



# ON ENHANCING THE PERFORMANCE OF BUFFERLESS NETWORK-ON-CHIP

By

Mohamed Assem Abd ElMohsen Ibrahim

A Thesis Submitted to the  
Faculty of Engineering at Cairo University  
in Partial Fulfillment of the  
Requirements for the Degree of  
MASTER OF SCIENCE  
in  
Computer Engineering

FACULTY OF ENGINEERING, CAIRO UNIVERSITY  
GIZA, EGYPT  
2016

# ON ENHANCING THE PERFORMANCE OF BUFFERLESS NETWORK-ON-CHIP

By  
Mohamed Assem Abd ElMohsen Ibrahim

A Thesis Submitted to the  
Faculty of Engineering at Cairo University  
in Partial Fulfillment of the  
Requirements for the Degree of  
MASTER OF SCIENCE  
in  
Computer Engineering

Under the Supervision of  
Dr. Hatem M. El-Boghdadi

Professor  
Computer Engineering Department  
Faculty of Engineering, Cairo University

FACULTY OF ENGINEERING, CAIRO UNIVERSITY  
GIZA, EGYPT  
2016

# ON ENHANCING THE PERFORMANCE OF BUFFERLESS NETWORK-ON-CHIP

By  
Mohamed Assem Abd ElMohsen Ibrahim

A Thesis Submitted to the  
Faculty of Engineering at Cairo University  
in Partial Fulfillment of the  
Requirements for the Degree of  
MASTER OF SCIENCE  
in  
Computer Engineering

Approved by the  
Examining Committee

---

Dr. Hatem M. El-Boghdadi, Thesis Main Advisor  
- Professor at the Faculty of Engineering, Cairo University

---

Dr. Amr G. Wassal, Internal Examiner  
- Associate Professor at the Faculty of Engineering, Cairo University

---

Prof. Dr. Mohammad Z. Abdel Majeed, External Examiner  
- Professor at the Faculty of Engineering, Al-Azhar University

FACULTY OF ENGINEERING, CAIRO UNIVERSITY  
GIZA, EGYPT  
2016

**Engineer's Name:** Mohamed Assem Abd ElMohsen Ibrahim  
**Date of Birth:** 07/07/1988  
**Nationality:** Egyptian  
**E-mail:** [mohamedassem@eng.cu.edu.eg](mailto:mohamedassem@eng.cu.edu.eg)  
**Phone:** +2 01007846437  
**Address:** 407N, Pyramids Gardens  
Giza, Egypt  
**Registration Date:** 01/10/2010  
**Awarding Date:** ....../...../2016  
**Degree:** Master of Science  
**Department:** Computer Engineering



**Supervisors:** Dr. Hatem Mahmoud El-Boghdadi

**Examiners:** Prof. Dr. Mohammad Zaki Abdel-Majeed (External examiner)  
- Professor at the Faculty of Engineering, Al-Azhar University  
Dr. Amr Jalal El-Deen Wassal (Internal examiner)  
Dr. Hatem Mahmoud El-Boghdadi (Thesis main advisor)

**Title of Thesis:**  
On Enhancing the Performance of Bufferless Network-on-Chip

**Key Words:**  
Bufferless Network-on-Chip; Selection Function; Maximum Flexibility; Ranking Policies; Congestion Management;

### **Summary:**

With the arrival of chip multiprocessor systems, Network-on-Chip (NoC) has started to form the backbone of communication within a microprocessor chip. However, unfortunately, the performance of NoC is bounded by the limited power and area budgets. Bufferless NoC has emerged as a solution to reduce power and area. Bufferless NoC eliminates the buffers used for routing and/or flow control and handle contention using packet dropping or packet deflection. In this thesis, we focus on enhancing the performance (latency and deflection count) of deflection-based bufferless NoC running latency-sensitive applications.

First, we present an analytical study for the traffic in bufferless NoC under the Maximum Flexibility (MaxFlex) selection function with different step sizes. We also provide an experimental study under MaxFlex. Simulation results show that with large values of step size, the latency could be reduced by 97% over using Straight Line selection function. The proposed analysis explains the outperforming experimental results.

Then we propose different flit ranking policies that focus on decreasing the deflection count of the flits. Simulation results show that the proposed ranking policies can reduce the latency by up to 58% compared to Oldest First policy.

Finally, we consider relaxing the effect of congestion in bufferless NoC under high injection rate. We propose two approaches for congestion prevention. The first considers running applications on NoC with extra nodes. The second considers dividing a certain load into a sequence of lighter loads. Simulation results show that the proposed approaches enhance the latency by up to 61% in addition to operating at higher injections rates.

