



Asynchronous switching for low-power networks-on-chip

Magdy A. El-Moursy^{a,*}, Heba A. Shawkey^b

^a Mentor Graphics Corporation, Cairo, Egypt

^b Microelectronics Department, Electronics Research Institute, Cairo, Egypt

ARTICLE INFO

Article history:

Received 24 April 2011

Received in revised form

27 September 2011

Accepted 10 October 2011

Available online 27 October 2011

Keywords:

NoC

Power dissipation

Asynchronous

ABSTRACT

Asynchronous switching is proposed to achieve low power Network on Chip. Asynchronous switching reduces the power dissipation of the network if the activity factor of the data transfer between two ports α_{data} is less than $A\alpha_c + B\alpha_{clk}$. Closed form expressions for power dissipation of different network topologies are provided for both synchronous and asynchronous switching. The expressions are technology independent and are used for power estimation. Asynchronous switching is compared with synchronous switching for different network densities N/L_cXL_c . The area of the asynchronous switch is 50% greater than the area of the synchronous switch. However, the power dissipation of asynchronous switching decreased by up to 70.8% as compared to the power dissipation of the conventional synchronous switching for Butter-Fly Fat Tree (BFT) topology. Asynchronous switching is more efficient in CLICHE topology than in both BFT and Octagon topologies achieving higher power reduction 75.7%. Asynchronous switching becomes more efficient as technology advances and network density increases. A reduction in power dissipation reaches 82.3% for 256 IPs with the same chip size. Even with clock gating, asynchronous switching achieves significant power reduction 77.7% for 75% clock activity factor.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

As the difficulty of increasing the frequency at which Integrated Circuits IC can operate increases, the industry is moving towards more functionality rather than higher frequency. System on Chip (SoC) with increasing number of cores in a single chip is the trend in today's and tomorrow's chip design. Integrating more Intellectual Properties (IPs) in the chip requires more communication channels among those IPs. More metal layers (however necessary) is not sufficient to provide the required capacity of communication. Networks on Chip (NoC) are emerging as natural expansion for the existing dedicated wires. Like the telephone and computer networks when fully connected nodes are not feasible, switches and routers are used to efficiently use the resources while reducing the overhead [1–15]. Many architectures have been proposed for NoC. Butterfly Fat Tree (BFT), Octagon, and CLICHE are among the well accepted power-efficient architectures [14].

Synchronization, clock delivery, process variation, and signal integrity are among the big challenges of large chips with many IPs. Asynchronous communication has been investigated and

used in many systems [16–21]. Very few articles have discussed synchronization in NoC. Previous research in synchronization in NoC [22–27] considered few case studies without analytical analysis or general guidelines to use synchronous or asynchronous mechanism. Also, limited number of architectures are considered in the published articles. Asynchronous communication could be more efficient in NoC since the system is big and has many IPs. However, applying the asynchronous transfers among all communicating blocks and sequential elements is an expensive solution. Handshaking overhead, if applied thoroughly, slows down the system and eats up more resources in terms of metal resources and power dissipation. However, Globally Asynchronous Locally Synchronous (GALS) methodology could solve the global synchronization problem with less overhead [16]. The conventional synchronous and proposed asynchronous systems are shown in Fig. 1. The conventional synchronous system which requires a clock distribution network to deliver the clock signal to all sequential elements of the system is shown in an abstract block diagram in Fig. 1(a). The asynchronous system does not require the clock network. Alternatively, handshaking signals are necessary to synchronize the transmitter and the receiver IP as shown in Fig. 1(b).

Low-power is becoming number one concern in chip design nowadays. Building a system without evaluating how much power it may dissipate or without looking at different power efficient alternatives, may lead to a useless system. Many power

* Corresponding author.

E-mail addresses: magdy_el-moursy@mentor.com,
magdyaelmoursy@gmail.com (M.A. El-Moursy),
heba_shawkey@eri.sci.eg (H.A. Shawkey).

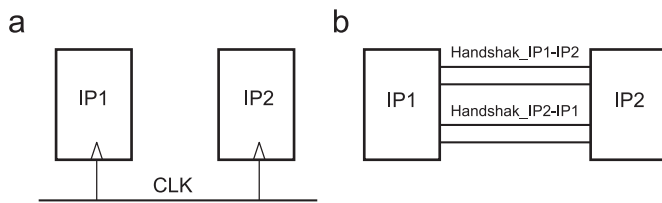


Fig. 1. IP interface (a) synchronous system and (b) asynchronous system.

efficient techniques have been proposed. Clock distribution network along with its loads represent 40–70% of the power dissipation of a single chip [28]. Clock gating is very efficient in reducing power dissipation of the clock distribution network. However, clock gating is efficient when the system is idle for long time. Applying clock gating techniques increases the overhead since the gating logic may itself dissipate considerable amount of power if the system is active most of the time.

In this paper GALS methodology is adopted as an efficient technique to perform on-chip synchronization in NoCs. GALS system is compared with a fully synchronous system to demonstrate the efficiency of the proposed synchronization technique. The paper is organized as follows: In Section 2, synchronous switching is described. Asynchronous switching is presented in Section 3. In Section 4, closed form expressions for power dissipation in both asynchronous and synchronous switching systems are provided. Some simulation results are presented in Section 5. The primary conclusions are provided in Section 6.

