To Study the Impact of Process Variations on NoC Performance: A Circuit-Centric Approach

A THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF TECHNOLOGY

BY

ANUSHKA

ELECTRONICS AND COMMUNICATION ENGINEERING

INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI- 110020

December 2023

THESIS CERTIFICATE

This is to certify that the thesis titled **To Study the Impact of Process Variations on NoC Performance: A Circuit-Centric Approach**, submitted by **Anushka**, to the Indraprastha Institute of Information Technology Delhi, for the award of the degree of **Master of Technology**, is an original research work carried out by her under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

Dr. Sujay Deb
Thesis Supervisor
Professor
Dept. of Electronics and Communication Engineering
IIIT Delhi, 110020

Place: New Delhi

Date: 21^{st} December 2022

ACKNOWLEDGEMENTS

I want to thank my guide, Dr. Sujay Deb for his constant guidance and support throughout the duration of this work. His approach towards helping me formulate the problem statement through thorough exploration and developing the work plan was very helpful.

I would like to acknowledge the contribution of my lab colleagues (members of AMS Lab) who helped me at various critical stages of the thesis work. I would like to thank them for regular brainstorming sessions, guidance with tool-related work, and for stimulating discussions on work-related and unrelated subjects.

Finally I would like to thank my parents for their love and support throughout the journey.

ABSTRACT

Due to technology scaling, achieving increasing bandwidth demands has become challenging and has led to the preference of Network-On-Chips (NoCs) over bus architecture as the communication infrastructure in multi-core systems-on-chips (MPSoCs). Simulation measurements of NoC performance metrics are used to estimate the expected real-time-performance of the NoC across various traffic types during the design stage. For timing-critical applications and advanced technology nodes, the granularity of simulation measurements becomes important. As per the existing literature, simulation is done using cycle-accurate system-simulators or at the register-transfer level (RTL), which takes behavioral description from the designer and uses standard cells from technology libraries to implement the design at the gate level. In this work, we implement a basic NoC design at the circuit level through different digital logic gate implementations for different parts of NoC router. This allows us to exploit various optimizations at circuit level which cannot be obtained by system-simulators or synthesis tools. Performance parameters are measured for these circuits, and Monte-Carlo simulations are done to estimate variations expected in these metrics. The delay and variation measurements are used to compare the performance as well as robustness of various implementations. Some applications may require fast operation, while some may require minimal variations in performance at the cost of speed. The approach of circuit-level analysis presented in this work is used to determine the suitability of a design for different traffic types and application requirements.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Why Network-on-Chip?

Two primary factors guiding chip design choices are application requirements and cost. Increasing performance requirements within a limited budget and design effort have led to a shift in the trend of architectures with a fast single processor to those with multiple simpler processors on a chip. As multiple processors, each with its own cache, run in parallel and use the same memory, a reliable communication infrastructure allowing data transfer between these processors is required. The traditional way is to connect the processing cores to a common bus and use bus protocols for data transfer. Increase in bandwidth requires wider buses, while increase in the number of cores increases traffic and hence the load over the bus, which may lead to congestion. Bus architecture thus becomes inefficient compared to Network-on-Chip (NoC), also known as On-Chip Network (OCN) or interconnection network, because it provides point-to-point connections. NoCs allow the routing of data packets along multiple possible paths from different message sources simultaneously, thereby allowing large bandwidth using multiple links of small width instead of using a single bus of large width.

Design choices are made at the architecture and micro-architecture levels. During operation, the NoC may not perform as well as expected from such an analysis. As manufacturing variations have a much bigger impact on smaller devices, deviations in performance will be larger as devices are scaled down.

The block diagram of an NoC router with its circuit blocks is shown in 1.1.

