



High Throughput Asynchronous NoC Design under High Process Variation



Rabab Ezz-Eldin^a, Magdy A. El-Moursy^{b,*}, Hesham F.A. Hamed^c

^a Electronics Engineering Department, Bani-Suef University, Bani-Suef, Egypt

^b Mentor Graphics Corporation, Cairo, Egypt

^c Electrical Engineering Department, Minia University, El-Minia, Egypt

ARTICLE INFO

Article history:

Received 2 January 2014
 Received in revised form
 28 October 2014
 Accepted 28 October 2014
 Available online 12 November 2014

Keywords:

Process variation
 Network on Chip
 Interconnect
 Clock skew
 NoC topologies
 Synchronous design
 Asynchronous design

ABSTRACT

Asynchronous switching is proposed as a robust design to mitigate the impact of process variation in Network on Chip (NoC). Circuit analysis is used to evaluate the influence of process variation on both synchronous and asynchronous designs. The impact of process variation is evaluated on different NoC topologies. Network on chip interconnects and clock distribution network are considered under process variation with the advance in technology. The variation in logic and interconnect are included to evaluate the delay, throughput and leakage power variation with different NoC topologies. In addition, the delay and throughput variation are evaluated for clock distribution network. For asynchronous NoC design, the throughput negligibly decreases under high process variation conditions in different NoC topologies. The throughput variation for synchronous design in all topologies rapidly decreases by up to 25% at the same variation conditions.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

With increasing number of cores, process variation (PV) is taking a lot of attention since it is dominating the manufacturing process issues in today's and tomorrow's technologies [1]. PV is inevitable in semiconductor manufacturing processes and it reflects on power consumption, performance and reliability of the circuit. It is becoming more challenging to determine the circuit performance with the continuous change in the circuit elements (logic gates and interconnects). Process variation has two sources; systematic and random. With technology scaling down, random variation becomes significantly larger than systematic variation [2]. Synchronous and asynchronous NoC designs are greatly influenced by process variation. One of the major problems in NoC design is the considerable mismatch between two identical devices that can occur when the amount of random variation increases. The effect of process variation on NoC has become a major issue with rapid technological evolution. Synchronizing big NoC is becoming more challenging under severe process variations. Data transfer in NoC could be done synchronously or

asynchronously. Asynchronous NoC scheme increases the area overhead and tend to be slow [3]. Nonetheless, the power and performance of the circuit can be improved with asynchronous NoC design [4]. Moreover, asynchronous NoC has the ability to avoid the clock skew and achieve robust circuit operation.

Gate process variation causes fluctuations in MOS parameters which make the manufactured gates different than the designed ones. Gate-length and threshold-voltage variations are the most influential variation parameters on logic gates. On the other hand, gate delay decreases as technology scales down while interconnect delay increases. NoC interconnects are becoming a major limiting factor for network performance. The propagation delay increases quadratically with the interconnect length. In addition, interconnect parameters determine the clock signal characteristics. The effect of process variation on interconnect lines is not negligible any more. Both, gate delay variation and interconnect delay variation need to be considered.

The variation of delay and leakage power impacts the functionality, yield and reliability of integrated circuits [5–7]. The logic gate variation causes uncertainty in the power consumption of the design [8]. With the technology scales down, the leakage power becomes significantly large and the trend is predicted to increase in future technologies. Therefore, the evaluation of the leakage power during random PV is essential for designing nanoscale CMOS circuits [9,10].

* Corresponding author.

E-mail addresses: rabab.ezz@eng.bsu.edu.eg (R. Ezz-Eldin),
magdy_el-moursy@mentor.com (M.A. El-Moursy),
hfab66@yahoo.com (H.F.A. Hamed).

The impact of process variation on NoC switch is presented in [11] along with a methodology to enhance the performance of the communication and reduce the average packet latency. In [12], the static process variation is studied with its effect on all main components of NoC switch. In [13,14], the impact of process variation on logic gates is provided while neglecting the interconnects. Other study focuses only on the influence of process variation on NoC interconnects [15,16]. In [17], the frequency variation in switches and links is presented under the process variation using 45 nm technology. The impact of process variation on asynchronous and synchronous NoC switches in addition to its effect on interconnect and clock network distribution are presented in [18,19]. The main focus of this paper is to demonstrate the impact of process variation on NoCs for different topologies. Moreover, synchronous and asynchronous switches are built to determine the delay, throughput and leakage power under sever process variation for large NoCs. The throughput variation of the NoC with different technology nodes is determined to possess the trend shown in Fig. 1.

The paper is organized as follows. In Section 2, different NoC switching schemes are adopted. NoC interconnection base on different NoC topologies are described in Section 3. The impact of high process variation on NoC performance is presented in Section 4. In Section 5, simulation results are provided. Conclusions are demonstrated in Section 6.

