Improving Energy Consumption in Networks on Chip using Optimized Algorithms

Mehdi Taassori

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical and Electronic Engineering

Eastern Mediterranean University
February 2016
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Cem Tanova
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Doctor of Philosophy in Electrical and Electronic Engineering.

Prof. Dr. Hasan Demirel
Chair, Department of Electrical and
Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Doctor of Philosophy in Electrical and Electronic Engineering.

Prof. Dr. Şener Uysal
Supervisor

Examining Committee

1. Prof. Dr. Hasan Hüseyin Balık
2. Prof. Dr. Hasan Demirel
3. Prof. Dr. Şener Uysal
4. Prof. Dr. Béla Vizvári
5. Assoc. Prof. Dr. Tonguç Ünlüyurt

ABSTRACT

Network on Chip (NoC) has been suggested as an appropriate and scalable solution for system on chip (SoC) architectures having high communication demands. Power dissipation has become a key factor in the NoCs because of their shrinking sizes. In the first part of the thesis, we propose a new encoding approach aimed at power reduction by decreasing the number of switching activities on the buses. This approach assigns the symbols to data word in such a way that the more frequent words are sent by less power consumption. This algorithm dedicates the symbols with less ones to high probable data and uses transition signaling to transmit data. The proposed method, unlike the existing low power encoding, does not rely on spatial redundancy and keeps the width of the bus constant.

Due to the limitation of the resources in NoC, suitable load distribution over limited resources which is known as mapping optimization problem is a challenging issue. The second part presents an OPtimization technique for Application specifIC NoCs (OPAIC), which aims not only to decrease the energy consumption but also to improve the performance and area of NoCs. Application specific NoCs are preferable since they can be customized to optimize all requirements of the specific applications. OPAIC is composed of two stages to find the optimum NoC; in the first stage, it uses a linearized form of a Quadratic Assignment Problem (QAP) to map tasks on cores to minimize the energy dissipation. In the second stage, due to the colossal effect of router reduction on power consumption of NoC, a Mixed Integer Linear Problem (MILP) is proposed to find the optimum number of the routers for the layout earned in previous stage.

It is also worth mentioning that even though in most of the traditional low power encoding algorithms and optimization techniques the effect of coupling capacitors is ignored, the results show that these capacitors have an increasing contribution in power consumption in the NoCs as the VLSI technology advances and the size of the transistor shrinks. In this dissertation, all evaluation results consider the effect of both self and coupling capacitances in the link power dissipation.

Keywords: Network on Chip, low power encoding, switching activity, power consumption, energy dissipation, latency, application specific, optimization, mapping.

ÖZ

Mikro Çip üzerindeki Ağ (MÇüA), Mikro Çip üzerindeki Sistem (MÇüS) mimarileri için yüksek iletişim taleplerine sahip uygun ve ölçeklenebilir bir çözüm olarak önerilmiştir. Küçülen boyutları yüzünden MÇüA'lardaki güç tüketimi oldukça önemli bir faktör haline gelmiştir. Bu tez çalışmasının ilk bölümünde veri yolları üzerindeki anahtarlama sayılarını azaltarak güc tüketiminin düşürülmesini hedefleyen yeni bir şifreleme yaklaşımı önerilmiştir. Bu yaklaşım, daha sık kelimelerin daha düşük güç tüketilerek gönderileceği şekilde sembolleri veri kelimelerine atamaktadır. Bu algoritma daha düşük bir sayılarına sahip sembolleri yüksek olasılıklı verilere tahsis edip veri gönderimi için geçiş sinyalizasyonunu kullanmaktadır. Önerilen yöntem, mevcut olan düşük güçlü şifreleme yönteminin tersine, mekânsal fazlalığa dayanmamakta ve veri yolu genişliğini korumaktadır.

MÇüA kaynaklarındaki sınırlılık dolaysıyla haritalama optimizasyon problemi olarak bilinen sınırlı kaynaklar üzerindeki uygun yük dağılımı tartışma konusu olmuştur. İkinci bölüm, yalnızca güç tüketiminin düşürülmesini hedef almayıp aynı zamanda MÇüA'ların performansı ve alanını da geliştirmeyi amaçlayan Uygulamaya Özel MÇüA'lar için Optimizasyon Tekniği'ni (UÖMOP) sunmaktadır. Belirli uygulamaların tüm gereksinimlerini iyileştirmek üzere özelleştirilebilme özellikleri Uygulamaya özel MÇüA'ları tercih edilebilir kılmıştır. UÖMOP, ideal MÇüA'yı bulmak için iki aşamadan oluşmuştur. Birinci aşamada, enerji tüketiminin en aza indirgenmesi amacıyla çekirdekler üzerinde görevleri planlamak üzere Kareli Atama Problemi'nin (KAP) lineerleştirilmiş bir şekli kullanılmaktadır. İkinci aşamada ise yönlendirici indirgemesinin MÇüA'nin güç tüketimi üzerindeki muazzam etkisi nedeniyle bir önceki aşamada elde edilen düzen için ideal yönlendirici sayısının bulunması amacıyla bir Karışık Tamsayı Lineer Problemi (KTLP) önerilmiştir.

Geleneksel düşük güçlü şifreleme algoritmaları ve optimizasyon tekniklerinin çoğunda bağlantı kapasitörlerinin etkisi dikkate alınmasa bile sonuçların Çok Büyük Boyutlu Entegrasyon (ÇBBE) Teknolojisinin ilerlemesi ile birlikte bu kapasitörlerin MÇüA'lardaki güç tüketimi konusunda artan bir katkıya sahip olduklarını gösterdiği bahsetmeye değer bulunmaktadır.

Anahtar Kelimeler: Mikro Çip üzerindeki Ağ, düşük güçlü şifreleme, anahtarlama işlemı, güç tüketimi, enerji israfı, gecikme, uygulamaya özel, optimizasyon, haritalama.

Dedicated To My Dear Parents

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS/ABBREVIATIONS

α	Cooling ratio
B_{avg}	Bit Average
BI	Bus Invert
BMP	Bitmap
C	Capacitance
CABI	Crosstalk Avoidance Bus Invert
CDBI	Coupling Driven Bus Invert
Clk	Clock Pulse
CoM	Cost of Mapping
CoR	Cost of Router
CTG	Communication Task Graph
DOCX	Microsoft Word Document in .docx format
DPM	Dynamic Power Management
DVS	Dynamic Voltage Scaling
E(x)	Expected Value
GA	Genetic Algorithm
GIF	Graphics Interchange Format
GR	Generation Rate
HTML	Hyper Text Markup Language
it_{GA}	Iteration of the Genetic Algorithm
ITRS	International Technology Roadmap for Semiconductors
it_{SA}	Iteration of the Simulated Annealing
JPEG	Joint Photographic Experts Group

L	Latency
LWC	Limited Weight Coding
Mb	Megabit
MFLP	Most Frequent Least Power
MILP	Mixed Integer Linear Problem
MOCA	Mesh based On Chip Interconnection Architectures
MPEG	Moving Picture Experts Group
MWD	Multi Window Display
NC	No Coding
NF	North First
NoC	Network on Chip
NP	Nondeterministic Polynomial time
OE	Odd Even
OPAIC	Optimization technique for application specific
P	Power Consumption
$\acute{P}$	Population of the Genetic Algorithm
PDF	Portable Document Format
PNG	Portable Network Graphics
QAP	Quadratic Assignment Problem
S.A	Switching Activity
SA	Simulated Annealing
SoC	System on Chip
TD	Time Duration
T_i	Initial Temperature of the Simulated Annealing
TXT	TeXT

VC	Virtual Channel
VFI	Voltage Frequency Island
VHDL	VHSIC (Very High Speed Integrated Circuit) Hardware Description Language
VLSI	Very Large Scale Integration
VOPD	Video Object Plane Decoder
WAV	Sound files in .wav format
WSM	Weighted Super Mesh

Chapter 1

INTRODUCTION

1.1 Introduction

As technology shrinks, the integration of many Intellectual Property (IP) cores on a chip is increased. Thus, it is indispensible to manage the huge number of IP cores on a chip. Advances in Very Large Scale Integration (VLSI) technology have led researchers to create a system on a chip which is called System on Chip (SoC). The interconnection in SoCs has a colossal effect on the energy consumption and this contribution increases new families of VLSI [1]. However, SoC has some drawbacks, such as lack of scalability, reusability, unpredictable latency and high energy consumption [2]. Network on Chips (NoCs) have been proposed to alleviate today's communication problem of SoCs [2,3]. These advantages in NoC architecture came at the cost of complexity that has a colossal effect on power consumption and performance. Nowadays, power consumption has become a key issue in many core chip architecture. By shrinking the transistors in new VLSI families, the power consumed in cores and logics would not be dominant portion of power any more. Instead, the dominant portion is going toward the interconnection between nodes and logics [1].

1.2 Network on Chip Architecture

Nowadays, the integration of many cores on a single chip becomes technologically possible. VLSI design is moving toward of hundreds of processor and memory elements in System on Chip (SoC) architectures. Interconnection networks are utilized for various applications. Researchers have used an infrastructure to improve these interconnections by borrowing the concept of networking from computer network field which is called Network on Chip (NoC) [3]. Many challenges in SoC can be solved by NoC architecture [3]. Although these days some commercial products are using the NoC infrastructure to enjoy its privileges, there are still many challenges in this kind of network which have significant effect on SoCs [4].

The main advantages of NoC compared to the traditional bus based interconnections are as follows:

- NoCs avoid the crosstalk in ultra deep submicron technologies.
- NoCs has more scalability for segmentation of wires.
- NoCs are reliable, predictable and energy efficient.
- NoCs are performance efficient. Better deal with large bandwidth traffic.
- NoCs provide Globally Asynchronous Locally Synchronous (GALS) paradigm.
- NoCs use wire efficiently when using physical links among the IP cores.
- NoCs increase the degree of freedom in design due to the decentralization.
- NoCs are customizable. The customization allows designer to plan the NoCs to the specific applications.
- NoCs increase modularity. Hence, the procedure of the design is shortened.

Figure 1.1 illustrates a 4x4 mesh NoC. As shown in the Figure, NoC is composed of links, routers and Network Interfaces (NIs). Links are used as a channel to connect the nodes. Routers route the data based on the routing protocol. The routers have some buffers as well. NIs provide connection between the IP cores and the NoC to organize transmission and reception of packets which is segmented into flow control units (flits). NIs are implemented into the IP cores or the routers.

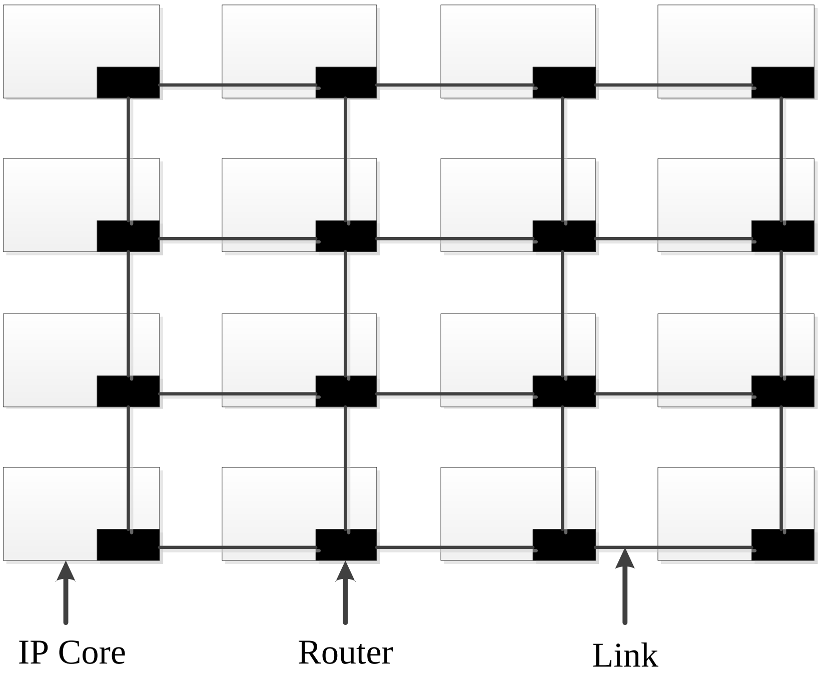


Figure 1.1. An NoC example (4x4)

1.3 Motivation

Power consumption is one of the most important factors in NoC architectures. The power consumption of the NoC consists of power consumed by links and routers [5]. Some of the researchers worked in this area. As technology shrinks, due to the importance of the power reduction, there is a demanding need for improvement of power dissipation. In this dissertation, novel approaches are proposed to reduce the link power consumption and the power dissipation of routers along with the performance improvement.

The motivations behind this dissertation are as follows:

- To present a new low power encoding approach to shrink the power consumption in NoCs.
- Propose analytical methods to obtain the optimum layout for NoCs to reduce the energy dissipation.
- Suggest meta-heuristic approaches and fuzzy logic algorithms to improve the power consumption and the performance in NoCs.

1.4 Thesis Overview

The thesis consists of four chapters which is organized as follows:

Chapter one gives the introduction of the thesis. Chapter two presents a low power encoding approach for on chip networks. It starts with an introduction of power consumption of interconnections in SoCs and low power encoding. The chapter follows by the proposed low power encoding method and its optimality. The chapter then surveys the effectiveness of the presented algorithm. Eventually, the proposed method is evaluated with different characteristics of NoC.

Chapter three presents an optimization technique to improve energy consumption and performance in application specific NoCs. The motivation of using optimization technique in NoCs is discussed in details. The mathematical models are proposed in two main sections. The analytical proposed approaches in this chapter are examined by comparison with the previous work.

Chapter four which is the conclusion and future work suggests fuzzy-based meta-heuristic mapping algorithms for NoCs. It is suggested to utilize meta-heuristic, hybrid meta-heuristic algorithm and fuzzy logic to solve the QAP to obtain the optimum layout for NoCs. The bi-objective fuzzy-based hybrid meta-heuristic algorithm approaches are proposed through this chapter.

Chapter 2

LOW POWER ENCODING FOR ON CHIP NETWORKS

2.1 Introduction

The technological trend in portable and battery-powered devices introduces the power as a new aspect of VLSI design [6,7]. The increased power consumption causes a lot of problems such as decreasing the life time, and increasing the cost of packaging [8]. A great deal of research is conducted to reduce the power consumption of interconnections in SoCs. Decreasing the swing voltage of power supply [9], using dual threshold voltage [10], voltage-frequency island (VFI) [11], activity postponement [12], Dynamic Voltage Scaling (DVS) [13], Dynamic Power Management (DPM) [14], statistical compression [15] and elimination of dispensable buffer slots [16] are some of the power reduction methods presented in the literature.

One of the solutions to decrease the power consumption in chip interconnections is low power encoding [17]. This method tries to decrease the number of switching activities and consequently the dynamic power. On the other hand, the power consumption of coder and decoder are the overhead of this method considered to evaluate its efficiency.

In this chapter, we propose a novel low power encoding approach to decrease the number of switching activities through decreasing the number of ones included in code words and sending the code words with transition signaling. Apparently, in transition signaling, the number of total switching activities is equal to the number of ones in the code words [18]. This thesis introduces a new algorithm to assign code words to symbols in such a way that the more frequent symbols consume less power. To approach this goal, the proposed Most Frequent Least Power (MFLP) encoding uses a tree-based infrastructure. The tree structure provides a set of symbols which assigns the fewer ones words to high probability data and vice versa. Based on the proposed algorithm the most frequent symbols are allocated to the least number of ones which results in the least power consumption.

