An Efficient Network-on-Chip Architecture Based on the Fat-Tree (FT) Topology

A. Bouhraoua   **M. E. Elrabaa**

Computer Engineering Department
King Fahd University of Petroleum and Minerals
P.O.Box 969, Dhahran, 31261
Saudi Arabia
Email: abouh,elrabaa@ccse.kfupm.edu.sa

Abstract—A novel approach for an efficient network-on-chip using a modified Fat Tree is presented. Contention is eliminated and latency is reduced through an improved topology and router architecture. The adopted topology increases performance without a substantial increase in the routing cost. This is achieved by using an improved buffer-less, paremeterizable router architecture. The proposed router architecture is simple to implement yet can achieve the required packet collision avoidance. Simulation results that show the level of performance achieved by both the topology and the router architecture are presented. A throughput of more than 90% is achieved way above the 40-50% usually seen in other networks on chips.

Index Terms—Networks-On-Chip, Systems-on-Chip, ASICs, Interconnection Networks

I. INTRODUCTION

THE ensuing growth of Systems-on-Chips (SoCs) complexity and their very short time to market constraints had led to new design methodologies. SoCs are mainly built by integrating many IP (Intellectual Property) blocks from different vendors to form the desired system. Connecting these different IP blocks together poses many issues at several levels of the design and implementation stages. Communication protocol compatibility, bandwidth requirements and performance are but some of the design issues. Clock distribution and the overall timing closure of the whole chip are implementation problems.

Networks-on-Chips (NoCs) paradigm has emerged as an alternative to ad-hoc wiring or bus-based global interconnect in SoCs [1-4]. It provides a systematic solution for issues of compatibility, bandwidth requirements, performance and eases the handling of the clock distribution task and makes the timing closure more manageable. Hence, the general consensus is that the communication requirements, as well as the design flow, of billion transistors SoC are best accommodated by shared, segmented interconnection networks [1,2].

The intensive study of interconnection networks in the 80s/90s for the purpose of connecting parallel processors produced many solutions that are adapted now for the NoC area. Since [2], the majority of the proposed NoCs have been directly derived from existing topologies and routing algorithms [5-7]. Some other NoCs focus only on the router architecture alone neglecting the other components of the network like the topology, the flow control and the routing scheme [8,9].

As a result of the above strategies, the complexity of routers did not change significantly compared to inter-chip interconnection networks, where routers were designed to fit on a single chip. However, NoC routers should be designed so that several instances can be easily integrated on-chip with a relatively negligible overhead.

Other contributions have introduced new routing schemes in order to reduce the router gate count[10,15]. The Nostrum network, built using a 2-D mesh with deflection routing, has taken the direction of reducing the size of the router. Deflection, routing enables a buffer-less router design thus maintaining a relatively low-cost. However, because of deflection routing and the mesh structure, latency and throughput are both sacrificed [10]. The throughput levels reached by the Nostrum router are just a small fraction of the maximum available bandwidth [11] and saturate very quickly [10]. Deflection routing imposes the adoption of a store-and-forward routing mechanism instead of wormhole routing because of deadlocks [12]. The store-and-forward mechanism introduces more delay. Packet size is very small (96 bits) to keep the inter-node wiring acceptable (128 bits) and the size of the input buffers small enough to claim the buffer-less status.

Repeating the same network topologies and routing algorithms as in inter-chip interconnection networks does not fully take advantage of the on-chip property. Designers are no longer limited by the number of I/O ports they can use. This is a real advantage that has not been stressed enough in the existing solutions.

In this work, a novel approach for efficiently designing NoCs is presented. The adopted design strategy is outlined in section 2 along with the supporting arguments. This includes

an analysis of NoC designs in general and their key properties. A detailed description of the proposed approach is presented in section 3. The basic properties of the FT are analyzed first to lay the background for the proposed improved FT and router architecture. Simulation results, that show the performance of the proposed approach, are presented in section 4 followed by conclusions in section 5.

II. DESIGN STRATEGY

As was explained in the introduction, most NoCs are based on derived solutions from the inter-chip interconnection networks area. Also, the throughput of most of these networks does not go beyond 40% to 50%. This is due to is the unavoidable conflicts in the allocation of router resources in these topologies. Conflicts occur when several packets destinations are on the same path and require to be routed through the same output port. Ideally the router architecture should have more output ports than input ports to prevent such conflicts from occurring.

The main strategy adopted in this work is to increase the number of output ports versus the number of input ports in the router. This might have posed many problems in the context of inter-chip interconnection networks. For NoCs, however, the network is frozen at design time. The number of client nodes (IP cores) is a constant, the network size is also a constant; hence the number of routers too. Adopting a parameterized architecture would enable the use of a single HDL (Hardware Description Languages) model to describe a family of routers with some parameterization, especially I/O ports.

