Microprocessors and Microsystems 000 (2016) 1–9

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

Circuit design of Clos-based on-chip interconnection networks☆

Yikun Jiang, Mei Yang*

Department of Electrical and Computer Engineering, University of Nevada, Las Vegas, NV 89154, United States

ARTICLE INFO

Article history:
Received 16 October 2015
Revised 31 January 2016
Accepted 3 March 2016
Available online xxx

Keywords:
Crossbars
Benes
Clos
Network-on-Chip (NoC)
Schematic
Layout

ABSTRACT

Single-hop non-blocking networks have the advantage of providing uniform latency and throughput, which is important for cache-coherent network-on-chip systems. This paper focuses on high performance circuit designs of multi-stage non-blocking networks as alternatives to crossbars. Existing work shows that Benes networks have much lower transistor count and smaller circuit area but longer delay than crossbars. To reduce the timing delay, we propose to design the Clos network built with larger size switches. Using less than half number of stages than the Benes network, the Clos network with 4×4 switches can significantly reduce the timing delay. The circuit designs of both Benes and Clos networks in different sizes are conducted considering two types of implementation of the configurable switch: with N-type metal-oxide-semiconductor logic (NMOS) transistors only and full transmission gates (TGs). The layout and simulation results under 45 nm technology show that the TG-based implementation demonstrates much better signal integrity than its counterpart. Clos networks achieve average 60% lower timing delay than Benes networks with even smaller area and power consumption.

1. Introduction

Networks-on-Chip (NoCs) have emerged as the key on-chip communication architecture for multiprocessor systems-on-chip and chip multiprocessors [1,27,29]. Achieving scaling performance for future many-core systems will require high-performance, yet energy-efficient on-chip interconnection networks [2,27,28]. Existing NoC topologies can be classified into two categories: (1) Multi-hop interconnection networks, like mesh [4], torus, concentrated mesh [3], etc., (2) Single-hop non-blocking indirect networks, like crossbar [25], Benes [26], Clos [19], etc.

NoC systems, such as the Tilera Tile64 [5], utilize a distributed mesh-based network to avoid the scaling issues of long wires. However, this improved scalability comes at the expense of non-uniform cache access (NUCA) latencies [15] and high variability in memory access latencies [6], as well as increased design complexity to guarantee correctness and fairness [23]. The study in [14] shows that mesh network's accepted throughput at any given node is highly dependent on the location of the destination node, as shown in Fig. 1. A number of solutions have been developed to solve the problem but at the cost of more complexity routing algorithms and adoption of additional buffers at each router. These buffers consume significant power and area. According to Intel's projections, the interconnection network itself consumes more than 30% of total chip power [16,30].

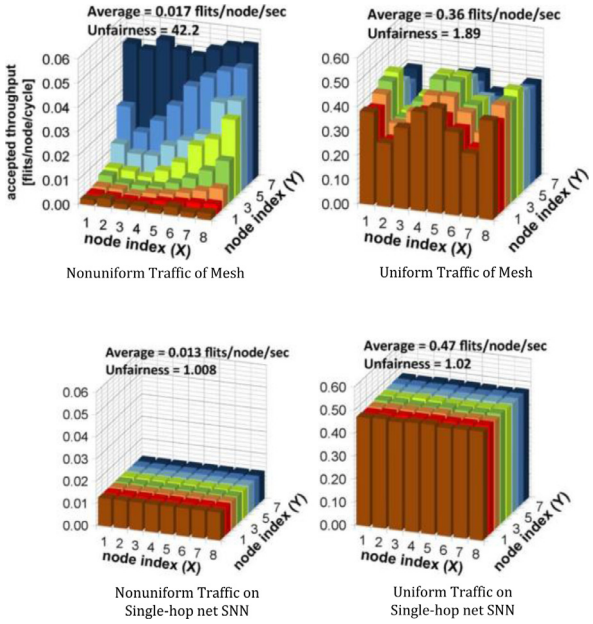

Fig. 1. Traffic analysis of mesh- and crossbar-based NoC [14].

Contrastingly, single-hop non-blocking networks eliminate the need of intermediate buffers thus can provide uniform latency and throughput, which are very important for cache-coherent

☆ This work was supported in part by NSF under grant no. CNS-1126688.
* Corresponding author: Tel.: +1 (702) 895 2364.
E-mail address: jiangy3@unlv.nevada.edu, mei.yang@unlv.edu (M. Yang).

http://dx.doi.org/10.1016/j.micpro.2016.03.008
0141-9331/

many-core systems [14,24]. Additionally, due to their high bisection bandwidth, non-blocking networks can potentially provide lower complexity solutions with quality-of-service guarantees than multi-hop networks [16,25]. The crossbar-based swizzle-switch network (SSN) achieves significant performance improvement in throughput (21%), cache miss latency (3.0x), and energy savings (25%) than the mesh-based network [14].

Our study is focused on high performance circuit designs of on-chip non-blocking networks, including single-stage networks (i.e., crossbar) and multi-stage networks (Benes and Clos networks) [21]. The scalability of crossbar designs is limited by the quadratically increased circuit complexity. As an alternative to crossbars, Benes networks have much lower transistor count and smaller circuit area but longer delay than crossbars [17]. In [20], 3D folded designs of Benes networks and Clos networks built with 2×2 switches are presented. But there is no exploration of Clos networks with larger size switches. In addition, the aforementioned designs are conducted under 130 nm or older technology. There is a need to evaluate these network designs under current newer technology.

