**International Refereed Journal of Engineering and Science (IRJES)**
*ISSN (Online) 2319-183X, (Print) 2319-1821*
*Volume 3, Issue 10 (October 2014), PP.97-101*

# Guaranteed Throughput in An on–Chip Permutation Netwok For Multiprocessor System on Chip Using Fpga

[1]T maheshwar, [2] b laxman, [3] p suresh
[1,2,3] *Department Of Electronics And Communication Engineering, Sree Dattha Institute Of Engineering & Science*

**ABSTRACT**: It is a challenging task in a network-on-chip to design an on-chip switch/router to dynamically support guaranteed traffic permutation in multiprocessor system- on-chip under the constraints of power, timing, area, and time-to-market. The proposed network implies a pipelined circuit-switching approach combined with a dynamic path-setup scheme under a multistage network topology. The dynamic path-setup under probing mechanism defines runtime path arrangement for arbitrary traffic permutations. The circuit-switching approach offers a guarantee of permuted data and its compact overhead enables the benefit of stacking multiple networks.

**Keywords**: FPGA, Guaranteed throughput, multistage interconnection network, network-on-chip, permutation network, pipelined circuit-switching, traffic permutation.

## I. INTRODUCTION

As the complexity of systems-on-chips (SoCs) increases, the network-on-chip (NoC) is being implemented as a scalable communication-centric solution for integrating numerous on-chip components i.e., sub blocks, processing elements(PEs), and intellectual properties (IPs). One of the most difficult tasks in NoC design is guaranteeing throughput or providing a quality-of-service (QoS) mechanism of the traffic offered by the on-chip components, particularly with the very limited cost of the on-chip sources.A trend of multiprocessor system-on-chip (MPSoC) design being interconnected by NOC is currently developing for applications of parallel processing, scientific computing, and so on [1]–[6]. A a traffi c pattern in which each input sends traffic to exactly one output and each output receives traffic from exactly one input, called "permutation traffic" is one of the important traffic classes exhibited from on-chip multiprocessing applications [7], [8].

Standard permutations of traffic occur in general-purpose MPSoCs, such as, polynomial, sorting, and fast Fourier transform (FFT) computations because shuffled permutation, and matrix transposes or corner-turn operations exhibit transpose permutation [6]. Recently, Turbo/LDPC decoding have been developed for application-specific MPSoCs and they exhibit arbitrary and concurrent traffic permutations due to multi-mode and multi-standard feature [3]–[5]Most noc in nature are general-purpose and use routing algorithms such as dimension-ordered routing and minimal adaptive routing. In general permutation traffic patterns, on-chip permutation networks using application-aware routings are needed to have better performance compared to the general-purpose networks [8].application-aware routings are used before running the applications and can be implemented as source routing or distributed routing. The drawback of, such application-aware routings is they cannot handle the dynamic changes of a permutation pattern, which is exhibited in many of the application phases [8]. The difficulty lies in the design effort to compute the routing to support the permutation changes in runtime, as well as to guarantee [9] the permutated traffics. This will cause a great challenge when these permutation networks need to be implemented under very limited on-chip power and area overhead .most on-chip permutation networks with regard to their implementation implies that most the networks employ a packet-switching mechanism to deal with the conflict of permuted data [3]–[6]. Their implementations uses first-input first-output (FIFO) queues for the conflicting data [3], [5],[6], or time-slot allocation in the overall system with the cost of more routing stages [5],. The choices of network design factors, i.e., topology, switching technique and the routing algorithm, have different impacts on the on-chip implementation.

Regarding topology regular direct topologies like mesh and torus are feasible for 2-D chip design where as indirect topologies like butterfly and Benes pose a challenge for physical implementation because of high wiring irregularity and larger router radix. However, an arbitrary permutation pattern with its high load on individual source-destination pairs stresses the regular topologies and that may lead to throughput degradation.

Hence on chip permutation pattern uses indirect multistage topologies. Regarding switching packet switching is takes the excessive amount of chip power and area and regarding routing deflection routing is not energy efficient Compared to minimal routing.

This paper presents a novel silicon-proven design of an on-chip permutation network to support guaranteed throughput of permutated traffics under arbitrary permutation. Unlike conventional packet-switching approaches, our on-chip network employs a circuit-switching mechanism with a dynamic path-setup scheme under a multistage network topology. The dynamic path setup tackles the challenge of runtime path arrangement for conflict-free permuted data. The pre-configured data paths enable a throughput guarantee.