III. THE PROPOSED APPROACH

The proposed approach aims at developing a new class of NoCs based on a sub-class of Multistage Interconnection Networks topology (MIN). More particularly, a class of bidirectional folded MINs; chosen for its properties of enabling adaptive routing. This class is well known in the literature under the name of Fat Trees (FT) [5]. Although known under the same name, this class of networks is very similar but not identical with the original FTs as defined in [16]. The main goal of this approach is to arrive to a topology derived from the FT topology by increasing the number of links in order to eliminate contention.

Before introducing the adopted modified FT topology the regular FT, which is a type of Multi-Stage Interconnection Network (MIN) properties are analyzed below.

A. Fat-Tree MIN Properties

1) Routing

Routing in folded FTs is reduced to routing in a binary tree [5], as shown in Figure 1. The output ports of a router that are connected to the input ports of upper stage routers are labeled as the UP links. The output ports of a router that are connected to the input ports of lower stage routers on the right are labeled RIGHT links. The output ports of a router that are connected to the input ports of lower stage routers on the left are labeled LEFT links. Routing in a tree makes packets take one of the three directions according to their destination address.

A packet is routed up until it reaches a router that has a downward connection for it to reach its destination. This router is called routing summit for convenience. The FT structure, based on a superposition of trees, naturally provides packets with several upward paths. Any upward path will eventually lead to a "summit" where a downward path to the packet's destination is provided. The downward path to the packet's destination is unique as it is the case for any regular tree structure.
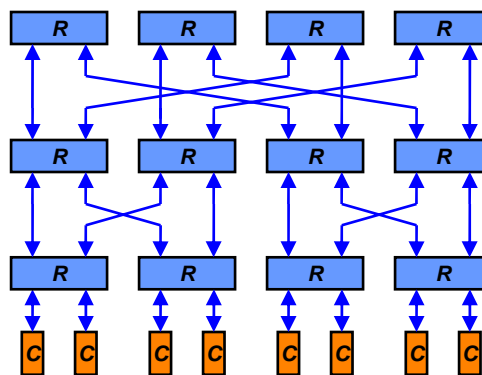

Figure 1 – Regular Fat Tree Topology

2) Contention in FT MINs

Considering a single router, the cases for packet routing are:

- Packets coming from the bottom links are either routed up or routed down. In this case this router is a summit
- Packets coming from the up links are always routed down.

This means that packets coming from the up links are never routed up only packets coming from the bottom links are routed up. Since the number of up links is equal to the number of bottom links, there cannot be any contention when routing up. Contention occurs when going down. Because of the fact that the bottom links are split in right and left links, deterministic routing of packets will lead to contention. Clearly, if several packets coming from the up links need to go right, there will be a contention. This means that one of them will earn the right to use the link while the others will be waiting for it to complete as shown in Figure 2.
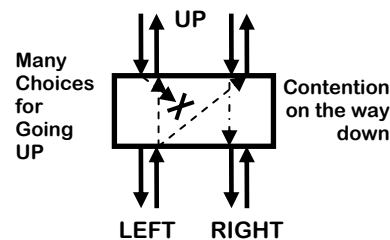

Figure 2 – Contention in FTs

B. Modified FT MIN

Contention can be removed if there are enough output ports in a router so that they can accommodate any combination of incoming packets. Since Contention occurs only on the downward path, doubling the number of output ports in the downward direction only will eliminate the contention. This is the adopted strategy for the proposed modified FT.
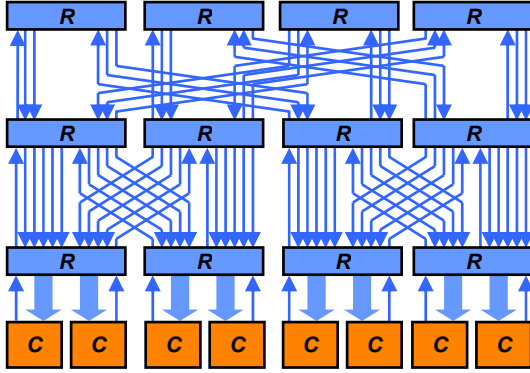

Figure 3 – Modified FT Topology

Figure 3 shows the modified FT where the down links are doubled. Doubling the output ports of a router also means doubling some of the input ports of the adjacent router, of lower stage, to which it is connected. This is needed to be able to connect all the output ports of the upper stage router. To avoid contention in the lower stage router, its output ports are twice as much as its input ports and four times the input ports of its upper stage adjacent router.

This modification does not induce any changes to the routing function which stays the same as for the regular Fat-Tree topology.