Most of the low power encoding algorithms increase the width of the transmission bus to send the data [18-21], whereas the proposed method does not rely on spatial redundancy. It is also worth mentioning that even though in most of the traditional low power encoding algorithms the effect of coupling capacitors is ignored, our results show that these capacitors have an increasing contribution in power consumption in the NoCs as the VLSI technology advances and the size of the transistor shrinks. In this thesis, all evaluation results consider capacitors, coupling and self, to calculate the power consumption of links. The experimental results show that by applying the proposed approach, power dissipation up to 46% is improved and with, on an average, 14.4% area overhead.

2.2 Literature Review

Several methods have been proposed to reduce the power consumption by encoding techniques. These include the algorithms that have been designed for data line [19], irredundant encoding [22], correlated data, like address buses [23, 24], parallel and serial which are used for the parallel and serial buses, respectively [25], redundant encoding method that raise either the number of transmission bus or clock pulses to send data [26], and adaptability [27-29].

One of the most well-known low power encoding is the Bus Invert coding [19]. This coding is appropriate for the uniform distribution data and the parallel bus which have spatial redundancy. Another scheme which tries to decrease the number of transitions is limited weight coding (LWC) [18]. In this algorithm, W is defined as a weight of each code word; that is, W is equal to the number of ones included in the code words. LWC applies transition signalling after assigning the code words and can be exploited in both the parallel that have spatial redundancy and serial buses with time redundancy. Beach coding [21] is suggested when the correlation of data pattern is computable. In this approach, the method of encoding is selected based on the pattern of data; therefore, it is strongly application dependent.

Since the power of links in NoCs is an important portion of power consumption, low power encoding also is applicable for this infrastructure. Researchers in [22] present an irredundant encoding and in [17, 30] a set of data encoding methods are proposed to decrease the link power consumption in the NoCs. This is worth mentioning that redundant encoding algorithm cannot decrease the power consumption in NoCs because this redundancy may cause redundancy in each router which is not compensated by power reduction in links. Moreover, due to the fact that in the advanced technology the links are too close to each other the low power encoding used in the NoCs should consider the transition on coupling capacitance as well.

2.3 Proposed Method

The main idea of the proposed method is to reduce the number of ones in code words. In fact, due to the transition signaling, the number of total switching activities is equal to the number of ones in code words [18]. The proposed method is a tree-based algorithm. This tree encompasses root, a number of nodes and leaves. In this tree, code words are represented according to the location of the nodes referring to the data words.

2.3.1 MFLP Encoding Approach

Our approach uses the tree-based structure to assign the code word with less ones to most frequent words; hence, we called it Most Frequent Least Power (MFLP) consumption coding. The objective is to minimize expectation of ‘1’ and in turn, decrease the switching activities as well as the power consumption by using transition signaling. The tree-based structure also allows us to assign the shortest code words to more frequent symbols. Hence, this coding algorithm not only decreases the power consumption but also compresses the amount of transmitted data. Required statistical knowledge about frequencies of symbols are collected in previous time sliding windows; in other word, while data is passing, the frequency of data can be counted and this knowledge can be used to encode the data for next time sliding windows; evidently, the current data is being coded based on the statistical information gathered in previous time sliding window. MFLP coding is collecting this information while data is passing and because the sliding windows is small enough, the characteristic of data is likely be same in consecutive period of data. At first, we need to choose a parameter called division factor. According to this factor, we divide the words into two parts. This factor indicates that we are going either decrease the power or compress data. The tree is made of nodes, where each node has a label indicating the sum of labels of its children. In the case of leaf, this label refers to the frequency of words represented by this node. This tree structure can be created reversely; after dividing the words of data in two portions according to division factor, we assign the sum of these nodes as a label of the root. The root's label represents the sum of label of its children. We continue the procedure until the leaf of the tree which refers to each word of the data. This function is implemented in hardware and inserted in coder and decoder. The pseudocode of the proposed algorithm is presented in Figure 2.1.

```
Given sorted frequencies of symbols as
{$A_i$ $1 \leq i \leq n$ | $\forall i, j, A_i < A_j \rightarrow f_i < f_j$ // symbol $A_i$ has frequency $f_i$
; chosen division factor = $\gamma$
function MFLP-tree ($S = \{ (A_j, f_j), \ldots, (A_k, f_k)\}$) // j and k are first and last index of symbols, respectively
$T_1 \leftarrow \sum_{i=1}^{n} f_i$ // root of tree whose label is sum of all frequencies in S
if ($j = i$)
    insert a node labeled $T_1$
else
{
    divide S into two subsets, $S_1 = \{ (A_j, f_j), \ldots, (A_{\lceil \gamma k \rceil}, f_{\lceil \gamma k \rceil})\}$, $S_2 = \{(A_{\lfloor \gamma k+1 \rfloor}, f_{\lfloor \gamma k+1 \rfloor}), \ldots, (A_k, f_k)\}$ // two sub sets are generated to create children of root
    MFLP-tree ($S_1$);   // function is called recursively for children named $S_1$ and $S_2$
    MFLP-tree ($S_2$)}
end
```

Figure 2.1. Pseudocode of the proposed algorithm

In this algorithm, A_i is the word of the data whose frequency is f_i and S is a set of data words. T_i is the MFLP tree node labeled by the sum of its children's frequency.

With reference to Figure 1, the tree construction can be further explained with the following steps:

- We sort the frequencies of symbols in descending order from higher frequencies to lower ones.
- We choose division factor (γ) according to the goal, either to decrease the power consumption or to compress the amount of data.
- MFLP function constructs the tree reversely. We have to provide the frequency of data words as input of this function. It divides the data words based on γ in two portions as upper and lower groups. Sum of the upper and lower group frequencies is allocated to the left and right nodes, respectively. After that, it invokes itself reversely to construct interior nodes. This algorithm continues till the leaf nodes are generated.
- The labels "0" and "1" are assigned to the edge of upper and lower group, respectively.
- To figure out the code words, we follow the labels of the edges. The code word is the sequence of the edge labels from root to the frequencies of the symbol.

The procedure of encoding in MFLP is composed of two steps: Counting and Coding. While data is passing from encoder the frequency of transmitted symbols can be counted in a time sliding window; this knowledge let encoder generate the tree structure and assign new code words to the symbols. These new codes are going to be used in the next sliding window. It is clear that meanwhile data is coded based on knowledge of previous window (coding), the frequencies of symbols in current window can be counted to be used in the next window (counting). It is obvious that these two steps can take place at the same time. According to the proposed algorithm, data stream should be divided into the sections with same time period namely sliding window. The frequency of data is counted in current window and will be used in the next sliding window to provide the final code words. Due to temporal locality, the frequencies generated in the previous window can be used in the current window. The same procedure is applied to the decoder to figure out the frequency of received data before decoding.

In the following example, we clarify the steps of the algorithm.

- First step: the symbols should be arranged according to their frequency of occurrence in descending order. For instance, there are 13 symbols which are given to be coded. At first we organize them in alphabetical order: A,B,C,D,E,F,G,H,N,P,Q,R,S.
- Second step: This step depends on the division factor. This value should be multiplied by the number of symbols. The selection of the symbols is based on the result of the last multiplication. Top symbols should be located on the left and the others on the right. This strategy is shown in Figure 2.2.

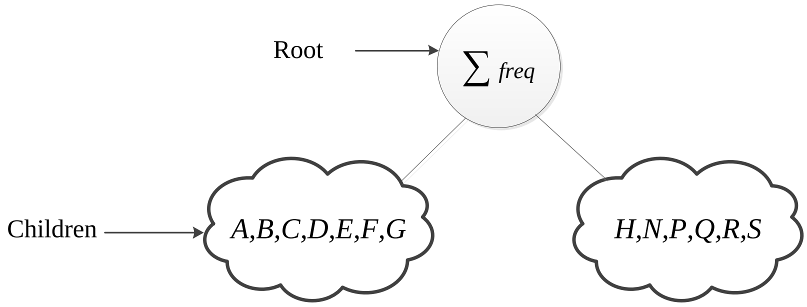


Figure 2.2. Root and its children

It is required to repeat the second step for the symbols which are included in the left hand side. Figure 2.3 shows the steps to reach to the symbols. This trend must be continued for each node either in the left hand side or in the right hand side till we get to one symbol in every set.

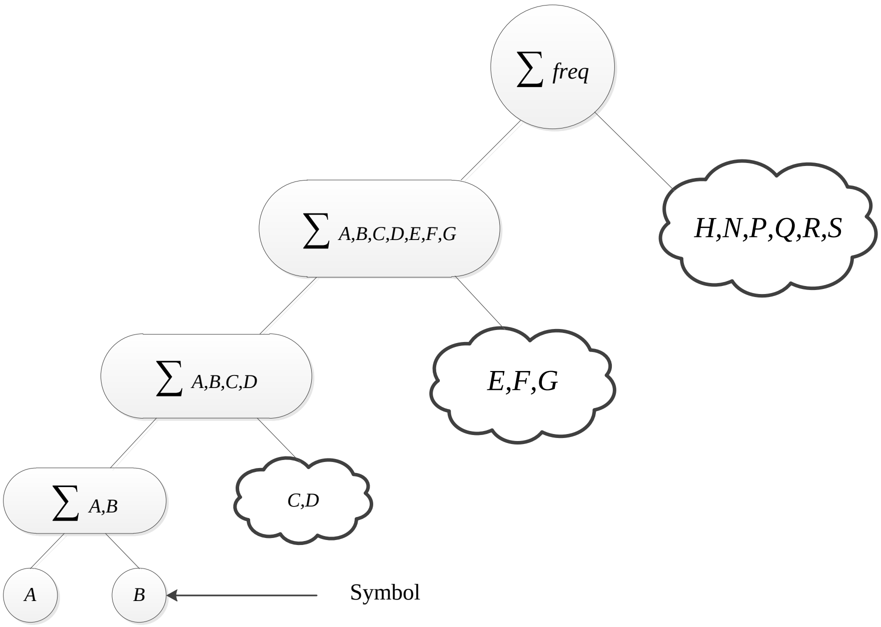


Figure 2.3. Generation of symbols

- Third step: in this step, we assign 0 and 1 to the left and right hand side of the leaves respectively.
- Forth step: in this last step, traversing from the root to the leaves of the tree the code word is found. The result of this example is shown in Table 2.1.

Table 2.1. The code word with different coding

Symbol	Frequency	MFLP	Huffman	3-LWC
A	20	0000	10	0111
B	18	0001	000	1110
C	4	1000	00101	0110
D	4	1001	00110	1000
E	3	1101	001110	0001
F	1	111	001111	0000
G	4	101	01000	0100
H	4	1100	01001	0010
N	6	0101	0101	0011
P	10	0010	011	0101
Q	6	011	00100	1100
R	10	0011	110	1010
S	10	0100	111	1001

In Table 1, the code word generated with MFLP is also compared with the Huffman tree and 3-LWC. We calculate the expectation of ones for symbols by Eq. 2.1.

$$E(x) = \sum_{i=0}^{symbol} F_i * N_i \tag{2.1}$$

where F_i is the frequency of the symbols and N_i is the number of ones for each symbol in the tree.

The tree structure assigns a code word with fewer ones to the more frequent data words. According to Eq. 1, the expectation of ones can be minimized by this strategy. When time duration remains constant, decreasing the power consumption can lead to decrease the energy dissipation. In the case of compression, the energy can be reduced due to decrease in the duration of time provided that either switching activity does not rise or its increment can be compensated by time reduction. Hence, there is a trade-off between the number of switching activities and compression ratio which depends on the division factor. The effect of division factor on compression and power consumption can be evaluated on these bases:

1- As the division factor is increased, we assign the symbols with fewer ones to more frequent data words resulting in less switching activities thereby reducing the power consumption.

2- By reducing the division factor, we can improve the compression ratio. Tree structure allocates small length symbols to more frequent data words at the expense of increasing the number of ones and consequently power dissipation.

To examine how the proposed method reduces the number of switching activities and power consumption, we evaluate the bit average by Eq. 2.2.

$$B_{avg} = \sum_{i=0}^{symbol} F_i * L_i \tag{2.2}$$

where F_i is the frequency of symbol whose length is L_i .

2.3.2 Optimality of MFLP

In this subsection, we present the mathematical proof to show that MFLP, which aims to reduce the power consumption by decreasing the number of ones in the code word, is able to reduce the expectation of ones. Therefore, the MFLP code is optimal if the expected value of ones is minimal. The frequency of symbols are ordered, so that $F_1 \geq F_2 \geq \cdots \geq F_i$. To prove that the $E(x)$ in MFLP code is minimal, we show that with any changes in MFLP's tree and code word the value of expected value is increased. We consider that C_w is an optimal code word which is the result of MFLP encoding. If $F_j \geq F_k$ then $N_k \geq N_j$. We then swap MFLP code words. Supposing that C'_w is the code words j and k of C_w interchanged, the expected value of C'_w is shown in Eq. 2.3.

$$E(C'_w) = \sum_{i=0}^{symbol} F_i * N'_i \tag{2.3}$$

N'_i is the number of ones for symbol after interchanging j^{th} and k^{th} code words.

$$E(C'_w) = \sum_{i=0}^{symbol} F_i * N'_i = F_j * N_k + F_k * N_j$$

$$E(C'_w) - E(C_w) = \sum_{i=0}^{symbol} F_i * N'_i - \sum_{i=0}^{symbol} F_i * N_i$$

$$= (F_j * N_k + F_k * N_j) - \left(F_j * N_j + F_k * N_k\right)$$

$$= \left(F_j - F_k\right)(N_k - N_j)$$

Based on MFLP, if $F_j \geq F_k$ then $N_k \geq N_j$, which means that $E(C'_w) - E(C_w)$ should be greater than zero ($E(C'_w) \geq E(C_w)$). It can be concluded that after changing the code word of MFLP, the value of expected value is increased. Hence, the minimum amount of expected value, the minimum number of ones, is related to MFLP code words and C_w is optimal.

2.4 Effective Criteria in the Efficiency of the Proposed Method

By adding coding algorithm to the system, the power consumption of coder and decoder are considered as overhead and is needed to be compensated. The power consumptions of transmission line without (2.5) and with (2.6) using encoding algorithm are calculated by:

$$P_{link} = P_{self} + P_{coupling} \tag{2.4}$$

$$P_{link} = \propto_s C_{self} V_{dd}^2 f + \propto_c C_{coupling} V_{dd}^2 f \tag{2.5}$$

$$P_{after} = P_{cod} + P_{dec} + \propto_{as} C_{self} V_{dd}^2 f + \propto_{ac} C_{coupling} V_{dd}^2 f \tag{2.6}$$

$$C_{link} = C_{self} + C_{coupling} \tag{2.7}$$

P_{link} is power dissipation before using encoding algorithm and P_{after} is the power after inserting MFLP. P_{link} is composed of power of self capacitance (P_{self}) and coupling capacitance ($P_{coupling}$). As shown in (2.5), $\propto_s$ and $\propto_c$ are switching activity of the self and coupling capacitances, respectively.