This paper further organized as follows section II describe on chip network topology and switch by switch inter connection section III describes dynamic path setup to support path arrangement section IV describes the common switch architecture section V describes the conclusion.

## II. ONCHIPNETWORK TOPOLOGY AND SWITCH BY SWITCH INTERCONNECTION

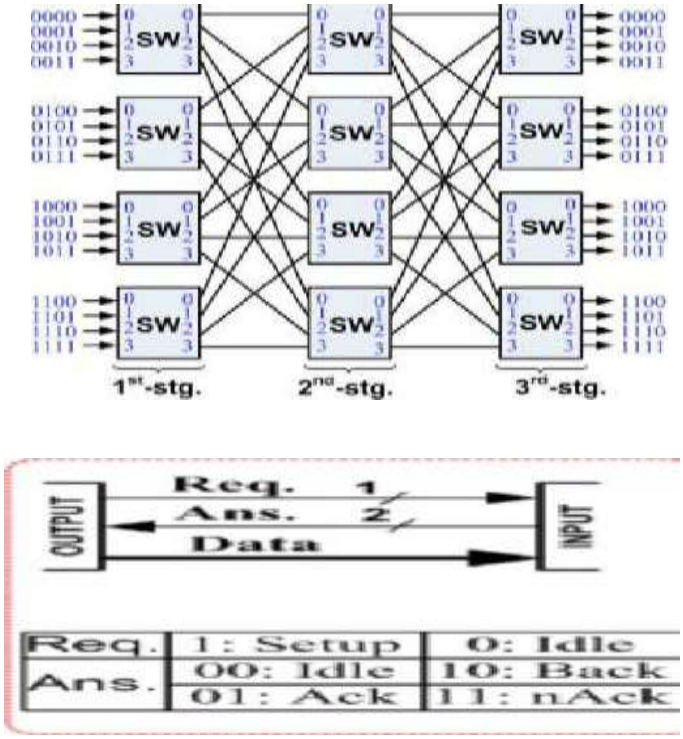



**Figure.1**. Proposed on-chip network topology with port addressing scheme and format of handshake signal

A network topology clos network is applied to build scalable commercial multiprocessors with thousands of nodes in macro systems. A three-stage Clos network is defined as c (n,m,p) where n represents the number of inputs in each of p first-stage switches and m is the number of second-stage switches. In order to support a parallelism degree of 16 as in most practical MPSoCs we proposed to use c(4,4,4) as a topology for the designed network (see Fig. 1). This network contains rearrangable property that can realize all possible permutations between its input and outputs. The choice of the three-stage Clos network with a modest number of middle-stage switches is to minimize implementation cost, whereas it still enables a rearrangable property for the network.

A pipeline circuit switch proposed for the design of network on chip which consists of three phase namely set up, transfer and release phase .a dynamic path set up scheme supporting run time path arrangement occur in the setup phase .Two support this circuit switching a switch by switch interconnection with hand shake signal is proposed. The handshake signal includes the bit format in which 1 bit Request (Req) and 2 bit Answer (Ans). Req1 is used when a switch requests an idle link and further which leads downstream switch in the setup phase. The Req1 is also kept during data transfer along the set up path. A Req0 denotes that the switch releases the occupied link and become idle. This code is also used in both the setup and the release

phases. An Ans01 (Ack) means that the destination is ready to receive data from the source. When the Ans01 propagates back to the source, it denotes that the path is set up, then a data transfer can be started immediately. An Ans11(nAck) is reserved for end-to-end flow control when the receiving circuit is not ready to receive data due to being busy with other tasks, or overflow at the receiving buffer, etc. Ans10 (Back) means that the link is blocked.