Figures 4 and 5 show the adopted router architecture and the simple routing function circuit, respectively. The size of the clients' address space reachable using the downside ports of this router is equal to 2^{r-1} x 2 which is 2^r. It is always a continuous interval of addresses of the form *[l, u]*. The lower bound *l* corresponds to the smallest address that can be reached from the router *(r,c)*. The range having 2^r clients means that the smallest address within the range is obtained by clearing the lowest *r* bits of the column *c*. $l = (c/2^r) x 2^r$. The upper bound *u* corresponds to the largest address that can be reached from the router *(r,c)*. The range having 2^r clients means that the largest address within the range is obtained by adding 2^r to the lower bound *l*. $u = l + 2^r$. The address range associated with each direction can be easily deduced:

- Route all packets UP if destination address is outside of *[l, [* which means it is within the complementary range: *[0, l[U [u, 2^{n+1} -1]*
- Route all packets RIGHT is destination address is within range *[l+2^{r-1}, u[*
- Route all packets LEFT if destination address is within range *[l, l+2^{r-1}]*
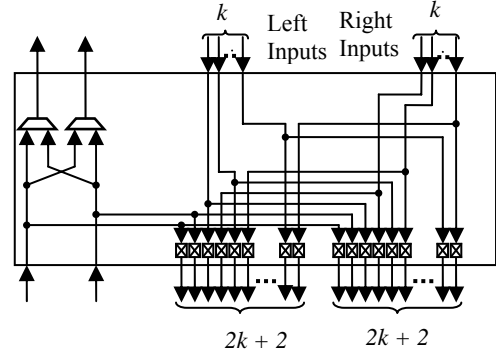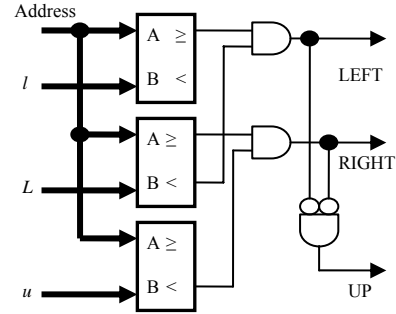

Figure 4 – The router Architecture

Figure 5 – The routing function

IV. PERFORMANCE EVALUATION

Simulations were used to evaluate the performances of the modified FT MIN. The simulation platform is cycle-based and written in C. Two networks were simulated: the regular FT and the modified FT that was proposed. The traffic generator model follows a uniform distribution. Bounded Variable size packets were generated. Figure 6 and Figure 7 show the average latency and the throughput as a function of the input load for networks of 16, 32 and 64 clients for a message length of around 128 bytes. The FT2 label on the figures refers to the modified FT and the FT label refers to the regular FT.
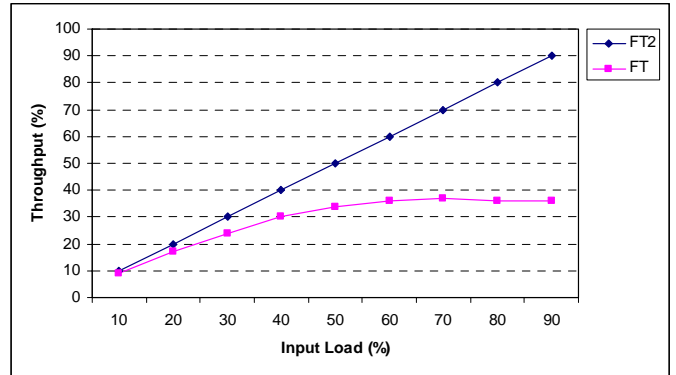

Figure 6 – Throughput

The results clearly show the following improvements achieved by the modified FT over the regular FT:

1. The throughput completely matches the input load even for very high loads (simulated up to 99%) where other networks saturate.
2. The latency incurred is within an acceptable range and does not increase exponentially after an input load of 50% where other networks enter saturation.
3. Simulation results for higher number of clients (up to 64) showed identical performance.
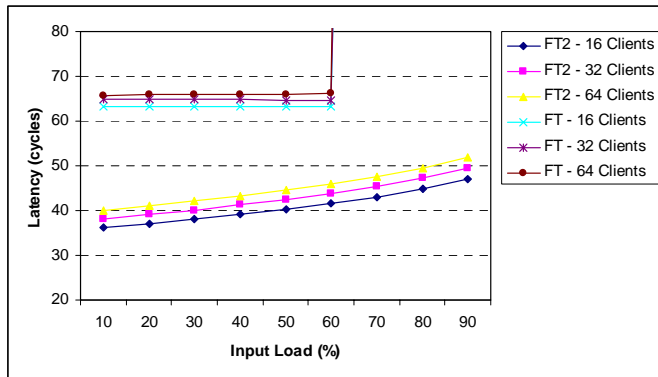
(a)
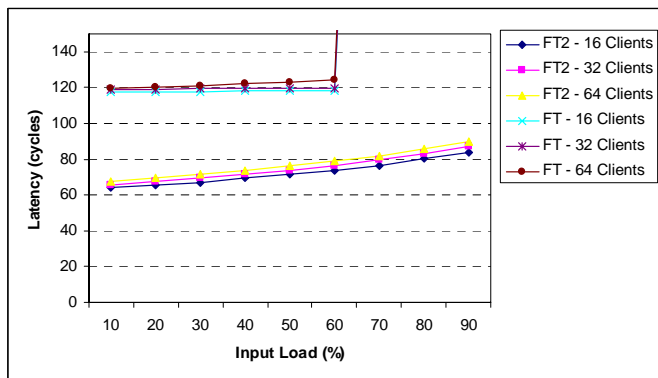
(b)