1.2 Related Work

Once deviations in expected performance due to process variations (PVs) are calculated, different approaches can be taken to account for them. In Nicopoulos *et al.* (2010), the

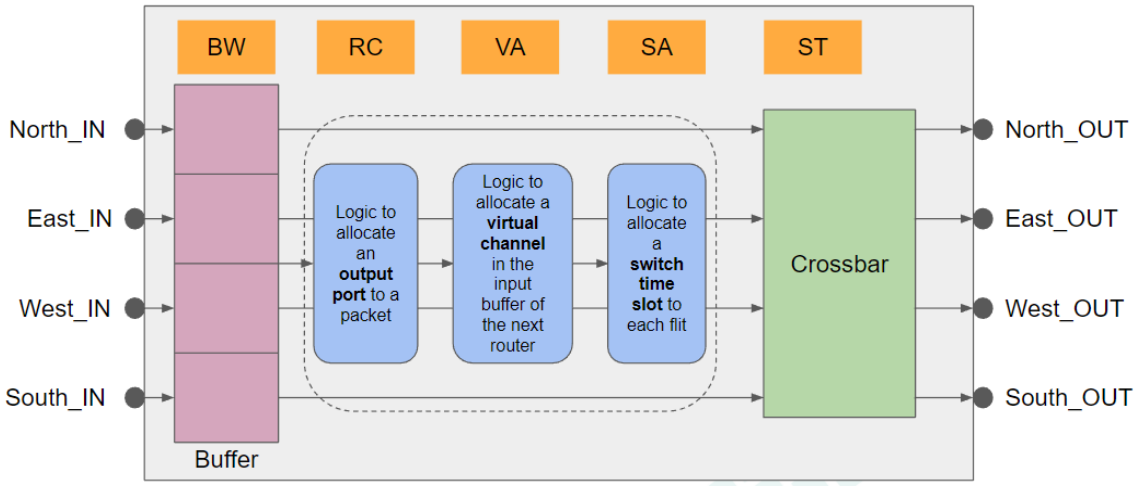

Figure 1.1: Delay variations across different logic implementations.

effect of process variations on the individual delays of different circuits used in an NoC router is studied. These variations are then incorporated into the full NoC simulation setup and performance is evaluated. While the study of delay variations is performed at circuit level, the impact of these variations on NoC performance is studied at flit level. Improvements in the *router microarchitecture* are suggested to mitigate the effect of PV in different ways. A similar approach is taken in Musavvir *et al.* (2020) and improvements in inter-router stages and links have been suggested.

Another approach is to use data regarding variations on different paths in the network in an adaptive routing algorithm. The routing algorithm suggested by Muhammad *et al.* (2019) and Ezz-Eldin *et al.* (2016) uses data related to process variations as input. Because PV data is calculated in real time, this approach makes the NoC process variation-aware.

In these approaches, to mitigate the effect or PVs, modifications are suggested at the architecture level, i.e., in topology, routing scheme, etc., or at the microarchitecture level, i.e., in the type of circuit used. The circuit-level implementation level is usually not discussed.

1.3 Problem Statement

While the approaches discussed in the previous section account for PV effects on NoC performance, not all of them study the variation in impact across different traffic types. Furthermore, none of these works simulate the router circuits using different logic families. This is indeed commonly considered unnecessary because circuit level implementation requires much more time and effort compared to cycle-accurate simulators which do the job with reasonable accuracy at flit level. As PV effects become more prominent with technology scaling, the accuracy required at the design stage during simulation becomes important. Moreover, calculating process variations on the entire circuit instead of its sub-parts would provide more reliable PV values. The expected variation in performance metrics thus calculated would provide as accurate a margin as possible, which is critical in choosing the kind of applications any NoC can give guaranteed performance for. The focus of this thesis is on the following:

- Implementing the circuits used in a router using different digital logic families.
- Studying the change in router delays with logic and finding the minimum-delay router design.
- Implementing a full NoC circuit and calculating performance metrics along with margins due to PV effects across various traffic types.

1.4 Organization

This thesis begins with a general overview of NoC performance metrics. Chapter 2 discusses the structure, design aspects and traffic expected in an NoC. Impacts of architecture-level design choices are briefly discussed. While such an analysis helps create design specifications, achieving these specifications may not always be feasible due to design constraints, hardware manufacturing imperfections, faults, and security attacks.

Chapter 3 presents the simulation details, results obtained, and a comparison of results obtained for different logic implementations. This is followed by a discussion on the significance of these differences. As a result of this work, opportunities for further exploration and future work have been identified in Chapter 4, followed by concluding remarks.