## Acknowledgments

بسم الله الرحمن الرحيم

"سُبْحَانَكَ لَا عِلْمَ لَنَا إِلَّا مَا عَلَّمْتَنَا ۖ إِنَّكَ أَنْتَ الْعَلِيمُ الْحَكِيمُ (٣٢)" البقرة.

"Glory to You (O Lord), we have no knowledge except what you have taught us. Indeed, it is You who is the knowing, the wise (32)" Al-baqarah

"وَمَا تَوْفِيقِي إِلَّا بِاللَّهِ ۖ عَلَيْهِ تَوَكَّلْتُ وَإِلَيْهِ أَنِيبُ (٨٨)" هود.

"And my success is not but through Allah. Upon him I have relied and to Him I return (88)" Hood.

I would like to express my sincere gratitude to my advisor, Dr. Hatem El-Boghdadi, for his huge support, patience and immense knowledge. His guidance helped me in all the time of working on this thesis. I could not have imagined having a better advisor and mentor for my masters study.

Also, I take this opportunity to express gratitude to all of the Computer Engineering department members for their help and support.

I would like to thank my family for their encouragement, support, and attention without which I would never have made it to the end.

Last but not the least; I would like to thank my wife, Yousra, for being always there for me and for her support and kindness. My uttermost gratitude goes to Allah that I met her in such critical point in my life.

# Table of Contents

<b>ACKNOWLEDGMENTS.....</b>	<b>I</b>
<b>TABLE OF CONTENTS.....</b>	<b>II</b>
<b>LIST OF TABLES.....</b>	<b>V</b>
<b>LIST OF FIGURES.....</b>	<b>VI</b>
<b>ABSTRACT .....</b>	<b>X</b>
<b>CHAPTER 1 : INTRODUCTION .....</b>	<b>1</b>
1.1.        BASIC BACKGROUND .....	2
1.1.1.        Buffered NoCs .....	2
1.1.2.        Bufferless NoCs .....	3
1.1.3.        Selection Functions .....	3
1.1.4.        Maximum Flexibility Selection Function.....	3
1.1.5.        Flit Ranking Policies .....	4
1.1.6.        Congestion Management.....	4
1.2.        RELATED WORK .....	5
1.3.        SCOPE OF THE THESIS .....	6
1.3.1.        Increasing and Varying Step Size Under MaxFlex .....	7
1.3.2.        Evaluating Flit Ranking Policies.....	7
1.3.3.        Preventing the Congestion .....	8
1.4.        CONTRIBUTION OF THE THESIS .....	8
1.5.        ORGANIZATION OF THE THESIS.....	9
<b>CHAPTER 2 : BACKGROUND.....</b>	<b>10</b>
2.1.        INTERCONNECTION NETWORK .....	10
2.2.        NETWORK-ON-CHIP (NoC).....	11
2.3.        BUFFERLESS NETWORK-ON-CHIP .....	12
2.4.        SELECTION FUNCTIONS .....	13
2.5.        FLIT RANKING POLICIES .....	14
2.6.        CONGESTION MANAGEMENT .....	15
<b>CHAPTER 3 : MODIFIED MAXIMUM FLEXIBILITY SELECTION FUNCTION.....</b>	<b>16</b>
3.1.        PROPOSED APPROACH.....	16
3.2.        ANALYSIS OF MMAXFLEX SELECTION FUNCTION .....	17
3.2.1.        Type 1 Packets .....	19
3.2.2.        Type 2 Packets .....	20
3.2.3.        Type 3 Packets .....	20
3.2.4.        Type 4 Packets .....	21
3.2.5.        Type 5 Packets .....	21
3.2.6.        Type 6 Packets .....	25
3.2.6.1.        Type 6 (a) .....	25