2. Synchronous switching

In synchronous switching system, the clock is distributed for all components of the chip. Clock distribution networks consist of two primary parts; global clock distribution network and local clock distribution network. Global clock network is the network of interconnects and repeaters which deliver the clock signal from the clock source of the chip to all blocks (switches and IPs in NoC). In addition to the interconnects and repeaters of the network, clock loads which are driven by the global clock network are important part of the network as described in Section 4. Given the symmetric structure of a network of IPs, H-tree clock network is adopted to distribute the global clock signal of the synchronous system [29–31]. H-tree guarantees minimum distance from the clock source to the entry of the local clock distribution of all switches and IPs as shown in Section 4. H-tree network is used to deliver the clock signal for all network topologies which are considered in this paper (BFT, CLICHE, and Octagon). In IC design, global clock network design is decoupled from local clock network design. Since GALS is adopted for the asynchronous switching system, same local clock distribution (except the global clock loads) is used for both synchronous and asynchronous systems. It is out of the scope of this paper to determine the distribution of the local clock network since it is common part for both synchronous and asynchronous systems.

The primary active element in an NoC is the switch which connects different IPs. Each NoC switch contains number of input and output ports. Different topologies differ in the number of input and output ports. Each port of a switch contains input FIFO, output FIFO, header decoder, and controller. The synchronous port structure is shown in Fig. 2.

In synchronous system, two signals are needed to prevent overflow in the input and output FIFOs. The *Write* and *Full* signals are used to control the operation of the synchronous port. The *Write* signal is used to indicate a full source (output) FIFO to the destination port requiring the destination to give the source port

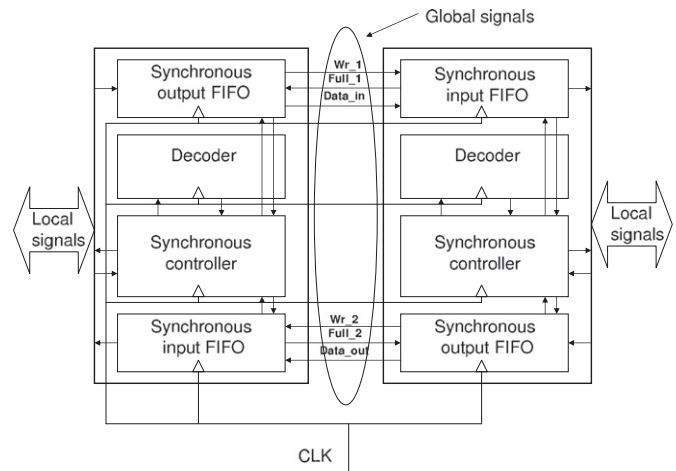


Fig. 2. Synchronous port interface.

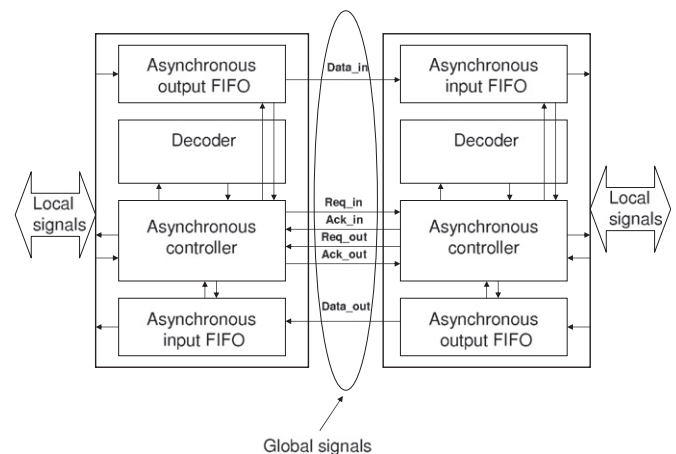


Fig. 3. Asynchronous port interface.

high priority to avoid backlog. The *Full* signal is sent from the destination port to the source port indicating that no more data could be accepted in the input FIFO. *Write* and *Full* signals are important flow control signals to avoid overflow and deadlock. In asynchronous switching, handshaking signals are sufficient for flow control. Asynchronous switching is presented in Section 3.

3. Asynchronous switching

In GALS system, communication within the port modules is still performed synchronously. Alternatively, communication is performed asynchronously in the global level between blocks (IPs and Switches). Using asynchronous switching system, the global clock distribution network is eliminated. Instead, *request/acknowledgment* protocol is used to coordinate each data transfer between two ports (in two communicating switches or in a communicating switch and IP). The asynchronous port structure is shown in Fig. 3. Asynchronous blocks are designed and implemented to build the asynchronous system. The asynchronous blocks require *request* and *acknowledge* signals for handshaking.

In NoC the transfer of data is in the form of messages. Each message is divided into fixed length flow control units (flits) [15]. When the output FIFO stores one whole flit, it sends a *request* signal to controller. If it is a header flit, the header decoder determines the destination. The controller checks the status of destination port. If it is available, the path between input and

Download English Version:

<https://daneshyari.com/en/article/542069>

Download Persian Version:

<https://daneshyari.com/article/542069>

[Daneshyari.com](https://daneshyari.com)