CHAPTER 2

NETWORK PERFORMANCE METRICS

A communication network, in any implementation, is expected to successfully transport messages between the source and destination nodes in a reasonable amount of time. The time constraints, message size, tolerance to transmission failures, and the need for error detection and correction mechanisms are requirements determined by the needs of the application. For example, if the messages generated by nodes are large in number and the message generation rate is high, then the network communication channels must provide sufficient transmission speeds to avoid congestion. If the message packets are large in size, then the channel should be wide enough to allow full packet transmission in the shortest possible time. If any node or link fails in a network, some node pair connections are bound to be lost. The network may either fail entirely or can be designed to assign alternative paths to affected node pairs, as required.

This chapter discusses metrics to quantify the performance of an interconnection network. The features of the network architecture, such as choices of topology, routing algorithm, flow control scheme, and traffic pattern, affect the performance of the NoC. The performance is also impacted by factors other than architecture-level design choices.

2.1 Throughput

Definition

The throughput is the rate at which the network transfers packets. The rate at which the source node generates the packets is the *offered traffic*, while the rate at which these packets are delivered—the *accepted traffic*—is the throughput. It is measured as the number of message units received at the destination per time unit.

Effect of design constraints

Throughput typically increases with the offered traffic up to saturation. The maximum throughput possible in a network is limited by its bandwidth, which in turn depends on the NoC design.

The degree of a node depends on its topology. Consider a network containing 16 nodes. A 1D network (ring) has nodes of degree 1, each having two input and output channels that connect it to the two neighbouring nodes. A 2D network (a 4-ary 2-cube) has nodes of degree 2, each having four input and output channels to connect with four neighbouring nodes. In other words, the number of nodes any given node is connected to is higher in a network with nodes of higher degree. So, a node of higher degree has more number of input and output channels.

As the number of pins available to each node is limited by the chip pin constraints, the number of pins per channel is lower in a node of higher degree. The packaging technology, i.e., the way these nodes are placed on a circuit board determines the maximum bandwidth—the *bisection bandwidth*—available to all channels running along the board. Although more bandwidth per channel is available if the number of channels is lesser, the channel size is already limited by pin constraints. If the channel width is thus smaller than that allowed by the bisection bandwidth, and the channels are less in number, the bisection bandwidth is not fully utilized. In contrast, a higher degree network will allow more channels of the same width to utilize the bisection bandwidth better.

In general, the throughput of a network can be increased by increasing the degree of nodes in the network.

2.2 Latency

Definition

Latency is the time required to deliver a message. Latency is a combination of two factors: hop latency and serialization latency. Hop latency depends on the the number of hops in the path. Serialization latency depends on packet size and channel width. If packet size is larger than the channel width, it will its transmission is serialized, and

increase in the number of serial transmissions increases serialization latency.

Effect of design constraints

Considering the topology example from the previous section, we recall that the 1D network has wider channels than those in the 2D network. As a node of lower degree has lesser number of channels, because of chip pin constraints, it has more IC pins available per channel than those available to a node of higher degree. Large packets need to be sent serially over channels, so a wider channel has lower associated latency.

A network with lower degree nodes has smaller serialization latency that with higher degree nodes. The average hop count and thus the hop latency is smaller for networks with higher degree nodes. But the impact of hop latency is much lower compared to that of serialization latency Dally and Towles (2004). So, the total latency, in general, is higher in networks with higher degree nodes.

CHAPTER 3

SIMULATION RESULTS

3.1 Gate implementation through various logic schemes

For the current work, we have chosen a 4x4 NoC mesh to test the effect of PV on a router's different micro-architectural blocks. We have tested the multiplexer used in the crossbar (XB) block of the router. The XB is responsible for routing the physical data packets (often split across flits) from correct input port to the required output port. The decision of choice of correct input and output ports is taken by the Switch-Allocator (SA) which sets the select lines of XB at correct clock cycle for the data to be transmitted from one directional port to the other.

So as to choose the right implementation which is robust against the process variations while achieving desired timing constraints for a given NoC based application, different logic styles need to be studied statistically. Gaussian distribution variation of device parameters indicates the behaviour of the chip lying in a process lot through simulation. Therefore, the comparison of various implementations (logic styles) of the same micro-architectural block allows us to choose the most suitable implementation. In this work, the XB has been implemented in different digital design styles: Complementary-MOS (CMOS), Pass Gate Logic (PTL) and Transmission Gate Logic (TGL). The SVT-Low-Power 65nm model library by STMicroelectronics is used for the circuit design on Cadence Virtuoso and simulations have been carried out on Eldo (by MentorGraphics). Chosen PVT for current analysis is TT, 1.2V, 25 C.