## II. DYNAMIC PATH SETUP SUPPORT TO PATH ARRANGEMENT

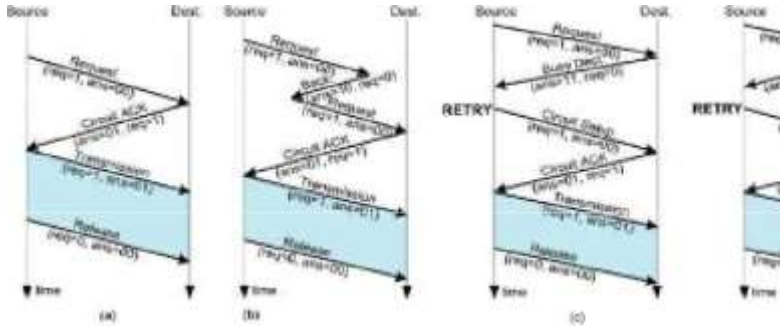



**Figure.2**. Example of EPB with switch to switch handshake signal

Dynamic path set up is the key point of the design which supports the run time path arrangement when permutation is changed. Dynamic path set up in which a path stars from input in search of corresponding output, is based on the dynamic probing mechanism. in which a probe (or setup flit) is dynamically sent under a routing algorithm in order to establish a path towards the destination. Exhausted profitable backtracking (EPB) is proposed to use to route the probe in the network. Now the question arises that EPB based probing mechanism can support all possible permutation between input and output of proposed clos network. A path arrangement with full permutation consists of sixteen path setups, whereas a path arrangement with partial permutation may consist of a subset of sixteen path setups.AS the clos network rearrangable if m>n or m=n. the proposed clos network with m=n=4 so can rearrangable. Directly applying the Exhaustive Property of the search into rearrangable c (4, 4, and 4) shows that the EPB-based path setup can always find an available path within the set of four possible paths between the input and the idle output. Based on this EPB-based path-setup scheme, it is obvious that the path arrangement for full (as well as partial) permutation can always be realized in the in proposed network with close topology

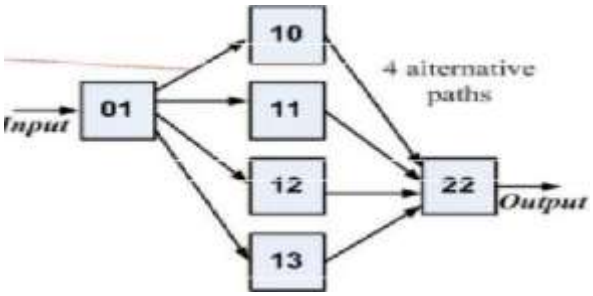



**Figure.3**. Switch-by-switch interconnection and path-diversity capacity

Each Input sends a probe containing a 4-bit output address to find a path leading to the target output..During search the probe move forward if found a free link. if it is found block link then it backtrack for the available link by this way the probe found a available link between the inputs and outputs .fig 3 explains a path set up between the input and output. The input named switch (01) or source trying to set up a path to the destination named switch (22) first, the probe will non-repetitively try paths through the second-stage switches in the order of 10-11-12-13.

Assuming that the link 01-10is available, the probe first tries this link and (REQ10 then arrives at switch 10 If

link 10-22 is available, the probe arrives at switch 22 and meets the target output. An Ans=Ack then propagates back to the input to trigger the transfer phase. If link 10=22 is blocked, the probe will move back to switch 01(Ans=Ack) and link01-10 is released (REQ=0) from switch 01, the probe can then try the rest of idle links leading to the second-stage switches in the same manner. By means of moving back when facing blocked links and trying others, the probe can dynamically set up the path in runtime in a conflict avoidance manner.

## IV. SWITCHING NODE DESIGNS

Three kinds of switches are proposed for the on chip network

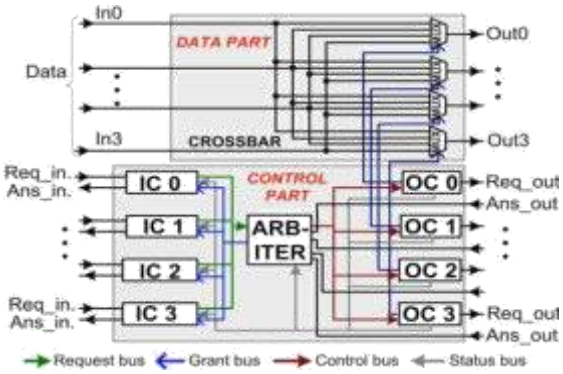
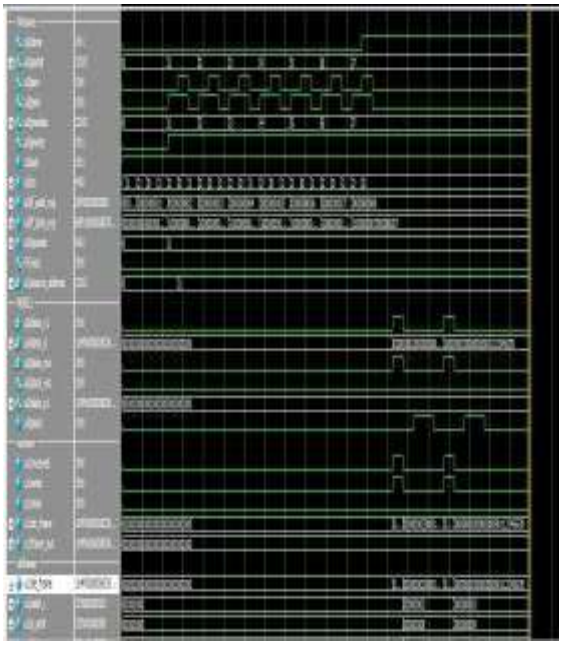