To reduce the timing delay, we propose to design the Clos network built with larger size switches. Using less than half number of stages than the Benes network, the Clos network with 4×4 switches can significantly reduce the timing delay. The circuit designs of both Benes and Clos networks in different sizes are conducted considering two types of implementation of the configurable switch: with NMOS transistors only and full transmission gates (TGs). The layout and simulation results under 45 nm technology show that Clos networks achieve average significant lower timing delay than Benes networks with even smaller area and power consumption.

The rest of the paper is organized as follows. Section 2 reviews existing work on circuit designs of single-hop non-blocking networks. Section 3 describes the properties of non-blocking networks. Section 4 presents the circuit design methodology. Section 5 presents the simulation results. Section 6 concludes the paper.

2. Related work

2.1. Crossbar designs

To provide better bandwidth, crossbars are used to replace bus-based interconnect fabrics in early multi-core systems, like in the Niagrara2 [7] and IBM BlueGene [13]. Crossbar-based architectures not only can provide the uniform memory access latency that is unachievable in multi-stage NoC systems, but also can potentially provide higher bisection bandwidth and lower complexity solutions for quality-of-service guarantees than NoC designs. Despite these advantages, large crossbars are generally considered infeasible because the area and power of traditional matrix-style crossbars grow quadratically with crossbar radix [8]. Therefore, it is commonly believed that they become overly expensive for radixes above 32 or 64 [9]. In some work, the crosspoint queueing (CQ) was adopted to replace the traditional input queueing (IQ) to simplify complexity and improve the performance of crossbar scheduling [10]. However, crosspoint queueing was found expensive due to the high partitioning of the switch memory; since there is one memory per crosspoint, the total number of memories grows as $O(N^2)$, which is costly for flow and congestion control algorithms [11,12]. Furthermore, the work in [10] proposed the hierarchically-queued crossbar (HQ) as an organization that lowers memory partitioning. In this organization, an $N \times N$ crossbar is partitioned in $(N/k)^2$ $k \times k$ sub-crossbars and memories are placed only at the inputs and outputs of the sub-crossbars. Hence, the total number of memories is reduced from $O(N^2)$ to $O(N^2/k)$. Unfortunately, this organization has a major disadvantage: although partitioning is lowered, it remains unacceptably high, especially when N is large. The reason is that each sub-crossbar has to be relatively small in order to be efficiently scheduled, which, in turn, implies a small k and a quick growth rate of total number of memories.

Existing crossbar circuits mainly adopt MUX-based designs and matrix-based designs [17]. For MUX-based crossbar designs, the latency experienced by a signal depends on crossbar size. For data paths constructed using 2-to-1 multiplexers, doubling the number of inputs results in an additional multiplexer on each line [17]. Pipelined crossbars are proposed to speed up MUX-based designs. Both IBM C64 [18] crossbar and 128×128 crossbar [22] are pipelined MUX-based designs. The IBM Cyclops64 is a 96×96 96-bit-wide crossbar implemented in a 90 nm technology, running at 533 MHz, and occupying 27 mm^2, including the circuits for queueing, arbitration, and flow control [18]. In [22], a 128×128 32-bit-wide crossbar switch is implemented in 90 nm CMOS standard-cell ASIC technology. The crossbar operates at 750 MHz and provides a port capacity above 20 Gb/s, while fitting in a silicon area as small as 6.6 mm^2 by filling it at 90% level (control not included). Though the throughput is dramatically improved with pipelined crossbar designs, the port-to-port latency is kept undesirable.

The complex wire interleaving in traditional MUX-based crossbars causes the layout challenge at high bus widths [14]. Most recent crossbars use matrix-style structures. The results in [17] show that for the same size crossbar, compared with the MUX-based design, the matrix-based design reduces the transistor count by 90% and latency up to 50%. Conventional matrix-based crossbars consist of the switching fabric and a separate arbiter that configures the crossbar. This decoupled approach imposes the routing challenge and complexity in arbiter design when the radix of the crossbar increases. Passa's work proposed a novel microarchitecture that inverts the locality of wires by orthogonally interleaving the input with the output arbiters, thus reducing the routing area from $O(N^4)$ to $O(N^2\log^2N)$. However, the prohibitive overhead of the arbiter (consuming 60% of total crossbar area) still limits the design to scale when implementing high-radix crossbar.

2.2. Benes/Clos designs

Though matrix-based crossbars overbeat MUX-based designs in both area and timing delay [17], their scalability is still limited by the quadratically increased circuit complexity. As an alternative to crossbars, Benes networks have much lower transistor count and smaller circuit area but longer delay than crossbars [17]. The circuit design of 2D Benes network shows that the Benes network significantly reduces the transistor count and power consumption compared with the same size matrix-based crossbar design. While the latency result of the Benes network is worse than that of the matrix-based crossbar counterpart [17]. In [20], 3D folded designs of Benes networks and Clos networks built with 2×2 switches are presented. The numerical analysis shows that 3D folded design can help improving the latency result. But this work only provides the 3D folded design in theoretical aspect, there is no actual layout of the circuit. As a matter of fact, the Through Silicon Via (TSVs) cannot satisfy the density requirement of high-radix crossbar connection because each TSV needs a large pad area to guarantee the quality. Besides, in this work there is no exploration of Clos networks with larger size switches.

Clos networks have been adopted to interconnect multi/many cores [16,30] in 2D and 3D structures. However, to the best of our knowledge, there is no work on using the Clos network as a replacement of crossbars.

In the literature, the studies on circuit design of crossbar and Benes/Clos networks are limited. Due to the lack of appropriate wire models, it's very inaccurate to conduct circuit level simulation for designs, which completely neglects any effect that

Please cite this article as: Y. Jiang, M. Yang, Circuit design of Clos-based on-chip interconnection networks1, Microprocessors and Microsystems (2016), http://dx.doi.org/10.1016/j.micpro.2016.03.008