

# A Simple, Low-Cost and Low-Power Switch Architecture for NoCs

Mahdi Nazm-Bojnordi, Mehdi Semsarzadeh, Abbas Banaiyan, and Ali Afzali-Kusha  
{m.bojnordi, m.semsar, a.banaiyan}@ece.ut.ac.ir, afzali@ut.ac.ir  
Low-Power High-Performance Nano-Systems Laboratory  
Electrical and Computer Engineering Department, University of Tehran, IRAN

**Abstract**— Considering the typically large use of NoCs as a solution to alleviate complexity of the current SoCs, designing an efficient NoC is an important key issue. This paper proposes a simple, low-cost and low-energy switch architecture for the NoCs using mesochronous clocking scheme for on-chip communication. Switches are designed in Verilog HDL. Experimental results show that the 4-ports instances of our proposed architecture are more efficient to be used as a basic element for constructing the NoCs.

**Index Terms**— GALS, globally asynchronous and locally synchronous, synchronizer, mesochronous clocking, transmitter, and receiver.

## I. INTRODUCTION

Nowadays, the major challenge for System on Chip (SoC) designers is to tame problems of deep sub-micron effects under timing, area, and low power constraints arisen during the system development as their functionalities and applications increase. Network on Chip (NoC) is a new paradigm in designing the efficient SoCs which exploit the multi-billion transistor capacity composed of hundreds interconnected IP blocks, such as CPU and DSP cores, application specific hardware and large amounts of memory. It also has a profound impact on simplifying the on-chip clock distribution networks[1]. Since in the NoCs communication between the cores and switches is path-specified, so they are more scalable and power efficient than the bus-based SoCs.

Basic idea for the NoCs is borrowed from traditional large-scale multi-processors and the wide-area networks domain. According to the ITRS (International Technology Roadmap for Semiconductors), by the end of the decade, using 50-nm transistors operating below one volt, SoCs will grow to 4 billion transistors running at 10 GHz. On-chip physical interconnections will present a limiting factor for performance and, possibly, energy consumption [2].

NoCs can be defined as a structured set of switches and point-to-point links interconnecting the processing cores of an SoC, in order to support communication among them. Such a structure can be described as a graph with switches on the nodes and links on the arcs, and it is named *topology* [3]. An important design decision for NoCs is the choice of topology. Some researchers envision NoCs as regular topologies (such as mesh networks, multi-stage and fat-trees), which are suitable for interconnecting homogeneous cores in a chip

multiprocessor. However, many SoCs involve heterogeneous cores having varied functionality, size and communication requirements. If a regular interconnect is designed to match the requirements of few communication-hungry components, it is bound to be largely over-designed with respect to the needs of the remaining components. This is the main reason why most current SoCs use irregular topologies like bridged busses and/or dedicated point-to-point links and irregular on-chip network [4].

In such systems, synchronization is a very important key issue[9]. There are two common methods available for synchronization of these systems. The most commonly used method is the synchronous system. In this system a global clock is distributed over the system with low skew. This clock is used to synchronize all the events and transactions in the system. Another method is running the system completely asynchronous. This method needs to use handshaking or special timing circuitry for both computations and communications in order to keep synchronization within the cores. These two methods have some problems which make them inefficient for the new SoCs. In the synchronous method, the system consumes a huge amount of power; also its speed is limited to the global clock signal. The asynchronous method has more design overhead and handshaking signals; consequently it becomes hard to check for its correct functionality. There is a third method uses the profits of the two former methods and has not the drawbacks of the globally synchronous and asynchronous systems and uses the synchronous processing blocks which communicate asynchronously with each other –better known as the globally asynchronous, locally synchronous (GALS) methodology [9]. With respect to the current increase in the number of different clock domains used on a single chip, this is a very promising overall technique to use for IP block integration. A scheme of the GALS methodology is called the Mesochronous clocking technique [8]. This method of asynchronous communication reduces the handshaking overhead to just a strobe signal. According to the development of deep sub-micron technologies and the computational requirements of new applications of SoCs (such as multimedia) and their needs to communication on the chip, mesochronous clocking is a better solution which is more reliable and robust [8].

In current SoC environments there are many communication