3.2.6.2.	Type 6 (b).....	27
3.2.6.3.	Type 6 (c).....	29
3.2.6.4.	Type 6 (d).....	29
3.2.7.	Type 7 Packets .....	29
3.2.7.1.	Type 7 (a).....	30
3.2.7.2.	Type 7 (b).....	30
3.2.7.3.	Type 7 (c).....	31
3.2.7.4.	Type 7 (d).....	31
3.2.8.	Type 8 Packets .....	31
3.2.8.1.	Type 8 (a).....	35
3.2.8.2.	Type 8 (b).....	36
3.2.9.	Type 9 Packets .....	36
3.2.9.1.	Type 9 (a, c).....	36
3.2.9.2.	Type 9 (b, d).....	37
3.2.10.	Type 10 Packets .....	38
3.2.10.1.	Type 10 (a, c).....	38
3.2.10.2.	Type 10 (b, d).....	39
3.2.11.	Type 11 Packets .....	40
3.2.11.1.	Type 11 (a).....	40
3.2.11.2.	Type 11 (b).....	42
3.2.11.3.	Type 11 (c).....	43
3.2.11.4.	Type 11 (d).....	43
3.2.12.	Type 12 Packets .....	44
3.2.12.1.	Type 12 (a).....	44
3.2.12.2.	Type 12 (b).....	45
3.2.12.3.	Type 12 (c).....	46
3.2.12.4.	Type 12 (d).....	46
3.2.13.	Summary of Packets Count Calculations .....	48
3.3.	PROOF OF PACKET TYPES COMPLETENESS .....	49
3.4.	PACKETS DISTRIBUTION ANALYSIS RESULTS .....	51
3.5.	EXPERIMENTAL SETUP .....	51
3.5.1.	Experimental Methodology.....	52
3.5.2.	Interconnection Network Model .....	52
3.5.3.	Evaluation Metrics .....	52
3.6.	SIMULATION RESULTS .....	53
3.7.	ESTIMATION OF THE VALUE OF THE STEP SIZE .....	55
3.8.	CONCLUDING REMARKS .....	55
<b>CHAPTER 4 : VARIABLE STEP SIZE MAXIMUM FLEXIBILITY SELECTION FUNCTION .....</b>		<b>56</b>
4.1.	MOTIVATION.....	56
4.2.	PROPOSED VARIABLE STEP SIZE APPROACHES .....	56
4.2.1.	Using the Manhattan distance between NoC nodes (NMDVS) .....	57
4.2.2.	Using the Manhattan distance between NoC regions (RMDVS) .....	58
4.2.3.	Using In-Region and Out-Region routing (IORVS) .....	58
4.2.4.	Using the Manhattan distance between NoC nodes for Out-Region routing (ORMDVS).....	59
4.3.	SIMULATION RESULTS .....	59
4.4.	CONCLUDING REMARKS .....	73

<b>CHAPTER 5 : NEW FLIT RANKING POLICIES FOR DEFLECTION-BASED BUFFERLESS NOCS .....</b>	<b>75</b>
5.1.            MOTIVATION.....	75
5.1.1.        Oldest First Ranking Policy (OF).....	75
5.1.2.        Most Deflection First Ranking Policy (MDF) .....	76
5.2.            PROPOSED FLIT RANKING POLICIES .....	76
5.2.1.        Deflection Age Ratio Ranking Policy (DAR).....	76
5.2.2.        Deflection Distance Ratio Ranking Policy (DDR).....	77
5.2.3.        Last Dimension Ranking Policy (LD).....	77
5.3.            SIMULATION RESULTS .....	77
5.4.            CONCLUDING REMARKS .....	79
<b>CHAPTER 6 : TIME-SENSITIVE CONGESTION MANAGEMENT MECHANISMS .....</b>	<b>80</b>
6.1.            MOTIVATION.....	80
6.2.            PROPOSED APPROACHES .....	81
6.2.1.        Using Larger NoCs (LNoC).....	81
6.2.2.        Using Sequential Injection (SI) .....	81
6.3.            SIMULATION RESULTS .....	82
6.4.            CONCLUDING REMARKS .....	85
<b>CHAPTER 7 : DISCUSSION AND CONCLUSION .....</b>	<b>86</b>
7.1.            FUTURE WORK.....	86
<b>REFERENCES .....</b>	<b>88</b>
<b>APPENDIX A: 2D MESH TERMINOLOGIES.....</b>	<b>92</b>
<b>PUBLICATIONS.....</b>	<b>94</b>