2. NoC switching

NoC infrastructure is composed of switches, interconnects and network interface controller. In order to determine the network throughput under process variation conditions, synchronous and asynchronous NoC switches are designed. Different asynchronous NoC designs are presented, in ANoC [20], MANGO [21], QNoC [22], QoS [23] and ASPIN [24]. The packets are divided into fixed length flow control units (flits). The first flit (header) of the packet includes the coordinates (X,Y) of destination address. The deterministic XY routing algorithm is used to rout the packet from input port to the output port. At each switch, the destination address is looked up and the routing path is determined depending on hardware implementation. Round-Robin arbitration algorithm is employed to decide which one of the input ports will access the output port according to the requests. If the output port is busy, the header flit and all subsequent flits will be blocked in the buffers of input port. The routing request is scheduled until a connection between input and output ports is established. Each switch is connected with its neighbor using multiple number of interconnects. While, each Processing Element (PE) is connected to a local port of switch through a network interface controller. In Section 2.1, asynchronous NoC switch architecture is described. The corresponding design for synchronous NoC switch is presented in Section 2.2. The network interface controller for synchronous and asynchronous switches is introduced in Section 2.3.

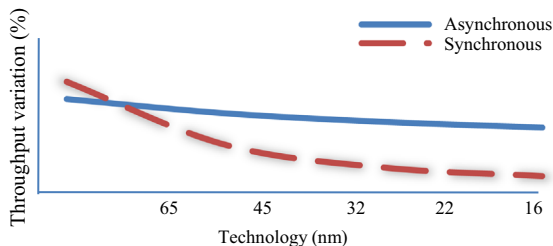


Fig. 1. Throughput under process variation with technology generation for synchronous and asynchronous designs.

2.1. ASynchronous Switch (ASS)

Bidirectional ports are used in the design of ASS. The Input Port (IP) is divided into two main parts, converter stage and XY routing algorithm. The converter stage includes Dual-to-Single Converter (DSC), asynchronous single rail FIFO and Single-to-Dual Converter (SDC) as shown in Fig. 2. The Output Port (OP) is composed of two main modules; module to perform the scheduling of round-robin algorithm and dual-rail module. The handshake protocols are the bundled-data encoding for single rail protocol and the delay-insensitive encoding for dual rail protocol [25]. The conversion of protocols is exploited to reduce the delay between the control and data lines which exists in bundled-data encoding. Using dual rail encoding, the request signal is embedded in data signals, and the number of data lines is doubling. Furthermore, dual rail encoding increases the efficiency of data transmission [24]. Two-phase handshaking is selected for the proposed switch to organize the data transfer. The incoming packets are directed to converter stage in ASS. The header of packet is separated to extract the destination address. The destination address is compared with the address of local switch to direct the packet to certain output port. More than one input port may simultaneously send a request to the same output port. The round-robin arbiter is employed to allow only one input port to access an output port. When the packet arrives to output port successfully, acknowledgment signal is generated to complete the handshaking sequence and permit the next packet to access input port. When the packet is directed incorrectly, the packet will be discarded. A full design for the ASS is implemented and the netlist is realized.

2.2. Synchronous Switch (SYS)

Synchronous switch is designed to compare it with ASS. The synchronous FIFO and the module to perform XY routing algorithm are the main components in the IP, while the round-robin scheduling module is the main block in the OP as shown in Fig. 3. When the incoming packet is received, Write signal is asserted to store data in synchronous FIFO. The destination address is extracted. Full signal is asserted after the flit is stored and the XY routing algorithm module sends request to output port to access the port and receive the incoming flit. When the output port receives more than one request simultaneously, round-robin arbiter is used to select one input port to allocate the output port to serve the incoming request. A communication path is established between input port and the dedicated output port to send all the subsequent flits of the corresponding packet until the tail flit. Once the output port finishes transferring the current flit,

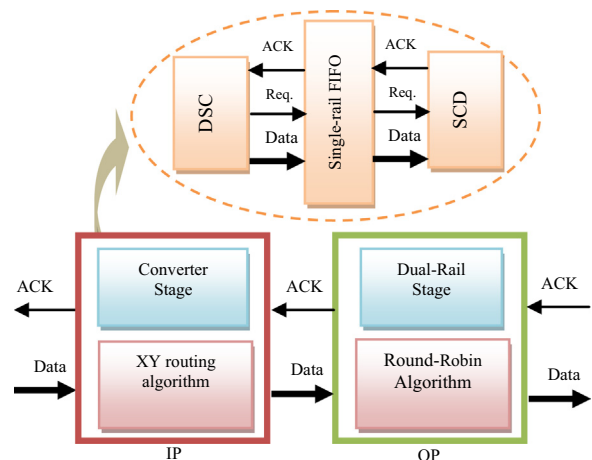


Fig. 2. Input/output port of ASS.

Download English Version:

<https://daneshyari.com/en/article/540976>

Download Persian Version:

<https://daneshyari.com/article/540976>

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