P_{after} is power consumption after using encoding approach. P_{cod} and P_{dec} are the power dissipation of coder and decoder, respectively,$\propto_{as}$and $\propto_{ac}$ are switching activity on self and coupling capacitances after applying data coding approach.

$C_{link}$is the total capacitance which is the summation of the self (C_{self}) and coupling ($C_{coupling}$) capacitance, f is the clock frequency and V_{dd} is the power supply of the system.

$\propto_s$ and $\propto_{as}$ which are the self-switching activity before and after using encoding method are evaluated based on the number of transition (high to low and vice versa) on the link. The coupling switching activity before and after using MFLP ($\propto_c$ and $\propto_{ac}$) are calculated according to the direction of switching activities happening on the consecutive wires which is shown in Table 2.2.

The evaluation of self and coupling capacitance is based on the type of the switching activity. In Table 2.2 the number of self and coupling transition for different type of switching activities are depicted.

The coding algorithm can decrease the power consumption, provided that P_{after} is less than the power consumed before applying MFLP. The more the number of switching activities decreased, the more effective our method is. Efficiency factor (β) is introduced in order to evaluate MFLP.

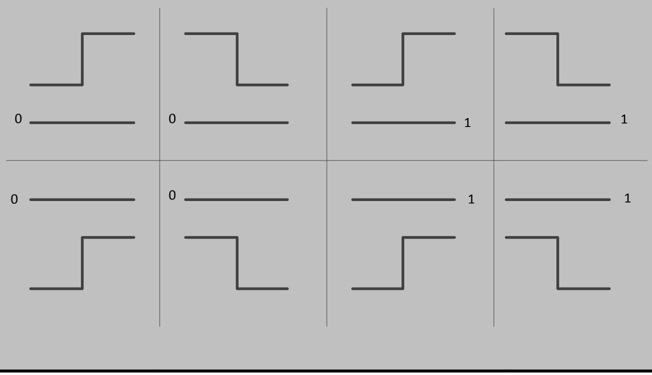
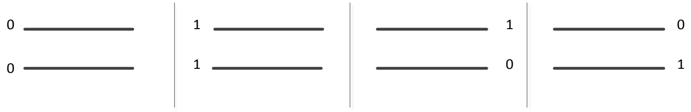
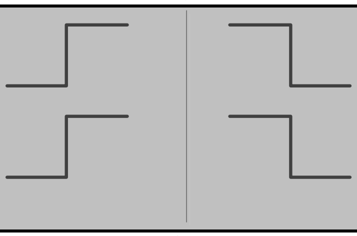
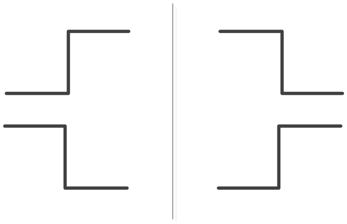
$$P_{after} = P_{codec} + \propto_{as} C_{self} V_{dd}^2 f + \propto_{ac} C_{coupling} V_{dd}^2 f \tag{2.8}$$

where P_{codec} is sum of the power consumption of coder and decoder. As a result, the efficiency factor can be expressed as

$$\beta = \frac{(\propto_s - \propto_{as}) C_{self} V_{dd}^2 f + (\propto_c - \propto_{ac}) C_{coupling} V_{dd}^2 f}{P_{codec}} \tag{2.9}$$

MFLP can reduce the power dissipation if the value of efficiency factor (β) is more than one.

Table 2.2. Number of Self and Coupling capacitances for different type of switching activities

Type	Number of Self Transition	Number of Coupling Transition
	1	1
	0	0
	2	0
	2	4

Assessment of some of the parameters' effectiveness of our approach is presented below:

Distance: One of the most important criterion that affects the efficiency factor is the distance between the transmitter and receiver nodes. Distance has an important role on the amount of capacitance of the link and consequently on the power consumption of the NoC when the switching activity occurs. In other words, by increasing the distance between the transmitter and receiver, the value of the capacitance of links increases. This shows that reduction of the number of transitions on the link plays a more effective role in the improvement of power consumption of the NoC. It is evident that according to Eq. 2.9, the value of the efficiency factor (β) increases due to the increased value of C. Thus, our approach is more effective in longer distances.

Family: With the growth of advanced VLSI technology, the transistors shrink and the length of the wire remains constant or even increases. Eventually, the capacitance of the wire gets more dominant. Therefore, based on Eq. 2.9, the efficiency factor increases and consequently MFLP becomes much more effective.

2.5 Evaluation

The power of the NoC is consumed in two parts, the routers and the links. It should be mentioned that the power of Network Interface (NI) is included in the power of router. In our experiment, the baseline network contains 16 nodes which are connected in a mesh topology whose router algorithm is XY; each router has 2 virtual channels. Packet length is 32 flits. We use power compiler tool from Synopsys[1] to calculate the power of the routers. Power compiler considers the static and dynamic power consumptions. The number of transitions is the major factor indicating dynamic power consumption in data transmission. Despite the fact that the growth of VLSI technology and shrinking the transistor size make the static power dominant part of the power consumption, the research has shown that in the NoC infrastructure the dynamic power still remains the prevalent portion of the power consumption due to its architecture.

[1] Synopsys is registered trademarks

The power of the links is determined by Eq. 2.4. We used 65nm technology for the simulations of the proposed method. According to the International Technology Roadmap for Semiconductors [1], for this technology V_{dd} is defined as 1 Volt and the clock frequency is set to 500 MHz based on the critical path of the system. The length of the metal wires is selected as 2 mm for the mesh topology. The self capacitance of the wire links and coupling capacitance are selected as 0.2 pF/mm and 0.6 pF/mm, respectively. The transitions of wires are calculated by Modelsim[2].

In this section, the coder and decoder are inserted in the local link, between the routers and process elements. In other words, this service is delivered in the transport layer of the NoC which is offered in transmitter and receiver. Hence, the data encoding is done end to end. The coding methods and the NoC infrastructure are implemented in VHDL.

2.5.1 Evaluation of the Proposed Algorithm

It does not matter which infrastructure the designers have chosen, either the traditional bus or the novel NoCs, this coding can be useful for all. To show the effectiveness of our algorithm, we examine its effect in decreasing the power consumption or the amount of data by using some real-life streams. We assess MFLP in the following cases: using buses as a traditional infrastructure and the NoC as a new one.

2.5.1.1 On the Bus

To evaluate our approach we consider a system including a transmitter, a communication bus and a receiver. The power can be calculated in two cases:

[2] Modelsim is registered trademarks

original data and coded version. The power of the link consists of power consumed in the coupling and self capacitances.

On the serial bus, length of the metal wires is assumed as 2 mm and the self capacitance of the wire links is selected as 0.2 pF/mm [1]. It is worth mentioning that on the serial bus we do not have any significant coupling capacitance. The designer needs to decide whether power reduction or decreasing the amount of data is the final goal. According to this decision we need to change the division factor. The more we increase the division factor, the more the bit average goes up. That is, we have gained more power reduction in expense of increasing the amount of data. We evaluate our approach in various division factors for the serial bus using MFLP encoding and the results are shown in Table 2.3 and Figure 2.4.

In the serial system, the energy is calculated by multiplying the power consumption and time duration. It is apparent that the time duration can be estimated by:

$$T = B_{avg} * S * Clk \tag{2.10}$$

Where T is time duration, B_{avg} is bit average, S indicates the number of transmitted symbols, and *Clk* is the period of clock in the transmission system. Consequently, the bit average is able to represent the time duration because other parameters are constant with different division factors.

The energy dissipation before applying encoding algorithm and after using MFLP are evaluated based on the following formula:

$$E_{B.C.} = E_{Router} + E_{Link} \tag{2.11}$$

$$E_{A.C.} = E_{Router} + E_{Codec} + E_{CLink} \tag{2.12}$$

Where $E_{B.C.}$ is energy consumption before using coding method, E_{Router} is energy dissipation of router and NI and E_{Link} is energy which is consumed in the physical links while $E_{A.C.}$ is energy that is consumed after coding which contains E_{Router}, energy consumed in routers, E_{Codec}, enery dissipation in coder and decoder, and E_{CLink} which is consumed in links after using coding algorithm.

The evaluation of different encoding algorithms on various media formats such as text, PDF, color image and so on is reported in [17]. We also assess MFLP and other data encoding approaches on the following data streams which belong to the several media formats like the previous works, namely: TXT: the text file in the .txt format, GIF, JPEG, BMP and PNG: the image files in .gif, .jpg, .bmp and .png format, respectively, WAV: the sound files in the .wav format, HTML: the MHTML Document file in the .mht format, PDF: a PDF format file, and DOCX: Microsoft Word Document in .docx format.

Table 2.3. The bit average and energy consumption with various division factors in serial bus

γ	N.C.		MFLP 10%		MFLP 20%		MFLP 30%		MFLP 40%		MFLP 50%		MFLP 90%	
File name	B.A.	Energy (nj)	B.A.	Energy (nj)	B.A.	Energy (nj)	B.A.	Energy (nj)	B.A.	Energy (nj)	B.A.	Energy (nj)	B.A.	Energy (nj)
.TXT	8	4.23	4.98	5.85	4.90	4.85	4.96	4.05	5.22	3.34	5.70	3.06	13.05	1.95
.GIF	8	891.28	14.77	2229.17	10.35	1419.15	8.71	1034.60	8.17	842.41	7.99	737.65	18.37	516.30
.WAV	8	58.39	15.44	103.51	9.87	73.42	8.53	57.19	7.88	47.04	7.77	41.46	20.57	25.70
.HTML	8	26.70	8.30	53.16	6.65	37.64	6.23	30.30	6.25	25.27	6.53	22.13	15.60	14.16
.JPG	8	72.34	16.51	280.52	11.13	177.80	9.10	129.40	8.35	105.11	7.96	61.05	18.14	37.95
.BMP	8	36.15	14.93	2578.93	10.67	1795.95	8.86	1339.96	8.21	1116.49	7.97	1006.85	17.93	637.40
.PNG	8	3.11	8.98	6.30	6.92	4.26	6.83	3.31	7.22	2.71	7.81	2.42	18.15	1.59
.PDF	8	43.58	16.56	115.39	10.94	73.33	9.15	53.50	8.38	43.56	7.96	38.20	21.49	27.12
.DOCX	8	24.95	14.21	69.85	10.01	44.37	8.42	32.28	8.11	26.26	7.98	22.97	17.19	15.12

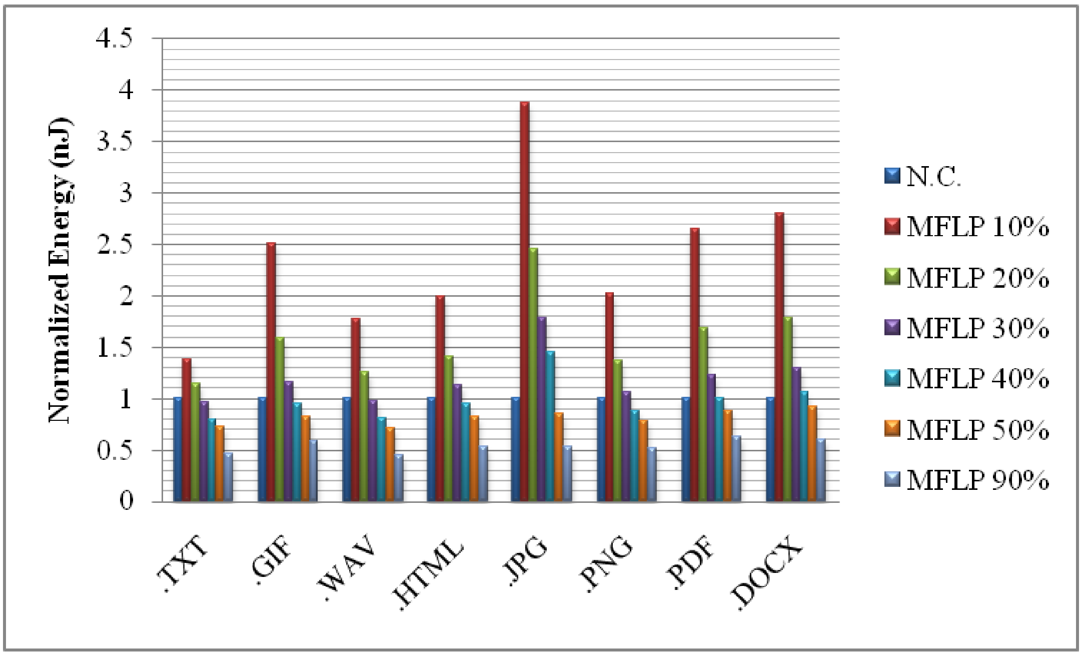


Figure 2.4. Comparison of energy consumption with various division factors on the serial bus

Figure 2.4 is normalized to the energy consumption of No Coding (N.C.) for each benchmark. According to the results given in Table 2.3 and Figure 2.4, it can be deduced that with increasing the percentage of division factor, the energy dissipation decreases, but in the extreme points of 10% and 90%, the value of the bit average is high and is not appropriate. By increasing the percentage from 10% to 20% and up to 50%, the bit average decreases. This trend can be seen from 90% to 50% as well. Hence, the optimum point for an appropriate energy consumption and bit average is 50%, but there is flexibility in encoding to reach a trade-off between energy dissipation and bit average. In introduced encoding algorithm, according to tree based structure used in this coding, to decrease the number of “1s” in each symbol, the length of each symbol increases necessarily and as a result, the bit average goes up. In other words, in this encoding although the total number of “1s” reduces which means the energy consumption of link gets improved, the total length of data increases, meaning the efficiency of compressor decreases and the bit average goes down as well.

Using this conclusion, we use a division factor of 50% for the rest of the implementation. In the transition signaling approach, the number of ones included in the code word is the same as the number of the transition activity [18]. Therefore, decreasing the number of ones induces the switching activity reduction. In our assessment, the switching activity reduction ratio can be defined as follows:

$$S.A. = \frac{S.N_{NC} - S.N_{MFLP}}{S.N_{NC}} * 100 \tag{2.13}$$

where $S.N_{NC}$ is the number of switching activity without encoding algorithm (No Coding) and $S.N_{MFLP}$ is the number of switching activity with applying MFLP approach. The link power dissipation is evaluated based on the number of switching activity.

Table 2.4. The number of switching activities on the parallel bus

S.A.	Link & Coupling		
File name	N.C	MFLP	Improvement (%)
.TXT	37961	27882	26.55
.GIF	5424217	4483953	17.33
.WAV	416717	291368	30.08
.HTML	213646	164755	22.88
.JPG	668279	548911	17.86
.BMP	6117243	4250834	30.51
.PNG	24772	18700	24.51
.PDF	280947	241432	14.06
.DOCX	166708	150560	9.68

Table 2.5. The link and total power consumption on the parallel bus

Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	17.72	13.18	25.64	17.72	13.19	25.55
.GIF	24.74	19.58	20.84	24.74	19.69	20.39
.WAV	20.23	13.99	30.80	20.23	14.06	30.49
.HTML	20.32	15.59	23.28	20.32	15.61	23.15
.JPG	24.83	19.24	22.52	24.83	19.35	22.08
.BMP	25.94	16.77	35.32	25.94	16.88	34.92
.PNG	14.19	10.67	24.79	14.19	10.73	24.37
.PDF	23.72	19.64	17.17	23.72	19.75	16.72
.DOCX	21.00	18.71	10.93	21.00	18.81	10.42

We have examined our approach on the parallel bus as well. In this case, we assume an eight bit bus between the transmitter and receiver whose length is 2mm; similarly, the self and coupling capacitances are considered as 0.2pF/mm and 0.6pF/mm, respectively [1]. The power of the system before coding is represented by the power of the link where the original data is passing; while the power of the encoder and decoder plus the power of links where the coded data is passing can be considered as the total power of the system after coding. The results in Tables 2.4 and 2.5 illustrate that the power of the link after coding decreases so that it can compensate the power of overhead of coding. As depicted in Table 2.5, link power dissipation can be decreased up to 35%. It is obvious that the power consumption of MFLP coder and decoder is the overhead of our design.