Figure 7 –   Average Latency for a message length of:
(a) About 64 bytes and (b) about 128 bytes

V. CONCLUSIONS

A novel approach for efficiently designing NoC was introduced. The proposed approach takes full advantage of the on-chip context. The proposed router architecture is very simple, can be fully parameterized and requires no buffering, thus reducing the on-chip imprint of the router. Simulation results show a clear advantage over regular FT.   The throughput completely matched the input load even for very high loads (simulated up to 99%) where other networks saturate.

REFERENCES

[1] L. Benini and G. D. Micheli, "Networks on chips: A new SoC paradigm", *IEEE Computer*, 35(1):70 – 78, January 2002.

[2] W. J. Dally and B. Towles. Route packets, not wires: On chip interconnection networks. In *Proceedings of the 38thDesign Automation Conference*, pages 684–689, June 2001.

[3] J. Henkel, W. Wolf, and S. Chakradhar, "On-chip networks: a scalable, communication-centric embedded system design paradigm," in *Proc. 17th Int'l Conf. VLSI Design*, 2004, pp. 845-851.

[4] A. Hemani, A. Jantsch, S. Kumar, A. Postula, J. Oberg, M. Millberg, and D. Lindqvist, "Network on chip: an architecture for billion transistor era," in *Proc. IEEE NorChip Conf.*, 2000.

[5] P. Guerrier and A. Greiner, "A Generic Architecture for On-Chip Packet-Switched Interconnections", *Proc. IEEE Design Automation and Test in Europe (DATE 2000), IEEE Press, Piscataway, N.J., 2000, pp. 250-256.*

[6] E. Beigne, F. Clermidy, P. Vivet, A. Clouart and M. Renaudin, "An Asynchronous NOC Architecture Providing Low Latency Service and its Multi-Design Framework". In *Proceedings of the 11th Symposium on Asynchronous Circuits and Systems (ASYNC'05).*

[7] P. Zipf, H. Hinkelmann, A. Ashraf, M. Glesner, " A Switch Architecture and Signal Synchronization for GALS System-on-Chips", *SBCCI'04, September7-11 , 2004, Pernambuco, Brazil*

[8] S. K. Hasan, A. Landry, Y. Savaria, M. Nekili, "Design Constraints of a HyperTransport-Compatible Network-On-Chip", *IEEE 2004.*

[9] M.D. Osso, G. Biccari, L. Giovannini, D. Bertozzi and L. Benini, "xpipes: A Latency Insensitive Parameterized Network-on-Chip Architecture For Multi-Processor SoCs", *Proceedings of the 21st International Conference on Computer Design (ICCD'03)*

[10] E. Nilsson. "Design and Implementation of a hot-potato Switch in a Network on Chip". Master's thesis, Royal Institute of Technology, IMIT/LECS 2002-11, Sweden, June 2002.

[11] E. Nilsson, M. Millberg, J. Öberg, and A. Jantsch, Load Distribution with Proximity Congestion Awareness in a NoC, Proceedings of the Design,Automation and Test in Europe Conference and Exhibition (DATE'03)

[12] C. J. Glass and L. M. Ni, "The turn model for adaptive routing". In Proceedings of 19th Annu. Int. Symposium on Computer Architecture, May 1992.

[13] E. Nilsson and J. Öberg, "Reducing power and latency in 2-D mesh NoCs using globally pseudochronous locally synchronous clocking", CODES+ISSS 2004.

[14] E. Nilsson and J. Öberg, "Trading off power versus latency using GPLS clocking in 2D-mesh NoCs", ISSCS 2005

[15] K. Goossens, J. Dielissen, A. Radulescu, "Æthereal network on chip: concepts, architectures, and implementations", IEEE Design and Test of Computers, Volume 22, Issue 5, Sept.-Oct. 2005 Page(s)414 – 421

[16] C. Leiserson, "*Fat-Trees: Universal Networks forHardware-Efficient Supercomputing*", IEEE Transactions on Computers, vol. C-34, no. 10, pp. 892-901, October 1985.