3.1.1 Crossbar: Multiplexer

The baseline Cross-Bar (XB) has been implemented using five 5x1 Multiplexers (MUXes). The five inputs to each MUX correspond to North, South, East, West and Local inputs and five outputs for the same directional outputs. All of these input-output ports are 32-bits long. Three select lines S0, S1 and S2 (common to all the MUXes) decide the

direction an input Packet must take in order to go to correct next router node of NoC. These select lines hold the value as an output from SA stage. Following sub-section highlight various MUX implementations:

CMOS Logic

A combination of Inverters, 4-input NAND, 5-input NAND gates have been used to implement a conventional CMOS 5x1 MUX. The transistors have been sized following the ratioed sizing of CMOS circuits to obtain delays like a unit inverter. A total of 56 Transistors (28 each of NMOS and PMOS) are required to implement 5x1 MUX in this manner. This, although saves time, transfers strong zero and one and, isolates input from output but at the cost of huge area due to huge transistor count. As the NoC based systems are increasingly becoming the preferred choice for large-scaled heterogeneous MPSoCs and complex systems, increase in area becomes exponential. Hence, such a large number of transistor count makes the XB a very costly element of a NoC Router. Also, the variability due to so many devices is much larger than other implementations as shown in the following section. Furthermore, the static power dissipation is compulsorily non-zero which becomes a more critical aspect at lower technologies and large NoC designs with many routers having more ports owing to torus-type topologies.

Pass Gate Logic (PMOS)

PGL PMOS only deploys half transmission gates by removing NMOS from them. These circuits consist of a PMOS whose gate is controlled by a signal (in this case, a select line) which when goes high, blocks the input applied at the PMOS source from the drain. When the signal on gate goes to zero, source is able to transfer a strong one and a weak zero to the drain. The transistor count for current 5x1 MUX is 14 which is less than one-third of CMOS implementation. Major benefit is that it reduces the area requirement by half and more. There is no static power dissipation as the gate is not permanently attached to the power supply unlike CMOS. Disadvantages are, that the parasitic capacitance effects are more pronounced since the input is given at source diffusion and this type also allows the back-gate driving. Weak zero is another challenge which can be solved by a simple helper NMOS connected at output to support only the

cases when zero arrives or by using sense-amplifier type comparator at the end, still saving more than half area.

Transmission Gate Logic

The transmission gate is similar to PTL, only difference being the presence of NMOS connected in parallel to every PMOS thereby increasing the required area approximately by two times (22 transistors) which is still less than half required by CMOS implementation. All the features are same as PTL apart from the fact that this can transmit a strong zero as well which justifies the area increase. 5x1 MUX implemented using TGL is also analyzed.

3.1.2 Path Delay Measurement

The rise and fall delays (50%-50%) of the XB are observed in the experiments for the setup explained earlier. These results are the averaged values of delays observed when all the combinations of select lines are inserted and the rise-fall transitions at output ports are observed with respect to input ports. The CMOS implementation has the nominal input to output delay of 67.0065 ps. The PGP implementation has the nominal input to output delay of 40.91 ps. The TGL implementation has the input to output delay of 61.825 ps. We can observe that CMOS implementation has the highest delay time which indicates that CMOS based MUX is not suited for high frequency applications for XB in NoC. Although the PMOS PGP gives best performance, TGL can still be preferred so as to obtain strong zero as well.