**Figure.4.** Common switch architecture

These switches are based on the common switch architecture shown in the figure4.They input control(IC), output control(OC) and arbiter and crossbar .arbiter has two function first it connect Ans_Out and IC through grant bus second, as a referee for the requests from the ICs. The IC is implemented with finite-state machine (FSM). The probe routing algorithm and the operation of the switches are controlled by the FSM implementation in the IC. In order to support the probing path setup, ICs are implemented with different probe routing algorithms depending on its switch stage.

## V. SIMULATION RESULTS

Verilog language is used to simulate the proposed structures in this section. Dynamic path set up scheme under multistage network can be realized in FPGA.



## VI. CONCLUSION

This paper has presented an on-chip network design supporting traffic permutations in MPSoC applications. By using a circuit-switching approach combined with dynamic path-setup scheme under a Clos network topology, the proposed design offers arbitrary traffic permutation in runtime with compact implementation overhead. A silicon-proven test-chip validates the proposed design and suggests availability for use as an on-chip infrastructure-IP supporting traffic permutation in future MPSoC researches.

## REFFERENCES


[1]. [1] S. Borkar, "Thousand core chips—A technology perspective," in Proc.ACM/IEEE Design Autom. Conf. (DAC), 2007, pp. 746–749.

[2]. P.-H. Pham, P. Mau, and C. Kim, "A 64-PE folded-torus intra-chip communication fabric for guaranteed throughput in network-on-chip based applications," in Proc. IEEE Custom Integr. Circuits Conf.(CICC), 2009, pp. 645–648.

[3]. C. Neeb, M. J. Thul, and N.Wehn, "Network-on-chip-centric approach to interleaving in high throughput channel decoders," in Proc. IEEE Int.Symp. Circuits Syst. (ISCAS), 2005, pp. 1766– 1769.

[4]. H. Moussa, A. Baghdadi, and M. Jezequel, "Binary de Bruijn on-chip network for a flexible multiprocessor LDPC decoder," in Proc. ACM/IEEE Design Autom. Conf. (DAC), 2008, pp. 429–434.

[5]. H. Moussa, O. Muller, A. Baghdadi, and M. Jezequel, "Butterfly and Benes-based on-chip communication networks for multiprocessor turbo decoding," in Proc. Design, Autom. Test in Euro. (DATE), 2007,pp. 654–659.

[6]. S. R. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz,D. Finan, A. Singh, T. Jacob, S. Jain, V. Erraguntla, C. Roberts, Y.Hoskote, N. Borkar, and S. Borkar, "An 80-tile sub-100-w Tera FLOPS processor in 65-nm CMOS," IEEE J. Solid-State Circuits, vol. 43, no.1,

[7]. pp. 29–41, Jan. 2008.

[8]. W. J. Dally and B. Towles," Principles and Practices of Interconnection Networks",:. San Francisco, CA: Morgan Kaufmann, 2004.

[9]. N. Michael, M. Nikolov, A. Tang, G. E. Suh, and C. Batten, "Analysisof application-aware on-chip routing under traffic uncertainty," in Proc.IEEE/ACM Int. Symp. Netw. Chip (NoCS), 2011, pp. 9–10.

[10]. P.-H. Pham, J. Park, P. Mau, and C. Kim, "Design and implementation of back tracking wave-pipeline switch to support guaranteed throughput in network-on-chip," IEEE Trans. Very Large Scale Integr.(VLSI)Syst.,10.1109/TVLSI.2010.2 096520