2.5.1.2 In the Network on Chip (NoC)

The power consumption is one of the most important factors in the NoCs. Therefore, we assess the proposed method in this infrastructure to decrease the power consumed. The impact of MFLP is assessed on the parallel bus of the NoC. The simulation is carried out based on the specific characteristics which are explained in detail on the experimental results section. Nowadays, the link power dissipation of the NoCs is a significant portion of the total power consumption [30]. As shown in Tables 2.6 by applying MFLP, the number of switching activities can be decreased by up to 45%. In Table 2.7 comparison of power consumption between proposed method and baseline is presented. In the second column of Table 2.7, the link power dissipation in both cases, baseline and MFLP is shown. The router's power consumption before and after using MFLP are demonstrated in third column. As mentioned, the power of coder and decoder as overhead of the proposed approach is presented in forth column. In the last columns the total power consumption for baseline and MFLP are depicted. Table 2.7 shows that after using MFLP, link and total power dissipation can be decreased up to 46% and 16%, respectively.

Table 2.6. The number of switching activities in the NoC

S.A.	Link & Coupling		
File name	N.C	MFLP	Improvement (%)
.TXT	12862708	9446574	26.55
.GIF	14725152	10107890	31.35
.WAV	14285602	7851098	45.04
.HTML	13385518	11025158	17.63
.JPG	15633886	9862332	36.91
.BMP	7513140	6067254	19.24
.PNG	10511526	7399920	29.60
.PDF	15077978	9851948	34.66
.DOCX	13456156	8712344	35.25

Table 2.7. Comparison of power consumption between MFLP and no coding

Power	Link (mW)		Router (mW)		Coder & Decoder (mW)	Total (mW)	
File name	N.C	MFLP	N.C	MFLP	MFLP	N.C	MFLP
.TXT	27.15	19.51	53.42	53.45	0.03	80.57	72.96
.GIF	31.08	20.88	53.85	54.44	0.59	84.93	75.32
.WAV	30.15	16.22	53.87	53.88	0.01	84.02	70.1
.HTML	28.25	22.78	53.51	54.09	0.58	81.76	76.87
.JPG	33.00	20.37	54.12	54.53	0.41	87.12	74.90
.BMP	15.86	12.53	52.51	52.56	0.05	68.37	65.09
.PNG	22.19	15.29	52.12	53.53	1.41	74.31	68.82
.PDF	31.83	20.35	54.06	54.58	0.52	85.89	74.93
.DOCX	28.40	18.00	53.57	54.08	0.51	81.97	72.08

2.5.1.3 Experimental Results

In this subsection, we study the effectiveness of the proposed algorithm. Table 2.8 gives a comparison between the MFLP and the previous state of the art coding approaches. Regarding Table 2.8, it is obvious that LWC [18], BI [19], CDBI [20] and CABI [22] are not able to decrease the power consumption due to the one additional bit which is in the coded data. That is, these algorithms have spatial redundancy to encode data and this redundancy leads to increase the power consumption. Although, both LWC and BI cannot decrease the power consumption, the latter is better because of the simplicity of coder and decoder. That is, the overhead of BI is lower than LWC. The Beach coding [21], one of the well-known adaptive coding approaches, is application dependent and can be an appropriate solution for the application specific systems. In this case, the type of coding can be changed dynamically according to the relationship between the current data and the previous one.

Table 2.8. Power consumption for different coding approaches in the NoC

File name	N.C	BI	LWC	CDBI	CABI	Beach	MFLP
.TXT	80.5	123	143.3	136.0	122.8	134.4	72.96
.GIF	84.9	123.4	137.2	136.3	123	136.3	75.32
.WAV	84.0	124.4	135.8	141.7	124.4	138.5	70.09
.HTML	81.7	123.4	143.7	134.6	122.8	134.8	76.87
.JPG	87.1	124.1	137.9	137.2	123.9	137.0	74.90
.BMP	70.1	106.1	117.8	118.1	106.6	119.2	65.09
.PNG	74.3	115.6	126.2	126.5	115.6	127.6	68.82
.PDF	85.8	123.9	146.6	135.2	123.5	140.3	74.93
.DOCX	81.9	120.1	131.1	132.7	119.8	132.6	72.08

2.5.2 Evaluation of Sensitivity to Network Parameters

We assess the impact of the network parameters such as topology, routing algorithm, number of nodes, packet length and the number of virtual channels on effectiveness of our method. In this assessment, the default routing algorithm and topology is XY and mesh, respectively.

2.5.2.1 Topology

In this subsection, we investigate the effect of different topologies on the efficiency of our method. Two of the most prevalent topologies, mesh and torus are suitable to be implemented on the NoC with 16 nodes (4×4) due to their two dimensional structure. Tables 2.9-2.12 compare the switching activity, link and total power consumption before (No Coding) and after applying MFLP to the Mesh and Torus topology in the NoC.

Table 2.9. The number of switching activities with different topologies

S.A.	Link & Coupling					
Topology	Mesh			Torus		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	12862708	9446574	26.55	11249314	8381182	25.49
.GIF	14725152	10107890	31.35	12640726	9070620	28.24
.WAV	14285602	7851098	45.04	12442554	7058418	43.27
.HTML	13385518	11025158	17.63	11635454	9803266	15.74
.JPG	15633886	9862332	36.91	13458738	8783176	34.73
.BMP	7513140	6067254	19.24	6446056	5439334	15.61
.PNG	10511526	7399920	29.60	8940024	6693468	25.12
.PDF	15077978	9851948	34.66	12935374	8783236	32.09
.DOCX	13456156	8712344	35.25	11498922	7783884	32.30

Table 2.10. The link power consumption with different topologies

Power	Link					
Topology	Mesh			Torus		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	27.15	19.51	28.11	27.10	19.68	27.39
.GIF	31.08	20.88	32.81	30.45	21.30	30.06
.WAV	30.15	16.22	46.20	29.98	16.57	44.71
.HTML	28.25	22.78	19.38	28.03	23.02	17.89
.JPG	33.00	20.37	38.25	32.43	20.62	36.40
.BMP	15.86	12.53	20.95	15.53	12.77	17.76
.PNG	22.19	15.29	31.09	21.54	15.71	27.03
.PDF	31.83	20.35	36.04	31.16	20.62	33.82
.DOCX	28.40	18.00	36.62	27.70	18.27	34.03

Table 2.11. The total power consumption with different topologies

Power	Total					
Topology		Mesh			Torus	
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	80.57	72.96	9.44	81.94	73.22	10.64
.GIF	84.93	75.32	11.31	85.68	75.87	11.44
.WAV	84.03	70.09	16.58	85.28	70.55	17.26
.HTML	81.76	76.87	5.98	82.95	77.25	6.87
.JPG	87.12	74.90	14.02	87.95	75.27	14.41
.BMP	70.17	65.09	7.23	71.19	65.43	8.09
.PNG	74.31	68.82	7.38	74.96	69.36	7.48
.PDF	85.89	74.93	12.76	86.63	75.33	13.04
.DOCX	81.97	72.08	12.07	82.64	72.47	12.30

As shown, the mesh topology is more suitable in the case of link’s power consumption compared to the torus topology. In other words, the impact of our approach in the mesh topology is better. The reason is due to the extra link on each node in torus topology. The extra link looses the consecutiveness in the data. Hence, the power of link is increased. In terms of the total power dissipation, the effect of both topologies is approximately the same.

In Table 2.12, link, router, coder & decoder and total power consumption for Mesh and Torus topologies before applying encoding approach (N.C) and after using MFLP algorithm are presented separately.

Table 2.12. Comparison of power consumption between Mesh and Torus

Power (mW)	Mesh		Torus	
	N.C	MFLP	N.C	MFLP
Link	22.19	15.29	21.54	15.71
Router	52.12	53.53	53.42	53.65
Coder & Decoder	0	1.41	0	0.22
Total	74.31	68.82	74.96	69.36

2.5.2.2 Routing Algorithm

Routing algorithms can be classified into deterministic, partially adaptive and fully adaptive categories. We examine various routing algorithms, namely, XY, OE and Duato to analyze the efficacy of MFLP in power reduction. XY is a deterministic routing algorithm, OE is known as partially adaptive and Duato is fully adaptive routing algorithm. Tables 2.13-2.19 illustrate the efficiency of our method and show the percentage of power reduction with different routing algorithms as compared to the scheme that no data encoding algorithm is used.

To implement the Duato algorithm, we need two virtual channels to prevent deadlock. Thus, we assign two virtual channels for the other algorithms to have a fair compression. According to the results, it can be concluded, from the switching activity point of view, the Duato algorithm in average is the best and OE is the worst one. Similarly, from the link power dissipation perspective, Duato and XY can outweigh OE. With the increased consumption of the network power on the link, our approach is shown to be substantially better. On the other hand, when a routing algorithm is distributed uniformly, the power consumption of the link goes up because more traffic is distributed smoothly, more performance we have and, in turn, more power is consumed. However, the results also show that OE, as a partially adaptive algorithm, cannot distribute traffic more smoothly than XY as a deterministic algorithm. Therefore, for the efficiency of MFLP, Duato and XY outperform the OE algorithm.

Eventually, it can be concluded that using fully adaptive routing algorithm can pass packets more smoothly which leads link power increase. The more power links consume, the more effective of coding algorithm is.

Table 2.13. The number of switching activities with XY routing algorithm

S.A.	Link & Coupling		
Routing	XY		
File name	N.C	MFLP	Improvement (%)
.TXT	12862708	9446574	26.55
.GIF	14725152	10107890	31.35
.WAV	14285602	7851098	45.04
.HTML	13385518	11025158	17.63
.JPG	15633886	9862332	36.91
.BMP	7513140	6067254	19.24
.PNG	10511526	7399920	29.60
.PDF	15077978	9851948	34.66
.DOCX	13456156	8712344	35.25

Table 2.14. The link and total power consumption with XY routing algorithm

Routing	XY					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	27.15	19.51	28.11	80.57	72.96	9.44
.GIF	31.08	20.88	32.81	84.93	75.32	11.31
.WAV	30.15	16.22	46.20	84.03	70.09	16.58
.HTML	28.25	22.78	19.38	81.76	76.87	5.98
.JPG	33.00	20.37	38.25	87.12	74.90	14.02
.BMP	15.86	12.53	20.95	70.17	65.09	7.23
.PNG	22.19	15.29	31.09	74.31	68.82	7.38
.PDF	31.83	20.35	36.04	85.89	74.93	12.76
.DOCX	28.40	18.00	36.62	81.97	72.08	12.07

Table 2.15. The number of switching activities with Duato routing algorithm

S.A.	Link & Coupling		
Routing	Duato		
File name	N.C	MFLP	Improvement (%)
.TXT	13690716	9702068	29.13
.GIF	15496160	10567061	31.80
.WAV	15172374	8124978	46.44
.HTML	14191036	11440338	19.38
.JPG	16438828	10266371	37.54
.BMP	7840438	6086488	22.37
.PNG	10962578	7640550	30.30
.PDF	15851502	10235256	35.43
.DOCX	14048852	9031316	35.71

Table 2.16. The link and total power consumption with Duato routing algorithm

Routing	Duato					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	29.63	20.95	29.30	75.77	73.08	3.54
.GIF	33.54	22.82	31.97	80.07	76.01	5.07
.WAV	32.84	17.54	46.57	79.31	70.09	11.62
.HTML	30.72	24.70	19.57	77.94	77.56	0.49
.JPG	35.58	22.17	37.69	82.26	75.44	8.28
.BMP	16.97	13.14	22.55	64.78	64.29	0.76
.PNG	23.73	16.50	30.47	69.10	68.71	0.55
.PDF	34.31	22.10	35.58	80.95	75.41	6.83
.DOCX	30.41	19.50	35.87	76.71	72.30	5.75

Table 2.17. The number of switching activities with OE routing algorithm

S.A.	Link & Coupling		
Routing	OE		
File name	N.C	MFLP	Improvement (%)
.TXT	14101932	10539777	25.26
.GIF	15773014	11290329	28.41
.WAV	15464638	9071909	41.33
.HTML	14541624	12010685	17.40
.JPG	16584628	11121049	32.94
.BMP	9082774	7432265	18.17
.PNG	11639822	8690051	25.34
.PDF	16038786	10988705	31.48
.DOCX	14369404	9763117	32.05

Table 2.18. The link and total power consumption with OE routing algorithm

Routing	OE					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	18.28	13.35	26.95	69.12	62.59	9.44
.GIF	20.45	14.31	30.04	71.55	64.37	10.04
.WAV	20.05	11.49	42.66	71.16	61.29	13.87
.HTML	18.85	15.22	19.27	69.74	64.90	6.94
.JPG	21.50	14.09	34.46	72.75	64.28	11.65
.BMP	11.77	9.42	20.02	63.14	58.21	7.79
.PNG	15.09	11.01	27.03	65.13	60.45	7.17
.PDF	20.80	13.92	33.03	72.03	64.19	10.88
.DOCX	18.63	12.37	33.59	69.49	62.11	10.61

Table 2.19. Comparison of power consumption between XY, Duato and OE

Power (mW)	XY		Duato		OE	
	N.C	MFLP	N.C	MFLP	N.C	MFLP
Link	22.19	15.29	23.73	16.50	15.09	11.01
Router	52.12	53.53	45.37	52.21	49.44	50.03
Coder & Decoder	0	1.41	0	6.84	0	0.59
Total	74.31	68.82	69.10	68.71	64.53	61.04

In Table 2.19, comparison of link, router and coder &decoder power dissipation between different routing algorithms such as XY, Duato and OE without encoding algorithm (N.C) and after using the proposed method (MFLP) are shown.

2.5.2.3 Number of Nodes

We study our method with different number of nodes. The NoCs are considered with 4, 16 and 64 nodes. We compare the number of transitions, link and the total power for all the networks separately. The results in Tables 2.20-2.26 depict the improvement in switching activities and power reduction with various numbers of nodes, compared to the case that MFLP is not used.

