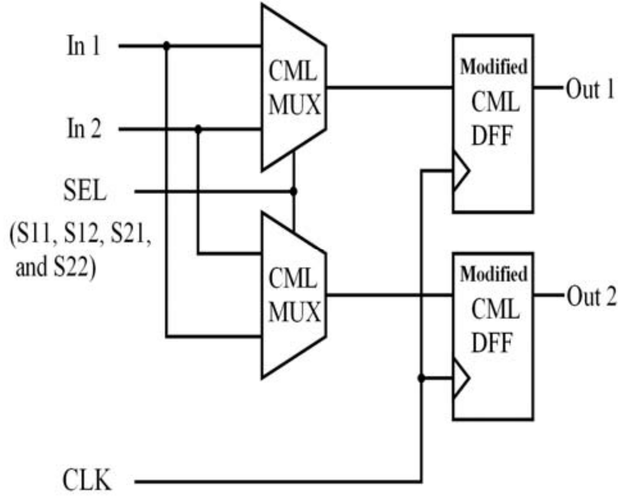


Fig5:Block diagram of 2x2 switch

Based on the selection line value the cross connection or bar connection are decided. If the selection line is “1” then crossbar function performed, else bar connection is done as shown in the following examples of 4 x4 switch function.

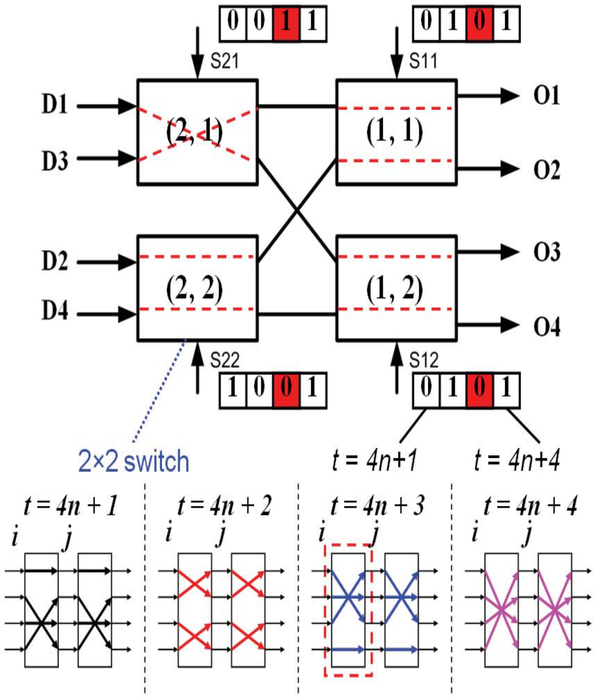
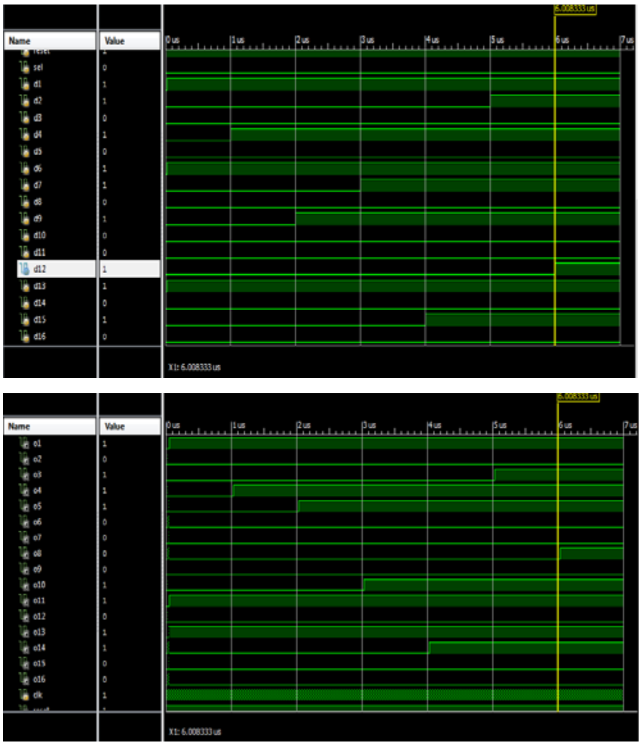


Fig6 :4×4 switch connection patterns

IV. Results and Performance Analysis

The 16×16 switch results are shown below. From results it is shown that maximum throughput is possible.



Name	Value
sel	0
d1	1
d2	1
d3	0
d4	1
d5	0
d6	1
d7	1
d8	0
d9	1
d10	0
d11	0
d12	1
d13	1
d14	0
d15	1
d16	0

Name	Value
o1	1
o2	0
o3	1
o4	1
o5	1
o6	0
o7	0
o8	0
o9	0
o10	1
o11	1
o12	0
o13	1
o14	1
o15	0
o16	0
clk	1

Fig7: result of 16×16 switch

Conclusion & Future Scope

Most of commercial devices communicate with computers with help of wired or wireless connections. Traffic will be increased with number of devices used in communication. To overcome this problem high speed switches are used. The 16×16 will be one of mostly used switch with maximum throughput. With the help of 16×16 we can also implement the higher order switches like 32×32 ,64×64 and also higher order like 1024 ×1024 with help of 16×16 as a basic component.

References

[1] Ching-te chiu,Yu-hao Hsu,Wei-CHih Lai,Jen wing Wu,ShawnS.H.HsuYang-Syu Lin, Fan-Ta Chen, Min-Sheng Kao, and Yar-Sun Hsu “Low propagation delay load balanced 4×4 switch fabric ic in 0.13µm CMOS technology”.

[2] N. Chrysos and G. Dimitrakopoulos, “Practical high-throughput crossbar scheduling,” IEEE Micro, vol. 29, no. 4, pp. 22–35, Jul.-Aug. 2009.

[3] A. Mekkittikul and N. McKeown, “A practical scheduling algorithm to achieve 100% throughput in input-queued switches,” in Proc. IEEE INFOCOM 17th Ann. Joint Conf. Comput. Commun. Soc., Mar.–Apr. 1998, pp. 792–799

[4] Y. Tamir and H. C. Chi, “Symmetric crossbar arbiters for VLSI communication switches,” IEEE Trans. Parallel Dist. Syst., vol. 4, no. 1, pp.13–27, Aug. 1993

[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] A. Mekkittikul and N. McKeown, “A practical scheduling algorithm to achieve 100% throughput in input-queued switches,” Proceedings of IEEE INFOCOM, 1998.

[7] J. Dai and B. Prabhakar, “The throughput of data switches with and without speedup,” in Proc. IEEE INFOCOM 19th Ann. Joint Conf. Comput. Commun. Soc., 2000, pp. 556–564

[8] Y. Shen, S. S. Panwar, and H. J. Chao, “Providing 100% throughput in a buffered crossbar switch,” in Proc. IEEE HPSR, New York, Jun. 2007, pp. 1–9.

[9] C. S. Chang, D. S. Lee, and Y. J. Shih, “Mailbox switch: A scalable two-stage switch architecture for conflict resolution of ordered packets,” *IEEE Trans. Commun.*, vol. 56, no. 1, pp. 136–149, Jan. 2008.

[10] Y. H. Hsu, M. H. Lu, P. L. Yang, F. T. Chen, Y. H. Li, M. S. Kao, C. H. Lin, C. T. Chiu, J. M. Wu, S. H. Hsu, and Y. S. Hsu, “A 28Gb/s 4 × 4 switch with low jitter SerDes using area-saving RF model in 0.13µm CMOS technology,” in Proc. IEEE Int. Symp. Circuits Syst., May 2008,pp. 3086–3089.