## List of Tables

Table 1: Up traffic passing through switch $C$ .....	23
Table 2: Type 5 <i>Count</i> calculation for an increasing diagonal switches under up traffic using different $SS$ values .....	24
Table 3: Down traffic passing through switch $C$ .....	24
Table 4: Type 5 <i>Count</i> calculation for an increasing diagonal switches under down traffic using different $SS$ values.....	24
Table 5: Up traffic passing through switch $C_{Solid}$ .....	27
Table 6: Type 6(a) <i>Count</i> calculation for the solid diagonal switches under up traffic using different $SS$ values .....	27
Table 7: Type 6(a) <i>Count</i> calculation for the dotted diagonal switches under up traffic using different $SS$ values .....	27
Table 8: Down traffic communication passing through switch $D_{Dotted}$ .....	28
Table 9: Type 6(b) <i>Count</i> calculation for the dotted diagonal switches under down traffic using different $SS$ values.....	28
Table 10: Type 6(b) <i>Count</i> calculation for the solid diagonal switches under down traffic using different $SS$ values.....	28
Table 11: Down traffic passing through switch $A$ .....	33
Table 12: Summary for the data collected in Table 11 .....	33
Table 13: Down traffic passing through switch $B$ .....	33
Table 14: Down traffic passing through switch $C$ .....	34
Table 15: Down traffic passing through switch $D$ .....	34
Table 16: Up traffic passing through switch $Z_{Solid}$ .....	42
Table 17: Up traffic passing through switch $Y_{Solid}$ .....	45
Table 18: Formulas for different traffic types .....	47
Table 19: Common variables used in Table 19 .....	49
Table 20: A and B values for up and down traffic .....	49
Table 21: Multiplier value for Type 8, Type 9 and Type 10.....	49
Table 22: Values for Type 9 up traffic communication .....	49
Table 23: First category cases .....	50
Table 24: Second category cases .....	50
Table 25: Step size to mesh dimension percentage .....	55

# List of Figures

Figure 1: Generic switch in a 2D mesh .....	2
Figure 2: Example of interconnection network .....	10
Figure 3: Generic switch in a 2D mesh .....	11
Figure 4: The operation of MaxFlex selection function using step size of one .....	13
Figure 5: The operation of MaxFlex selection function using step size of one .....	17
Figure 6: Increasing and decreasing diagonals in a 2D mesh .....	18
Figure 7: Up and down traffic in 2D mesh .....	18
Figure 8: Location of $W(i,j)$ in 2D mesh row .....	20
Figure 9: Type 3 example for a row in 5x5 and 6x6 meshes .....	21
Figure 10: Type 3 <i>Count</i> calculation for a row switch in a 5x5 mesh.....	21
Figure 11: Location of $W(i,j)$ in 2D mesh diagonal.....	22
Figure 12: Type 5 example for an increasing diagonal .....	23
Figure 13: Type 6 example for an increasing diagonal under both up and down traffics .....	25
Figure 14: Location of $W(i,j)$ in 2D mesh diagonal.....	26
Figure 15: Type 7 example for an increasing diagonal under both up and down traffics .....	30
Figure 16: Type 8 example for an increasing diagonal in a 12x12 mesh.....	32
Figure 17: Procedure for counting the packets passing through a switch for Type 8(a) 35	
Figure 18: Procedure for counting the packets passing through a switch for Type 8(b) 36	
Figure 19: Procedure for counting the packets passing through a switch for Type 9(a, c) .....	37
Figure 20: Procedure for counting the packets passing through a switch for Type 9(b, d) .....	38
Figure 21: Procedure for counting the packets passing through a switch for Type 10(a, c).....	39
Figure 22: Procedure for counting the packets passing through a switch for Type 10(b, d).....	40
Figure 23: Location of $W(i,j)$ in 2D mesh .....	41
Figure 24: Type 11 example for an increasing diagonal under both up and down traffics .....	42
Figure 25: Type 12 example for an increasing diagonal under both up and down traffics .....	45
Figure 26: Number of packet passing through sample border and core switches over different fixed step size values .....	51
Figure 27: Average packet latency for different fixed step size values .....	53
Figure 28: Average deflection count for different fixed step size values.....	53
Figure 29: Average packet latency for different fixed step sizes at flit injection rate = 0.22 flit/cycle/node .....	53
Figure 30: Average packet latency for fixed step size of 8 compared with different selection functions .....	54
Figure 31: Average deflection count for fixed step size of 8 compared with different selection functions .....	54
Figure 32: 4x4 mesh divided into four 2x2 regions .....	57
Figure 33: Average packet latency for NMDVS using different % values.....	60
Figure 34: Average deflection count for NMDVS using different % values .....	60