Table 2.20. The number of switching activities with 2*2 network

S.A.	Link & Coupling		
Node No.	2*2		
File name	N.C	MFLP	Improvement (%)
.TXT	1529752	1061680	30.59
.GIF	1761800	1136606	35.48
.WAV	1701722	833056	51.04
.HTML	1598542	1266752	20.75
.JPG	1868384	1081192	42.13
.BMP	864532	674046	22.03
.PNG	1219800	766912	37.12
.PDF	1794472	1092500	39.11
.DOCX	1584746	949274	40.09

Table 2.21. The link and total power consumption with 4 nodes

Node No.	2*2					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	4.38	3.04	30.59	17.73	17.64	0.48
.GIF	5.04	3.25	35.48	18.21	18.17	0.20
.WAV	4.87	2.38	51.04	18.02	17.08	5.20
.HTML	4.58	3.63	20.75	17.65	17.45	1.13
.JPG	5.35	3.09	42.13	18.56	18.03	2.85
.BMP	2.47	1.93	22.03	15.73	15.20	3.36
.PNG	3.49	2.19	37.12	16.90	16.80	0.57
.PDF	5.14	3.13	39.11	18.33	18.06	1.47
.DOCX	4.54	2.72	40.09	17.63	17.49	0.79

Table 2.22. The number of switching activities with 4*4 network

S.A.	Link & Coupling		
Node No.	4*4		
File name	N.C	MFLP	Improvement (%)
.TXT	12862708	9446574	26.55
.GIF	14725152	10107890	31.35
.WAV	14285602	7851098	45.04
.HTML	13385518	11025158	17.63
.JPG	15633886	9862332	36.91
.BMP	7513140	6067254	19.24
.PNG	10511526	7399920	29.60
.PDF	15077978	9851948	34.66
.DOCX	13456156	8712344	35.25

Table 2.23. The link and total power consumption with 16 nodes

Node No.	4*4					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	27.15	19.51	28.11	80.57	72.96	9.44
.GIF	31.08	20.88	32.81	84.93	75.32	11.31
.WAV	30.15	16.22	46.20	84.03	70.09	16.58
.HTML	28.25	22.78	19.38	81.76	76.87	5.98
.JPG	33.00	20.37	38.25	87.12	74.90	14.02
.BMP	15.86	12.53	20.95	70.17	65.09	7.23
.PNG	22.19	15.29	31.09	74.31	68.82	7.38
.PDF	31.83	20.35	36.04	85.89	74.93	12.76
.DOCX	28.40	18.00	36.62	81.97	72.08	12.07

Table 2.24. The number of switching activities with 8*8 network

S.A.	Link & Coupling		
Node No.	8*8		
File name	N.C	MFLP	Improvement (%)
.TXT	120091518	92546764	22.93
.GIF	133557408	99323522	25.63
.WAV	132419488	80039830	39.55
.HTML	123746808	105578336	14.68
.JPG	141200944	97543004	30.91
.BMP	72960376	63708268	12.68
.PNG	98358796	77087598	21.62
.PDF	136323046	96315828	29.34
.DOCX	121393460	86198546	28.99

Table 2.25. The link and total power consumption with 64 nodes

Node No.	8*8					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	124.34	96.43	22.44	337.02	332.33	1.39
.GIF	138.28	103.49	25.16	351.97	345.84	1.74
.WAV	137.10	83.39	39.17	351.35	321.26	8.56
.HTML	128.12	110.01	14.14	349.90	349.84	0.01
.JPG	146.20	101.63	30.48	360.97	344.17	4.65
.BMP	75.54	66.38	12.12	296.84	296.12	0.24
.PNG	101.84	80.32	21.12	318.74	317.56	0.36
.PDF	141.15	100.35	28.89	355.83	342.72	3.68
.DOCX	125.69	89.81	28.54	337.96	329.66	2.45

In Table 2.26, the components of the power consumption with different number of nodes in baseline and MFLP are depicted.

Table 2.26. Comparison of power consumption between 2*2, 4*4 and 8*8

Power (mW)	2*2		4*4		8*8	
	N.C	MFLP	N.C	MFLP	N.C	MFLP
Link	3.49	2.19	22.19	15.29	101.84	80.32
Router	13.40	14.60	52.12	53.53	216.90	237.24
Coder & Decoder	0	1.20	0	1.41	0	20.34
Total	16.90	16.80	74.31	68.82	318.74	317.56

In this case, one criterion is effective; the consecutiveness of the data. It is evident that when the distance between the transmitter and receiver increases the chance of interference among the flits of packet goes up; therefore, the effectiveness of our approach decreases. Based on this remark, with increasing number of nodes in the NoC, the consecutiveness of the data collapses as well as the effectiveness of our approach is diminished and consequently, power dissipation increased.

2.5.2.4 Size of the Packet Length

The proposed method is tested with different size of packet length. The topology and routing in NoC are mesh and XY, respectively. The number of nodes is 16. The results are shown in Tables 2.27-2.33.

Table 2.27. The number of switching activities with packet length of 16

S.A.	Link & Coupling		
Packet Length	16		
File name	N.C	MFLP	Improvement (%)
.TXT	6242730	4770530	23.58
.GIF	6701018	4801120	28.35
.WAV	6789368	4213084	37.94
.HTML	6353832	5207770	18.03
.JPG	6986728	4722718	32.40
.BMP	4794136	3865508	19.37
.PNG	5642822	4011132	28.91
.PDF	6789906	4726544	30.38
.DOCX	6416312	4419614	31.11

Table 2.28. The link and total power consumption with packet length of 16

Packet Length	16					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	26.33	19.39	26.32	80.48	73.68	8.43
.GIF	28.26	19.52	30.92	82.55	74.50	9.75
.WAV	28.63	17.13	40.17	83.04	71.96	13.34
.HTML	26.80	21.17	20.98	80.96	75.86	6.29
.JPG	29.46	19.20	34.83	83.92	74.37	11.38
.BMP	20.22	15.71	22.26	74.71	69.76	6.62
.PNG	23.80	16.31	31.47	77.15	70.73	8.32
.PDF	28.63	19.21	32.89	83.04	74.50	10.28
.DOCX	27.06	17.97	33.59	81.24	72.86	10.31

Table 2.29. The number of switching activities with packet length of 32

S.A.	Link & Coupling		
Packet Length	32		
File name	N.C	MFLP	Improvement (%)
.TXT	12862708	9446574	26.55
.GIF	14725152	10107890	31.35
.WAV	14285602	7851098	45.04
.HTML	13385518	11025158	17.63
.JPG	15633886	9862332	36.91
.BMP	7513140	6067254	19.24
.PNG	10511526	7399920	29.60
.PDF	15077978	9851948	34.66
.DOCX	13456156	8712344	35.25

Table 2.30. The link and total power consumption with packet length of 32

Packet Length	32					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	27.15	19.51	28.11	80.57	72.96	9.44
.GIF	31.08	20.88	32.81	84.93	75.32	11.31
.WAV	30.15	16.22	46.20	84.03	70.09	16.58
.HTML	28.25	22.78	19.38	81.76	76.87	5.98
.JPG	33.00	20.37	38.25	87.12	74.90	14.02
.BMP	15.86	12.53	20.95	70.17	65.09	7.23
.PNG	22.19	15.29	31.09	74.31	68.82	7.38
.PDF	31.83	20.35	36.04	85.89	74.93	12.76
.DOCX	28.40	18.00	36.62	81.97	72.08	12.07

Table 2.31. The number of switching activities with packet length of 64

S.A.	Link & Coupling		
Packet Length	64		
File name	N.C	MFLP	Improvement (%)
.TXT	27096036	19502414	28.02
.GIF	32517428	20569558	36.74
.WAV	30400572	16146124	46.88
.HTML	29692548	23781046	19.90
.JPG	34171814	20151080	41.03
.BMP	14823566	10908690	26.40
.PNG	21524722	14420272	33.00
.PDF	33601474	19821306	41.01
.DOCX	28233958	17346540	38.56

Table 2.32. The link and total power consumption with packet length of 64

Packet Length	64					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	28.50	20.04	29.69	81.42	72.79	10.60
.GIF	34.21	21.13	38.20	87.85	74.93	14.70
.WAV	31.98	16.59	48.12	85.62	69.83	18.44
.HTML	31.23	24.43	21.76	84.37	78.05	7.49
.JPG	35.95	20.70	42.39	89.84	74.62	16.94
.BMP	15.59	11.21	28.11	69.79	62.86	9.92
.PNG	22.64	14.81	34.55	74.08	67.64	8.68
.PDF	35.35	20.36	42.37	89.33	74.31	16.81
.DOCX	29.70	17.82	39.98	82.75	71.19	13.97

Table 2.33 depicts the power dissipation of link, router and overhead of introduced algorithm with various size of packet length.

Table 2.33. Comparison of power consumption between different sizes of packet length

Power (mW)	16		32		64	
	N.C	MFLP	N.C	MFLP	N.C	MFLP
Link	23.80	16.31	22.19	15.29	22.64	14.81
Router	53.35	54.42	52.12	53.53	51.44	52.83
Coder & Decoder	0	1.06	0	1.41	0	1.38
Total	77.15	70.73	74.31	68.82	74.08	67.64

By comparing the above results, it is worth mentioning that by increasing the packet length in the NoC, the effect of MFLP goes up. We have implemented our approach in the transport layer. It means that only the data part of the flits, not header and footer, is coded only in the transmitter and receiver node. Whenever we change the size of the packet, we change the number of data. In contrast, the number of header and footer remains constant. Hence, by increasing the packet size, the data increase and more data are coded. On the other hand, by decreasing the packet size, only the data section goes down and the other parts remain the same as before. In this case, the numbers of the data that are coded are less. Thus, the effect of our contribution is not much as before and the impact of our proposed method decreases.

2.5.2.5 Number of Virtual Channels

The number of virtual channels is effective on the throughput of the interconnection network. The significant portion of the power consumption in routers is consumed in the virtual channels. In this section, we study the effect of our proposed method with different number of virtual channels. Our approach is implemented in the mesh based with XY routing algorithm. The network has 16 nodes and the packet length is 32. The following results shown in Tables 2.34-2.40 which are the comparison of MFLP and no coding approach are obtained by changing the number of virtual channels.

Table 2.34. The number of switching activities with 1 virtual channel

S.A.	Link & Coupling		
VC. No.	1		
File name	N.C	MFLP	Improvement (%)
.TXT	9274734	6605398	28.78
.GIF	11293766	6977492	38.21
.WAV	10272172	5237046	49.01
.HTML	9800298	7740690	21.01
.JPG	11964952	6682744	44.14
.BMP	5741684	4267954	25.66
.PNG	7910460	4650282	41.21
.PDF	11398648	6685348	41.34
.DOCX	10268486	5887648	42.66

Table 2.35. The link and total power consumption with 1 virtual channel

Virtual Channel	1					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	14.94	10.32	30.92	37.42	35.64	4.76
.GIF	18.19	10.90	40.07	41.05	36.74	10.50
.WAV	16.55	8.18	50.55	39.25	33.66	14.23
.HTML	15.79	12.09	23.39	38.35	37.80	1.43
.JPG	19.27	10.44	45.82	42.17	36.31	13.89
.BMP	9.25	6.66	27.90	32.30	31.49	2.51
.PNG	12.74	7.26	42.98	35.03	32.56	7.05
.PDF	18.36	10.44	43.11	41.23	36.33	11.89
.DOCX	16.54	9.20	44.38	39.26	34.84	11.25

Table 2.36. The number of switching activities with 2 virtual channels

S.A.	Link & Coupling		
VC. No.	2		
File name	N.C	MFLP	Improvement (%)
.TXT	12862708	9446574	26.55
.GIF	14725152	10107890	31.35
.WAV	14285602	7851098	45.04
.HTML	13385518	11025158	17.63
.JPG	15633886	9862332	36.91
.BMP	7513140	6067254	19.24
.PNG	10511526	7399920	29.60
.PDF	15077978	9851948	34.66
.DOCX	13456156	8712344	35.25

Table 2.37. The link and total power consumption with 2 virtual channels

Virtual Channel	2					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	27.15	19.51	28.11	80.57	72.96	9.44
.GIF	31.08	20.88	32.81	84.93	75.32	11.31
.WAV	30.15	16.22	46.20	84.03	70.09	16.58
.HTML	28.25	22.78	19.38	81.76	76.87	5.98
.JPG	33.00	20.37	38.25	87.12	74.90	14.02
.BMP	15.86	12.53	20.95	70.17	65.09	7.23
.PNG	22.19	15.29	31.09	74.31	68.82	7.38
.PDF	31.83	20.35	36.04	85.89	74.93	12.76
.DOCX	28.40	18.00	36.62	81.97	72.08	12.07

Table 2.38. The number of switching activities with 3 virtual channels

S.A.	Link & Coupling		
VC. No.	3		
File name	N.C	MFLP	Improvement (%)
.TXT	12323846	9040062	26.64
.GIF	14236506	9716394	31.75
.WAV	13659634	7522608	44.92
.HTML	12875376	10561330	17.97
.JPG	15123398	9475854	37.34
.BMP	7175404	5775686	19.50
.PNG	10065004	7071758	29.73
.PDF	14531122	9422394	35.15
.DOCX	12969172	8348764	35.62

Table 2.39. The link and total power consumption with 3 virtual channels

Virtual Channel	3					
Power	Link			Total		
File name	N.C	MFLP	Imp. (%)	N.C	MFLP	Imp. (%)
.TXT	28.53	19.75	30.76	116.27	112.95	2.85
.GIF	32.96	21.23	35.58	123.79	115.99	6.29
.WAV	31.63	16.44	48.02	120.53	110.24	8.53
.HTML	29.81	23.08	22.58	118.30	117.28	0.86
.JPG	35.02	20.71	40.86	126.14	115.63	8.33
.BMP	16.61	12.62	24.03	105.72	104.13	1.50
.PNG	23.30	15.45	33.68	113.56	108.75	4.22
.PDF	33.65	20.59	38.80	124.37	115.49	7.13
.DOCX	30.03	18.24	39.24	120.45	112.32	6.74

Table 2.40. Comparison of power consumption between different number of virtual channels

Power (mW)	1		2		3	
	N.C	MFLP	N.C	MFLP	N.C	MFLP
Link	12.74	7.26	22.19	15.29	23.30	15.45
Router	22.29	25.29	52.12	53.53	90.25	93.30
Coder & Decoder	0	3.00	0	1.41	0	3.04
Total	35.03	32.56	74.31	68.82	113.56	108.75

Table 2.40 shows the effect of different number of virtual channels on the power consumption of link, router and coder & decoder with the proposed algorithm.

The impact of virtual channels on the effectiveness of coding depends on two criteria. Firstly, how much order of flits in the network will remain constant while passing through network, secondly, utilization of the bus. As shown above, by increasing the number of virtual channels, sequence of data would be more subject to change and, in turn, the impact of our coding decreases. On the other hand, the growth of number of virtual channels leads to have less congested links and consequently, the utilization of bus goes up. The results show that power consumption of links increases and consequently, the influence of the proposed method rises.

2.5.2.6 Link Length

In this subsection, the impact of link length on efficiency of proposed method is studied. As depicted in Figure 2.5, the link power consumption of MFLP encoding and baseline are compared in various link lengths. In Figure 2.5 the vertical axis is link power dissipation and horizontal axis is link length. As shown in Figure 2.5, by increasing the link length, improvement of the proposed method increases as well. The reason of this improvement is because the longer wires have the bigger capacitance and under this circumstance when the coding algorithm decreases the number of switching activities, more power improvement is possible.