3.1.3 Variation Measurement

500 Monte-Carlo simulations were performed on each of the three designs of XB . The standard deviation of CMOS implementation is around 1 ps, followed by 0.6 ps for TGL and 0.5 ps for PGL implementations. This clearly indicates that delay performance is more varied in CMOS. The plots shown in Fig. 3.1 also show this behavior as the distribution of delays with respect to nominal value is most flattened for the CMOS as compared to the other two. This means that across 500 chips manufactured, perfor-

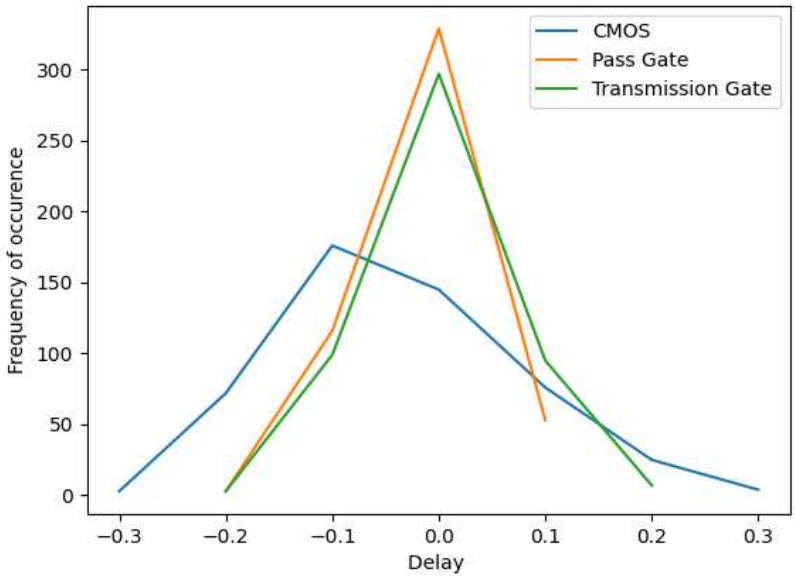

Figure 3.1: Delay variations across different logic implementations.

mance for Crossbars of NoCs is bound to be more varied in CMOS type XB and hence more unpredictable as compared to PGP or TGL which have a controlled behavior.

CHAPTER 4

CONCLUSIONS

In this work, circuit-level analysis of implementation of router logic blocks using different digital logic families has been done. Nominal delay values indicate which implementation is faster, while variation measurements indicate which is more robust. This approach can be used to evaluate performance of different logic blocks separately, and including these circuits in full NoC simulation can help decide which implementation provides the most optimized performance.

4.1 Limitations and Future Work

Logic implementations that use only either NMOS or PMOS devices do not pass one of the logic levels completely. This can cause data corruption. A study on the effect of logic implementation on data passing through the circuit can be done to analyze the extent of the risk of data corruption.

Full NoC simulation is required to evaluate the NoC performance metrics such as throughput, latency, and fault tolerance. The effect of circuit-level logic implementations can then be studied on the overall NoC performance metrics.

REFERENCES

1. **Dally, W. J.** and **B. P. Towles**, *Principles and Practices of Interconnection Networks*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004. ISBN 978-0-08-049780-8.

2. **Ezz-Eldin, R.**, **M. A. El-Moursy**, and **H. F. A. Hamed** (2016). Process Variation Delay and Congestion Aware Routing Algorithm for Asynchronous NoC Design. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, **24**(3), 909–919. ISSN 1557-9999. Conference Name: IEEE Transactions on Very Large Scale Integration (VLSI) Systems.

3. **Muhammad, S. T.**, **M. Saad**, **A. A. El-Moursy**, **M. A. El-Moursy**, and **H. F. Hamed** (2019). CFPA: Congestion aware, fault tolerant and process variation aware adaptive routing algorithm for asynchronous Networks-on-Chip. *Journal of Parallel and Distributed Computing*, **128**, 151–166. ISSN 07437315. URL `https://linkinghub.elsevier.com/retrieve/pii/S0743731518303794`.

4. **Musavvir, S.**, **A. Chatterjee**, **R. G. Kim**, **D. H. Kim**, and **P. P. Pande** (2020). Inter-Tier Process-Variation-Aware Monolithic 3-D NoC Design Space Exploration. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, **28**(3), 686–699. ISSN 1063-8210, 1557-9999. URL `https://ieeexplore.ieee.org/document/8936528/`.

5. **Nicopoulos, C.**, **S. Srinivasan**, **A. Yanamandra**, **D. Park**, **V. Narayanan**, **C. R. Das**, and **M. J. Irwin** (2010). On the Effects of Process Variation in Network-on-Chip Architectures. *IEEE Transactions on Dependable and Secure Computing*, **7**(3), 240–254. ISSN 1941-0018. Conference Name: IEEE Transactions on Dependable and Secure Computing.