Figure 35: Average packet latency for RMDVS compared with RMDVS` .....	61
Figure 36: Average deflection count for RMDVS compared with RMDVS` .....	61
Figure 37: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 1 using 2x2 region size.....	62
Figure 38: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 1 using 5x5 region size.....	62
Figure 39: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 1 using 2x2 region size.....	62
Figure 40: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 1 using 5x5 region size.....	62
Figure 41: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 2 using 2x2 region size.....	63
Figure 42: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 2 using 5x5 region size.....	63
Figure 43: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 2 using 2x2 region size.....	63
Figure 44: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 2 using 5x5 region size.....	63
Figure 45: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 3 using 2x2 region size.....	64
Figure 46: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 3 using 5x5 region size.....	64
Figure 47: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 3 using 2x2 region size.....	64
Figure 48: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 3 using 5x5 region size.....	64
Figure 49: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 4 using 2x2 region size.....	65
Figure 50: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 4 using 5x5 region size.....	65
Figure 51: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 4 using 2x2 region size.....	65
Figure 52: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 4 using 5x5 region size.....	65
Figure 53: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 5 using 2x2 region size.....	66
Figure 54: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 5 using 5x5 region size.....	66
Figure 55: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 5 using 2x2 region size.....	66
Figure 56: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 5 using 5x5 region size.....	66
Figure 57: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 6 using 2x2 region size.....	67
Figure 58: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 6 using 5x5 region size.....	67
Figure 59: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 6 using 2x2 region size.....	67

Figure 60: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 6 using 5x5 region size.....	67
Figure 61: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 7 using 2x2 region size.....	68
Figure 62: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 7 using 5x5 region size.....	68
Figure 63: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 7 using 2x2 region size.....	68
Figure 64: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 7 using 5x5 region size.....	68
Figure 65: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 8 using 2x2 region size.....	69
Figure 66: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 8 using 5x5 region size.....	69
Figure 67: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 8 using 2x2 region size.....	69
Figure 68: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 8 using 5x5 region size.....	69
Figure 69: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 9 using 2x2 region size.....	70
Figure 70: Average packet latency for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 9 using 5x5 region size.....	70
Figure 71: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 9 using 2x2 region size.....	70
Figure 72: Average deflection count for different <i>SSOutRegion</i> values under in <i>SSInRegion</i> = 9 using 5x5 region size.....	70
Figure 73: Average packet latency using different <i>SSInRegion</i> values and 60% under 2x2 region size.....	72
Figure 74: Average deflection count using different <i>SSInRegion</i> values and 60% under 2x2 region size.....	72
Figure 75: Average packet latency using different <i>SSInRegion</i> values and 60% under 5x5 region size.....	72
Figure 76: Average deflection count using different <i>SSInRegion</i> values and 60% under 5x5 region size.....	72
Figure 77: Average packet latency for different variable step size formulas.....	73
Figure 78: Average deflection count for different variable step size formulas.....	73
Figure 79: Average packet latency for different ranking policies.....	78
Figure 80: Average deflection count for different ranking policies.....	78
Figure 81: Average packet latency for LD enhancement over other ranking policies...	78
Figure 82: Average deflection count for LD enhancement over other ranking policies	78
Figure 83: Using 4x4 mesh instead of 3x3 mesh .....	81
Figure 84: Example of two phase sequential injection.....	82
Figure 85: Average packet latency for fifteen nodes in different mesh sizes .....	83
Figure 86: Average deflection count for fifteen nodes in different mesh sizes.....	83
Figure 87: Average packet latency for different number of extra nodes in different locations in 10x10 mesh .....	84
Figure 88: Average deflection count for different number of extra nodes in different locations in 10x10 mesh .....	84

Figure 89: Average packet latency for two phase SI using different number of nodes in different locations in 10x10 mesh .....	85
Figure 90: Average deflection count for two phase SI using different number of nodes in different locations in 10x10 mesh .....	85
Figure 91: Main increasing and decreasing diagonals in 5x5 mesh.....	92

# Abstract

Network-on-Chip (NoC) is commonly used to connect different computing components. With the arrival of chip multiprocessor systems, NoC has started to form the backbone of communication between cores and memory within a microprocessor chip. Although NoC has started to form the backbone of communication between cores, the performance of such interconnection network is bounded by the limited power and area budgets. Bufferless NoC has emerged as a solution to reduce power and area. Bufferless NoC eliminates the buffers used for routing or flow control and handle contention using packet dropping or packet deflection.

We focus on enhancing the performance (in particular, packet latency and deflection count) of deflection-based bufferless NoC running latency-sensitive applications. We divide the work to focus on three aspects of NoC. First, we focus on selecting an output port for the outgoing packet. After that, we shift our focus to ranking the flits in order to select which one to serve first. Finally, we investigate relaxing the effect of congestion under high injection rate.