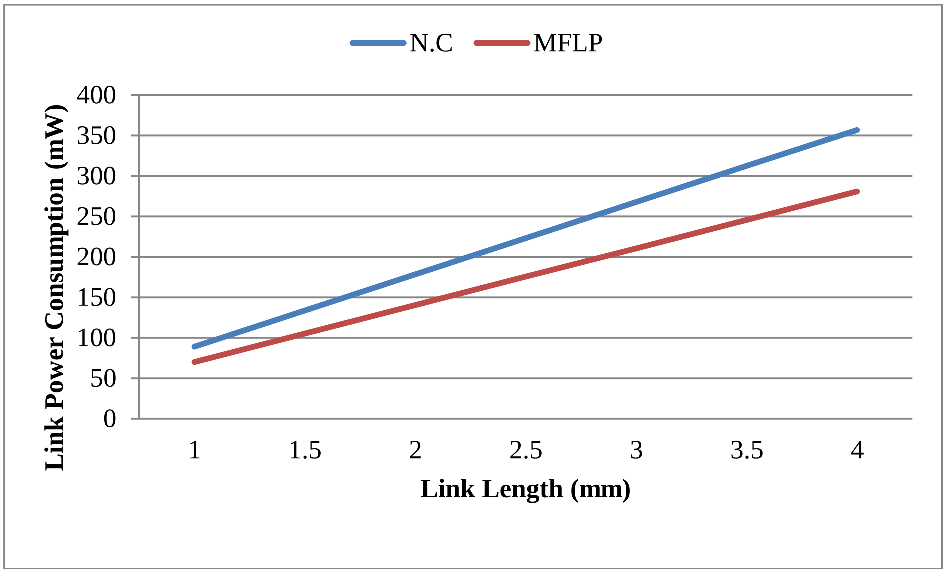


Figure 2.5. The impact of link length on efficiency of MFLP

2.6 Overhead

In this section the overhead of the proposed method on power consumption, critical path and area of routers is considered. The overhead is created by two extra modules, coder and the decoder of MFLP, which are inserted in routers. Entire system including encoding and decoding algorithms is implemented in VHDL and synthesized with Synopsys design compiler in 65 nm technology. According to the ITRS [1], in this technology V_{dd} is defined as 1 Volt and the clock frequency is 500MHz based on the critical path of the system. The topology is mesh with XY routing algorithm and the number of nodes is 16 while the packet length and number of virtual channels are 32 and 2, respectively. The power and area consumption of the coder and decoder are considered as the overhead of power and area which caused by our approach. It is worth mentioning that due to the fact that generating the coding and decoding trees are being done while the packets are transferring, the throughput of system remained unchanged. On the other hand, encoder and decoder can pose power, area and critical path overhead on the routers which are considered in efficiency evaluation of our method. Table 2.41 depicts the power, critical path and area overhead of the proposed method on routers.

Table 2.41. Power, critical path and area overhead of MFLP

Power (mW)			Critical Path (ns)			Area (μ$\boldsymbol{m^2}$)		
N.C.	MFLP	Overhead %	N.C.	MFLP	Overhead %	N.C.	MFLP	Overhead %
52.12	53.53	2.70	1.96	1.97	0.51	36108.32	41679.82	15.43

2.7 Summary

Network on Chip has been proposed as an appropriate solution for today's on-chip communication challenges. Power dissipation has become a key factor in the NoCs because of their shrinking sizes. In this thesis, we propose a new encoding approach aimed at power reduction by decreasing the number of switching activities on the buses. This approach assigns the symbols to data word in such a way that the more frequent words are sent by less power consumption. This algorithm dedicates the symbols with fewer ones to high probability data and uses transition signaling to transmit data. The proposed method, unlike the existing low power encoding, does not rely on spatial redundancy and keeps the width of the bus constant. Experimental evaluations show that our approach reduces the power dissipation up to 46% with 2.70%, 0.51%, and 15.43% power, critical path and area overhead in the NoCs, respectively.

Chapter 3

OPTIMIZATION TECHNIQUE TO IMPROVE ENERGY CONSUMPTION AND PERFORMANCE IN APPLICATION SPECIFIC NETWORKS ON CHIP

3.1 Introduction

SoC architecture can be categorized into regular and irregular topologies. A general NoC usually needs to use regular topology since the designers have to assume that the bandwidth among the different cores is same. Whereas, application specific NoC gives the opportunity to design custom NoCs which are the best choice for our application in terms of power consumption and performance [5,31,32].

Due to the limitation of resources in NoCs, tasks to cores mapping has a colossal effect on the performance, latency and physical link's power dissipation of NoCs. Improvement in link's power consumption and performance can be achieved by assigning the tasks to cores based on the optimum distance and bandwidth [5], [33]. As mentioned, considerable fraction of total power dissipation consumed in the physical links has a direct relation with length of the link and bandwidth of the transmitted data [5]. In order to map tasks to cores, a Quadratic Assignment Problem (QAP) can be defined. The problem assigns the tasks to the cores in a way that the connections with highest bandwidth are assigned to the shortest distances of the cores. The problem generally is categorized as NP-hard problem.

Although, the first priority in nanoscale technologies is energy consumption, this chapter focuses on the optimization technique not only to improve the energy consumption but also to boost the performance of the NoCs. The proposed method can be divided into two stages. The objective of the first stage is to construct an optimized mapping of the tasks onto the core regarding to the bandwidth, link length and latency. Linearization technique is used for Quadratic Assignment Problem to obtain optimal layout of NoCs. One of the most important constraints in mapping algorithms is the link length which is estimated precisely in our assumed topology named weighted super mesh (WSM). In this topology, all the cores are located like the mesh but there is an extra route in diagonal of cubes in comparison with regular mesh that connects every two cores directly to each other. Every link has a weight estimating the distance between two adjacent cores. In other words, traversing between all cores of a cube costs just one hop with different weight. This new topology provides us with more paths between two cores whose distance would be used while mapping the tasks to the cores.

The number of routers has a direct impact on the energy consumption, due to this fact the second contribution tries to obtain the optimum number of routers with a new algorithm.

To evaluate the power dissipation, the existing layout optimization methods consider just power of self-capacitance as a power of physical links. Some research activities [17,22] show that in new VLSI technologies coupling capacitances are as significant as an even more dominant rather than self-capacitance; therefore, in contrast to the existing method [5], we consider both the coupling and self-capacitance of the link to determine the link power consumption. Since the size of transistor is shrinking in every family of VLSI technology, the distance between two adjacent wires in chips keep decreasing significantly and in turn the contribution of coupling capacitance is getting more dominant rather than self-capacitance. As a result, ignoring this kind of capacitance in today’s families of VLSI has a dramatic misleading in the estimation of power consumption in wires on chips.

3.2 Literature Review

In the NoC realm, there are some significant studies on mapping algorithms. Previous works which applied optimization methods to find best mapping for NoCs could be classified in two separate categories; some of them start with a regular topology such as mesh [34-40] and the other group gets started with irregular topology [33]. Hu et al. [34] present the first mapping problem through branch and bound algorithm which maps cores onto a regular NoC with respect to power consumption minimization and the performance of the system is guaranteed through bandwidth reservation. The authors in [35] present a mapping algorithm with communication energy minimization objective such that the performance is guaranteed via bandwidth reservation. They construct a deterministic routing and branch and bound algorithm is used. Murali and De Micheli [36] and Murali et al. [37] introduced different mapping algorithms for mesh based NoCs. Although the proposed method in [36] is fast for mapping, only bandwidth constraint is considered in this method. Murali et al. [37] investigated a mapping and reconfiguration NoC which is able to satisfy communication constraints of different applications. This NoC can be reconfigured based on specifics characteristics of applications. In [38] authors proposed a branch and bound algorithm to map the tasks onto the mesh based NoC architecture and suggested a deadlock free deterministic routing algorithm.

Srinivasan and Chatha [39], introduced a mapping and routing algorithm to decrease the energy consumption of mesh based NoCs. In this work, bandwidth and latency are considered as constraints to solve the problem. This technique gave a mesh topology and a communication task graph as inputs to map the cores onto the routers. [40] presented a discrete particle swarm optimization to map applications onto mesh based Network on Chip architecture.

Obviously, when a mesh topology is considered as the starting point of the algorithm, the distances between cores are defined based on mesh structure; as a result, unintentionally, we are limited by constraints of mesh and we are not able to consider all distances which include an optimum solution. The presented method in this thesis does not fall into this category because weighted super mesh (WSM) helps us to consider all possible routes among different cores; therefore, in the proposed method all the existing cases would be regarded and find the optimum solution. Second category considers an irregular topology to start. Some works show that custom topologies outperform the regular ones. Note that irregular structures are more appropriate for application specific systems. In [33], researchers introduced optimization techniques for application specific NoCs with the objective of power dissipation. The approach is included of two steps; firstly, core to router mapping and secondly, an algorithm to generate a custom topology. Benini [41] describes the challenges and method of design in application specific NoC architecture. According to this research, energy consumption in application specific NoCs can be managed more efficiently than the general purpose applications. A method of core mapping and physical planning along with quality of service has been proposed in [42]. Srinivasan et al. [43] presented a technique to generate the topology of targeted application on chip interconnection architecture. They introduced a method to generate an application specific on-chip interconnection, by using linear programming techniques to map a core to router. In the core to router mapping techniques, topology of NoC is specified before mapping. In spite of this method, OPAIC is not limited by any particular topology in mapping step. Srinivasan et al. [5] suggested a method for application specific NoC architecture by using integer linear programming. The aim of their research is power reduction regarding to the performance. They solve the optimization problem of mapping the core to router and also generate the optimal topology. These researchers show that in terms of the power dissipation and area of NoCs, custom NoC is preferable to regular architectures. Srinivasan et al. [44] investigated an optimal mixed integer linear programming (MILP) regarding to the performance constraints for custom NoC architectures. Chatha et al. [45] presented a method to solve the problem of core to router mapping and generates optimal topology. These researchers divided optimization problem in two sections. Firstly, core to router mapping and secondly, generating the topology and routing for custom NoC architecture. The first goal in this research is decreasing the power consumption and number of routers reduction is the second objective. In contrast with these algorithms, our method is trying to optimize all specifications of NoC regarding the characteristics of application. A tool for finding the best topology for a specific application is explored in [46]. In [46] the authors proposed a tool to map cores onto the some standard topologies and this tool picks the best option among those predefined topologies based on the constraints of a designer whereas our approach is not bound by any specific topology. OPAIC tries to analyze all distances between the adjacent cores regardless of the topology. In the presented super weighted mesh, the cores are located like the mesh but an extra route in diagonal is defined to connect all cores to each other. Extra route enabled us to have a direct connection between each core. Applying this topology provides us with more paths between all the cores in the topology. As a result, the mapping of the tasks to the cores would be based on the accurate distance between the cores which is an important constraint in mapping problem. In [47], a mixed integer linear programming task scheduling and core mapping method for regular and irregular NoC architectures is proposed. The authors presented a graph model to evaluate energy dissipation and latency.

3.3 Motivation

To achieve the optimum NoC, we need to characterize the power consumption and performance of NoCs to figure out which characteristics are effective in contributing to power dissipation and performance. Apparently, Network on Chip consist of two main parts, physical links and routers. Since [48] believes that the length of links and bandwidth (generation rate) are two key parameters in power and performance of physical links, these two parameters are evaluated and their impact on power and performance are studied in part 3.3.1 and 3.3.2. On the other hand, [49] shows that virtual channel is one of the most important source of power consumption in router of NoC. Due to this fact, the effect of number of virtual channels in performance and power of NoCs are also evaluated.

We characterized this NoC in 65 nm technology. According to the International Technology Roadmap for Semiconductors [1], V_{dd} is equal to 1 Volt and the clock frequency is set to 500 MHz based on the critical path of the system. For this topology the length of the metal wires is 2 mm. The capacitance of the wire links and coupling capacitance are selected as 0.2 pF/mm and 0.6 pF/mm respectively. The transitions of wires are calculated by Modelsim. In these simulations, we use a 4*4 mesh topology NoC which has 2 virtual channels per physical channel and using X-Y as routing algorithm. Power of NoC is composed of the power of physical links and routers. Regarding the link power dissipation, we consider the self and coupling capacitances between adjacent wires. We use uniform distribution to send packets between the routers. This study is categorized as follow:

3.3.1 Power Consumption

In this sub section, we study the effect of link length, generation rate, and number of virtual channels in power consumption of NoC.

3.3.1.1 Link Length

Link length has a colossal effect on the link power consumption. The effect of link length on the link power is illustrated in Figure 3.1. In this Figure the generation rate is 0.035 packets/cycles. It is obvious that with increasing in link length there is a linear increment in link power dissipation.

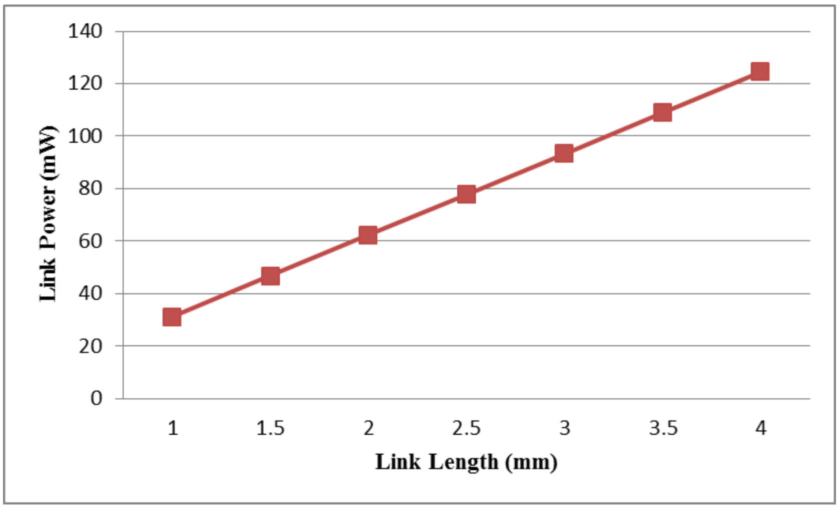


Figure 3.1. The effect of link length on link power consumption

3.3.1.2 Generation Rate

In Figures 3.2 and 3.3 the total and link power consumption versus generation rate for a clock cycle of 14 ns are shown, respectively.

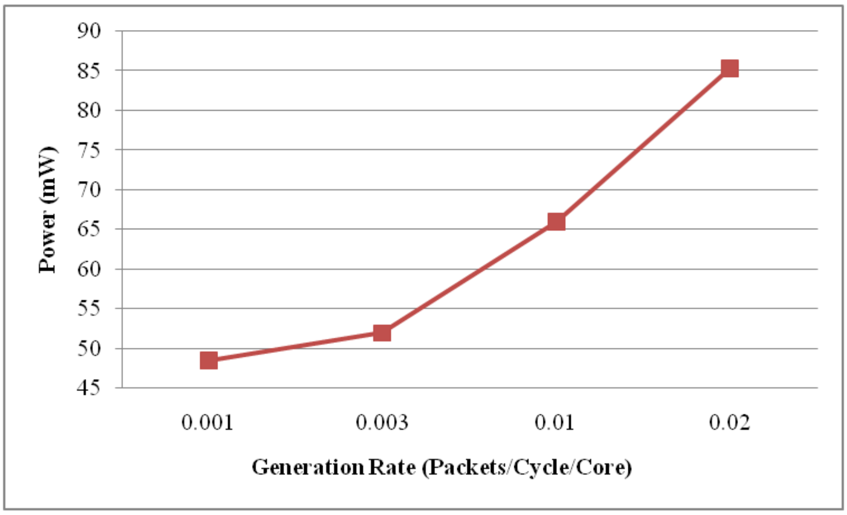


Figure 3.2. The effect of generation rate on power consumption

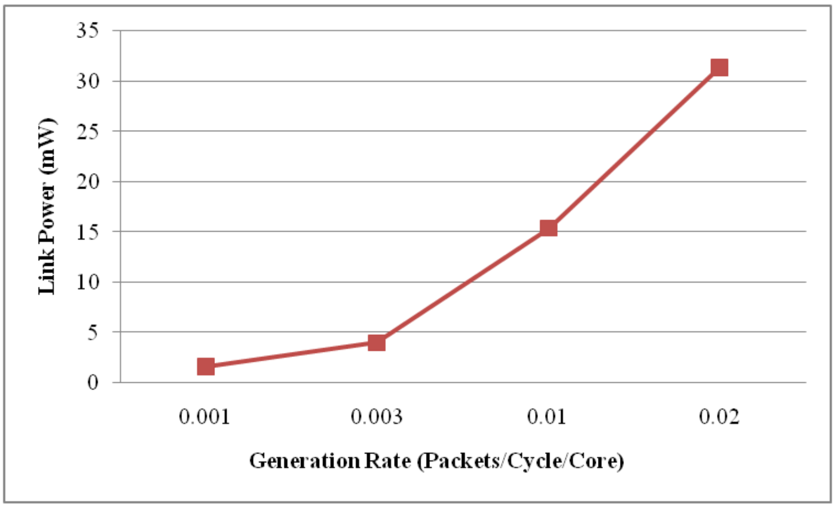


Figure 3.3. The effect of generation rate on link power consumption

According to these simulations, it can be concluded that both bandwidth and distance between the tasks affect the power consumption. Therefore, with the best mapping of the tasks to cores subject to bandwidth and distance, the power consumption is minimized.

3.3.1.3 Number of Virtual Channels

We investigate the impact of number of virtual channels on power consumption in NoC architecture in the presence of different routing algorithms.

The routing algorithms can fall in three categories; deterministic, partially adaptive and fully adaptive. We use X-Y as an example for deterministic algorithm, North-First, Odd-Even (NF/OE) and Duato as instances for partially and fully adaptive, respectively.

Figure 3.4 illustrates power consumption of NoC versus number of virtual channel using X-Y routing algorithm.

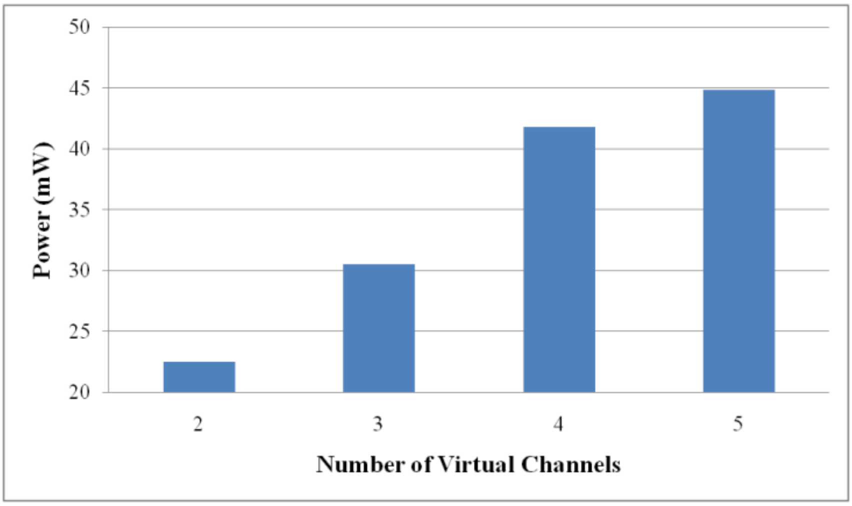


Figure 3.4. The effect of variety of number of virtual channels versus power consumption in X-Y routing algorithm

Regarding to Figure 3.4 the more number of virtual channels the more power would be consumed in NoCs.

Then, we examine the effect of partially adaptive routing such as OE (Odd-Even) and NF (North-First) in the specified NoC. Figure 3.5 shows power consumption versus variety of number of virtual channel in OE/NF routing algorithm.

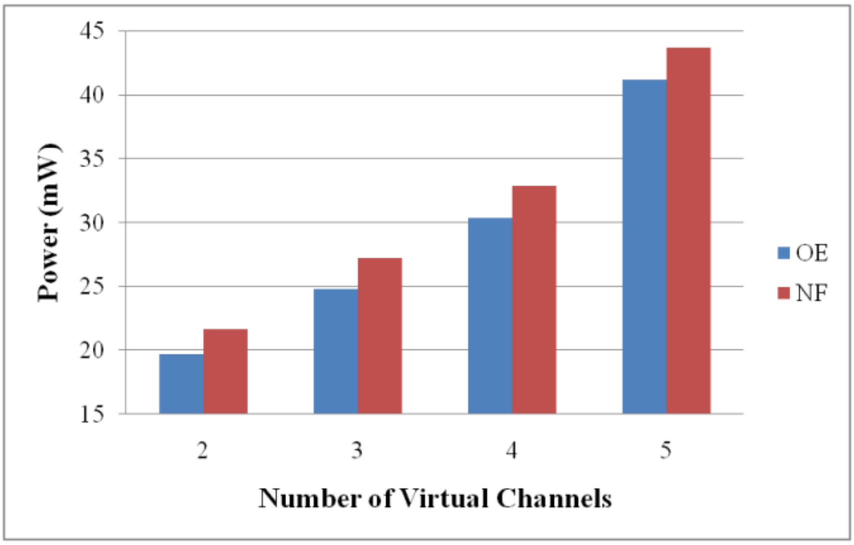


Figure 3.5. The effect of variety of number of virtual channels versus power consumption in OE and NF routing algorithm

As shown in Figure 3.5 by increment the number of virtual channels power will be increased. Finally, Duato algorithm as an example of fully adaptive routing would be evaluated.

Figure 3.6 plots the effect of different number of virtual channels on power consumption in Duato routing algorithm. As it can be observed by using more number of virtual channels the power consumption is increased.

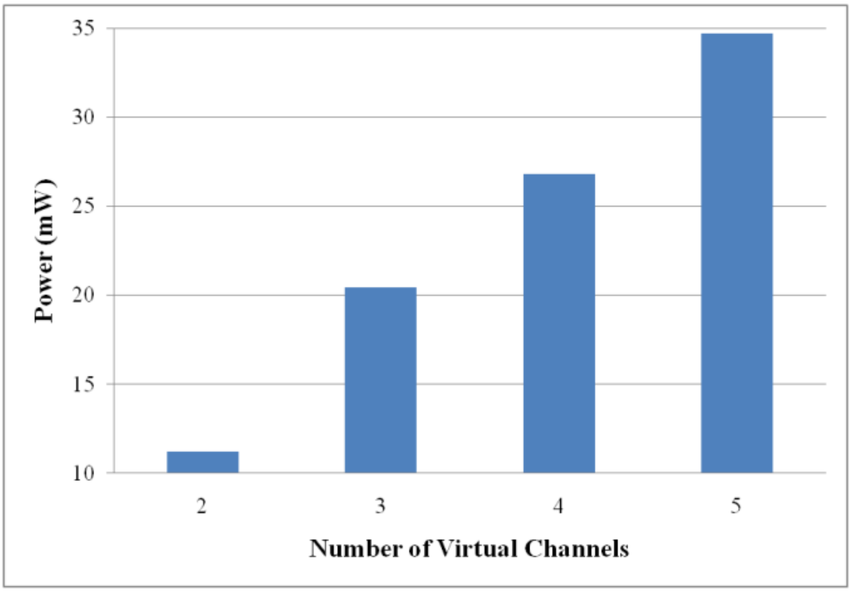


Figure 3.6. The effect of variety of number of virtual channels versus power consumption in Duato routing algorithm

3.3.2 Latency

Latency is the average delay time taken a packet to traverse between sender and receiver. As mentioned in the previous section, generation rate has an impact on the power consumption. On the other hand, there is a relationship between generation rate and latency. Hence, we need to evaluate the effect of the generation rate and number of virtual channels on the latency as a sign of performance of NoCs.

3.3.2.1 Generation Rate

Figure 3.7 shows latency versus generation rate in the NoC architecture. The latency is almost constant until the load of network gets heavy enough; after the network is filled with packets in such a way that network is about to be congested, more generation rate leads in more latency and as a result, the performance of the NoCs goes down. It is clear that the routing algorithm is also effective in when the network gets congested.

Based on the above remarks, it is concluded that after network has enough load the more generation rate we have, the less performance can be gained. It should be mentioned that this simulation has been done on a topology having symmetric cores in terms of number of virtual channels. Obviously, the number of virtual channels has a significant impact on these parameters.

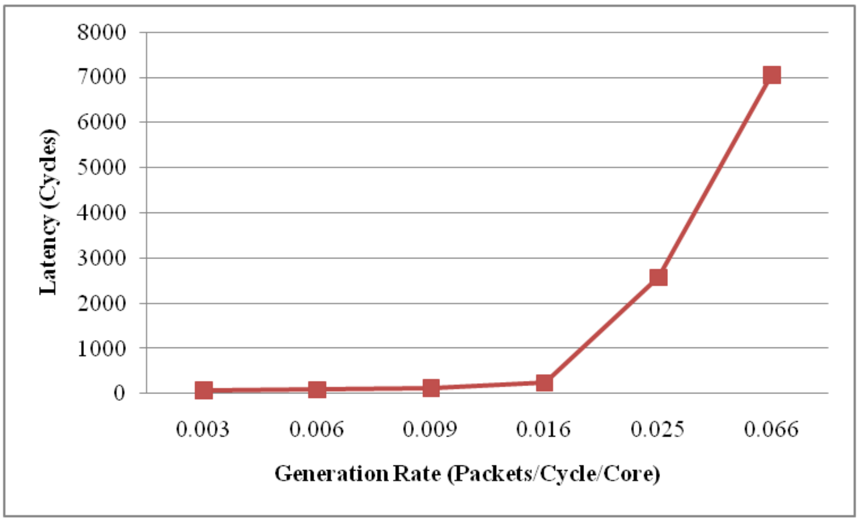


Figure 3.7. The effect of generation rate on latency

3.3.2.2 Number of Virtual Channels

We investigate the effect of different number of virtual channels on latency in NoC architecture in the presence of different routing algorithms. Figure 3.8 illustrates latency of NoC versus number of virtual channel using X-Y routing algorithm.

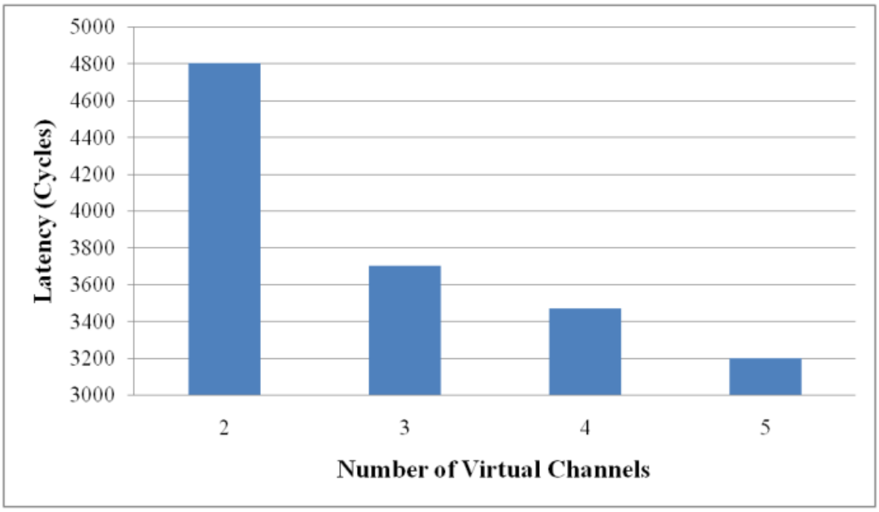


Figure 3.8. The effect of variety of number of virtual channels versus latency in X-Y routing algorithm

Clearly, by increasing the number of virtual channels the latency will be decreased. Putting in other words, with adding the virtual channels packets can traverse more smoothly and channel utilization is increase. Consequently, the performance of the network will be improved in X-Y deterministic routing algorithm. As a result, when the number of virtual channels changes, there is a trade-off between power dissipation and performance of NoC with X-Y deterministic routing. Then, we examine the effect of partially adaptive routing such as OE and NF in the NoC specified above. Figure 3.9 shows latency versus variety of number of virtual channel.

As shown in Figure 3.9, by increasing the number of virtual channels the bandwidth of link can be shared among more messages. This sharing can be done in a non-uniform distribution. Therefore, a message has to pass through several physical links with different degree of multiplexing which causes more latency. It means that adding the number of virtual channels in partially adaptive cannot be useful nor in power dissipation neither in performance.

Finally, Duato algorithm as an example for fully adaptive routing would be evaluated. Figure 3.10 plots the effect of different number of virtual channels on latency. As we see by using more number of virtual channels, the performance will diminish and power consumption increases as well.

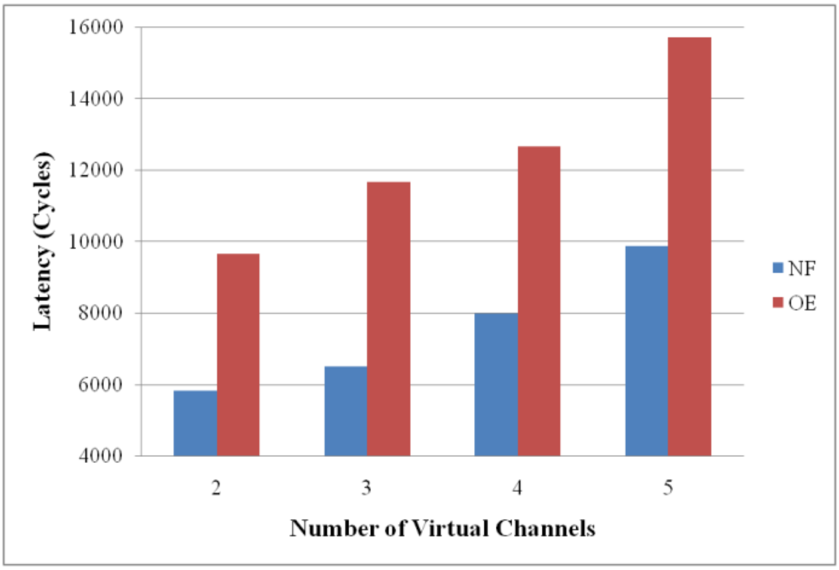


Figure 3.9. The effect of variety of number of virtual channels versus latency in NF and OE routing algorithm

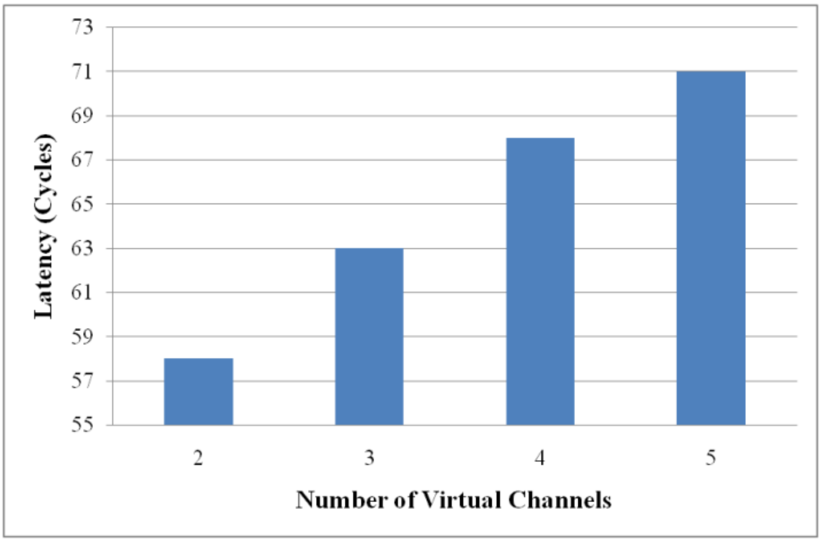


Figure 3.10 The effect of variety of number of virtual channels versus latency in Duato routing algorithm

3.3.2.3 Generation Rate and Number of Virtual Channels

We have evaluated the impact of number of virtual channel and generation rate on latency of NoCs separately so far. In this sub-section, we aim to study the effect of both these factors on NoCs because the impact of number of virtual channel varies on performance of system when generation rate of network gets changed. Obviously, combination of these factors is also worthy to be investigated. We monitor the effect of generation rate and different number of virtual channel on latency using two different routing algorithms, deterministic and fully adaptive algorithms.