In the first part, we study the effect of Maximum Flexibility selection function (MaxFlex) on 2D bufferless meshes when a fixed or a variable step size is used. The selection function selects an output channel from a set of channels supplied by the routing function. MaxFlex is a well-known selection function that tries to maximize the number of routing choices as a packet approaches its destination. We investigate the distribution of packets through the NoC via increasing and/or varying the used step size as improving the distribution leads to better utilization and thus better performance. Simulation results show that using a larger step size can enhance the performance by up to 95% compared to using Straight Line selection function. Also, the results show that using variable step size enhances the performance compared to fixed step size by up to 29 %.

Concerning the second part, we devise and evaluate different flit ranking policies. A flit ranking policy chooses which flit should be served first, thus it determines which flit can select an output port first. In this work, we propose novel ranking policies that take the deflection behavior of the bufferless NoC into account. Via the experimental study, we compare these policies to the Oldest First (OF) ranking policy. Simulation results show that the performance of the proposed policies excels over fixed step size MaxFlex with OF as ranking policy by up to 58%.

Finally, we focus on congestion prevention for bufferless NoC running latency-sensitive applications. NoC congestion is one of the main roadblocks that prevent the bufferless NoC to operate under high injection rates. Thus, by relaxing the congestion, bufferless NoCs can approach the performance of buffered NoCs but without the extra cost of using buffers (power and area). To address this problem, we propose prevention mechanisms that target the deflection count of the flits. The proposed approaches aim to give more space for the flits to roam leading to fewer deflections which directly affects the overall packet latency. Via simulation, we show that the proposed approaches enhance the packet latency by 61% compared to fixed step size MaxFlex.

# Chapter 1 : Introduction

In the last few years, there is an industry wide switch to many-core and multi-core systems. In such systems, the performance of the communication system is very critical to the performance of the whole system.

Network-on-Chip (NoC) has emerged as a solution for the limitations in the traditional communications approaches (e.g. buses) especially after the tremendous increase in the number of the communicating modules within a single silicon chip [1,2]. NoC is a group of switches connecting homogeneous or heterogeneous nodes in a multiple point-to-point fashion [3,4]. NoC switches forward the data to/from the nodes/switches over links equipped with input and output buffers.

Buffered NoCs became the de facto approach for communication between cores within chip as they are more scalable, reliable, and predictable. Buffered NoCs were shown to consume significant power and chip area. For instance, in the Intel Teraflops chip and the MIT RAW chip, NoC fabric consumes around 30% and 36% power respectively [5,6]. Focusing on a single NoC switch, a considerable fraction of power and area is used by the internal buffers of the switch. In [7,8], the buffers within a single switch consume around 37% power and 80% area. In addition to being heavy power and area consumers, buffered NoCs are more complex to design as they require extra handlers for packets placement and buffer overflow.

Bufferless NoC has emerged as a solution to decrease power and area requirements [9,10,11,12]. Bufferless NoC eliminates the buffers used within switches; which has a direct impact on power and area. In contrast to the traditional buffered NoC; when two packets compete for the same output port, the allocator either drops or deflects (misroute) the losing packet instead of buffering it. Dropped packet should be retransmitted again. On the other hand, deflected packet follows a non-productive port. Due to the hazards accompanying the dropping mechanism such as handling positive (ACK)/negative (NACK) acknowledgement (NACK buffers [9], NACK network [11]), storing the packet within the source node (extra storage), and retransmission (increase the total network load), in this thesis, we adapt the deflection approach.

Even though bufferless NoCs have their advantages regarding area and power consumption, they have their own problems. Eliminating buffers helps in decreasing the chip area and limiting the consumed power, but at the same time, the flits have no place to reside in case of port contention which leads to dropping or deflecting the flits. This dropping/deflecting mechanism results in increasing the NoC traffic volume which in turn consumes link bandwidth.

Both mechanisms under low to medium rates lightly affect the performance (packet latency and deflection count) leading to a performance approaching buffered NoCs. On the other hand, under high injection rates, the number of packets increases leading to more contention, as a result, using bufferless NoCs leads to reducing the total available bandwidth (as a result of increasing the traffic volume due to retransmitting the flits or deflecting the flits away from their destination) which eventually leads to a performance worse than buffered NoCs. Thus, bufferless NoC is shown generally to function efficiently under moderate loads and smaller NoC sizes [10].

In this thesis, we study several aspects of bufferless NoC to serve latency-sensitive applications. In other words, we aim to operate latency-sensitive applications on