As mentioned, the numbers of virtual channels and latency have direct effect on the power consumption. Splitting the physical line into several virtual channels increases the delay on the system. On the other hand, they can improve the throughput because of minimizing the chance of the congestion.

We study impact of number of virtual channel on the performance by considering the different kinds of routing algorithm, X-Y and Duato routing algorithms. In the Figures 3.11 and 3.12, the effect of generation rate and number of virtual channels on the latency in deterministic and fully adaptive routing algorithm is shown, respectively.

As shown in Figure 3.11, in the deterministic routing algorithm there is a trade-off between the throughput and the delay of the router by increasing the number of virtual channels. In fact, when each router has more virtual channels the time which takes packet to pass through that router will be increased. On the other hand, throughput of system is improved because in more virtual channels lead in less congestion in network and the packets can traverse more smoothly through network. Figure 3.11 illustrates that in the deterministic routing algorithm by increasing the number of virtual channels the throughput increases and the latency slightly goes up. By adding the number of virtual channels messages are allowed to pass more smoothly and it leads higher channel utilization and throughput. On the other hand, increasing the number of virtual channels cannot increase routing flexibility significantly and also it has negative effect on latency. If the number of virtual channels in the deterministic routing algorithm exceeds 8, it cannot be useful any more, neither for performance nor for latency of the NoC. We study the effect of number of virtual channel respecting the generation rate and latency in Duato algorithm in Figure 3.12.

According to the Figure 3.12, increasing the number of virtual channel cannot be helpful in Duato algorithm. Thus, it can be concluded that there is no benefit in both throughput and delay when the number of virtual channel in fully adaptive routing algorithm increases.

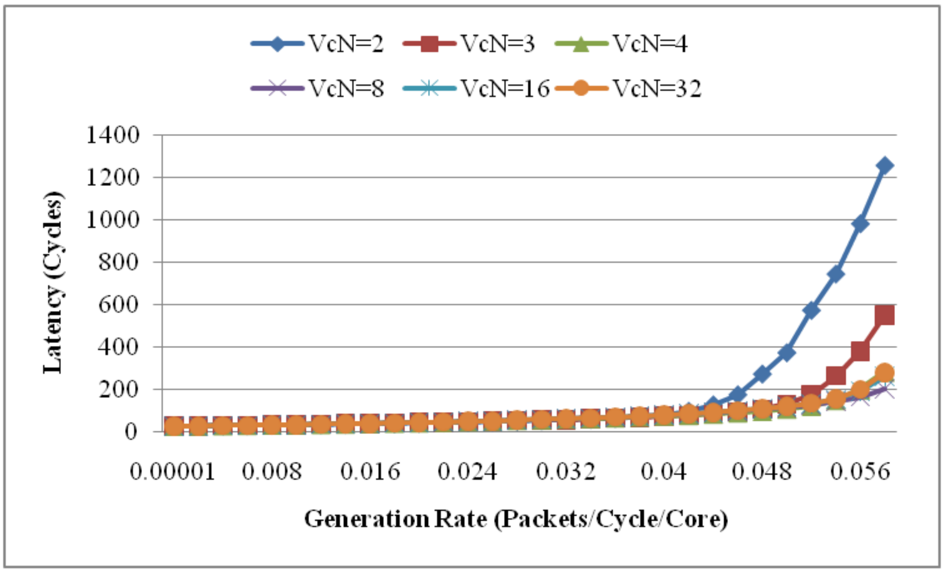


Figure 3.11. The effect of generation rate and variety of number of virtual channels versus latency in X-Y routing algorithm

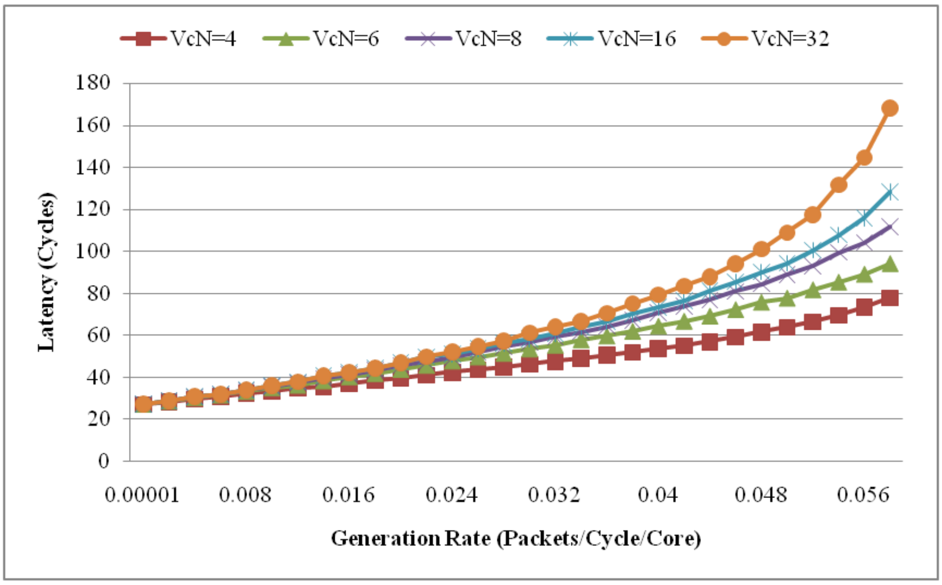


Figure 3.12. The effect of generation rate and variety of number of virtual channels versus latency in Duato routing algorithm

According to the above remarks, it can be concluded that the power consumption depends on the link length, bandwidth between the cores and the number of virtual channels, while performance is affected by bandwidth and the number of virtual channels. Therefore, the power dissipation and performance of NoC can be improved by obtaining the optimum topology based on the distance and bandwidth of the cores with the optimized number of virtual channels.

3.4 Mathematical Modeling

The main goal of the optimization is not only the minimization of the power and energy consumption in custom NoC architecture but also improvement in the overall performance. The proposed method is divided into two sections. First of all, the objective is to extract the optimal layout for application specific design by applying the method of integer programming. In this section, we assign the task to core with respect to distance between cores and bandwidth between the tasks. Secondly, we suggest a method to find the optimum number of routers in our layout considering performance of network to get improvement in energy consumed in NoC.

3.4.1 Task to Core Mapping

The input of this step is a communication task graph depicting how tasks are connected to each other as well as the bandwidth of each connection. To model and find the best mapping for this task graph, Quadratic Assignment Problem (QAP) is used. QAP is a combinatorial optimization problem that assigns a limited number of facilities to a limited number of fixed locations. This method is used to map the tasks to appropriate cores to minimize the total distance of tasks weighted by their related bandwidth. A general QAP denoted by P is defined as follows,

P)

$$min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{l=1}^{n} f_{ij} d_{kl} x_{ik} x_{jl} \tag{3.1}$$

Subject to

$$\forall\, k \in \{1, \ldots, n\}: \sum_{i=1}^{n} x_{ik} = 1 \tag{3.2}$$

$$\forall\, i \in \{1, \dots, n\}: \sum_{k=1}^{n} x_{ik} = 1 \tag{3.3}$$

$$\forall\, i, k \in \{1, \dots, n\}: x_{ik} \in \{0, 1\} \tag{3.4}$$

The variables and parameters of the model are defined according to our mapping problem (with n tasks and n cores) as follows.

- n is the number of tasks and cores of the NoC.
- $i(j)$ and $k(l)$ are indexes used for the tasks and cores respectively.
- f_{ij} is the bandwidth between tasks i and j.
- d_{kl} is the distance between cores k and l.
- x_{ik} is a binary variable that determines the mapping of task i to core k. The value of the variable is 1 if task i is assigned to core k. 0, otherwise.

The constraints set (3.2) ensures that only one task can be assigned to each core while the constraints set (3.3) guarantees that only one core can be assigned to each task. According to the constraints (3.4) the x variables take value of 0 or 1. The objective function of the model minimizes the total distance of tasks that is weighted by the bandwidth of links. Note that in our benchmarks the Euclidian distance of cores is considered.

Generally, QAP is categorized as NP-hard type problem that cannot be solved optimally in a polynomial CPU running time. Several meta-heuristic algorithms e.g. genetic algorithm, simulated annealing, grasp method, etc. were introduced to solve QAP in order to obtain a good feasible solution near to the optimal [50-52]. On the other hand, the quadratic terms of objective function can be linearized using some well-known linearization techniques. The linearized version of QAP is generally easier to be solved optimally by optimization solvers. The full explanation of these linearization techniques is presented by Chaovalitwongse *et al*. [53] and He *et al*. [54]. The linearized version $\bar{P}$ of the above-mentioned problem P is defined as follow,

$\bar{P}$)

$$min \sum_{u=1}^{N} \theta_u \tag{3.5}$$

Subject to the constraints which are equivalent to constraints (3.2) and (3.3)

$$\forall\, k \in \{1, \dots, n\}: \sum_{u=1}^{N} \omega_{ku}\varphi_u = 1 \tag{3.6}$$

$$\forall\, i \in \{1, \dots, n\}: \sum_{u=1}^{N} \beta_{iu}\varphi_u = 1 \tag{3.7}$$

$$\forall\, u \in \{1, \dots, N\}: \left(\sum_{v=1}^{N} \alpha_{uv}\varphi_v\right) - \gamma_u - \theta_u = 0 \tag{3.8}$$

$$\forall\, u \in \{1, \dots, N\}:\ \gamma_u\varphi_u = 0 \tag{3.9}$$

$$\forall\, u \in \{1, \dots, N\}:\ \gamma_u, \theta_u \geq 0 \tag{3.10}$$

$$\forall\, u \in \{1, \dots, N\}:\ \varphi_u \in \{0, 1\} \tag{3.11}$$

The variables and parameters of the model $\bar{P}$ are defined according to our mapping problem (with n tasks and n cores) as follow,

- $N = n^2$
- β_{iu} is a coefficient such that,

$$\forall\, i, u | i \in \{1, \dots, n\} \& \, u \in \{1, \dots, N\}: \beta_{iu} = \begin{cases} 1, & if (i-1)n + 1 \leq u \leq in \\ 0, & Otherwise \end{cases}$$

- ω_{ku} is a coefficient such that,

  $\forall\, k,u|k \in \{1,\dots,n\}\&\, u \in \{1,\dots,N\}$:

  $$\omega_{ku} = \begin{cases} 1, & if\ \exists z \in \{1,\dots,n\}: u = (z-1)n + k \\ 0, & Otherwise \end{cases}$$

- α_{uv} is a coefficient such that,

  $\forall\, u,v \in \{1,\dots,N\}: \alpha_{uv} = f_{ij}d_{kl}$

$$i = \begin{cases} \lfloor u/n \rfloor + 1, & if\ u/n \text{ is not integer} \\ u/n, & if\ u/n \text{ is integer} \end{cases}$$

$$j = \begin{cases} \lfloor v/n \rfloor + 1, & if\ v/n \text{ is not integer} \\ v/n, & if\ v/n \text{ is integer} \end{cases}$$

where $\lfloor v/n \rfloor$ shows the lower integer limit of $\frac{v}{n}$.

$$k = \begin{cases} u \bmod n, & if\ u/n \text{ is not integer} \\ n, & if\ u/n \text{ is integer} \end{cases}$$

$$l = \begin{cases} v \bmod n, & if\ v/n \text{ is not integer} \\ n, & if\ v/n \text{ is integer} \end{cases}$$

- φ is a binary variable presented by an N dimensional vector such that, $i^{th}n$ variables of the vector shows the assignment of task i ($i \in \{1,\dots,n\}$), therefore, decision variable x_{ik} in P is exactly equivalent with the decision variable φ_u in $\bar{P}$ such that $u = (i-1)n + k$ or $k = u \bmod n$. This equivalency is guaranteed by constraints (3.6) and (3.7).
- γandθ are continuous nonnegative variables.
- Based on the definition of α and φ:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n} f_{ij}d_{kl}x_{ik}x_{jl} = \sum_{u=1}^{N}\sum_{v=1}^{N} \alpha_{uv}\varphi_u\varphi_v$$

Now, the following theorem shows the equivalency of $P$and$\overline{P}$.

Theorem. Problems P and $\bar{P}$ are equivalent such that,

$\forall i, k \in \{1, \dots, n\}$: $x_{ik}{}^*$ is the optimal solution of P if and only if $\forall\, u \in \{1, \dots, N\}$: $(\varphi_u{}^*,\ \gamma_u{}^*,\ \theta_u{}^*)$ is the optimal solution of $\bar{P}$ under the assumptions $x_{ik}{}^* = \varphi_u{}^*$ and $u = (i-1)n + k$.

Proof. *Necessity*: Let $\forall\, i, k\ \in \{1, \dots, n\}$: $x_{ik}{}^*$ be an optimal solution of P. As all elements of α_{uv} are nonnegative, it is clear that,

$$\forall\, u \in \{1, \dots, N\}: \exists\, \gamma_u, \theta_u \mid \gamma_u, \theta_u \geq 0$$

such that

$$\forall\, u \in \{1, \dots, N\}: \left(\sum_{v=1}^{N} \alpha_{uv}\varphi_v\right) - \gamma_u - \theta_u = 0 \tag{3.12}$$

$$\forall\, u \in \{1, \dots, N\}: \gamma_u \varphi_u = 0 \tag{3.13}$$

Now based on (3.12) and (3.13), $\forall\, u \in \{1, \dots, N\}$: $\gamma_u{}^*, \theta_u{}^*$are selected to minimize $\sum_{u=1}^{N} \theta_u{}^*$. Then, the theorem is proved if we prove that $\forall\, u \in \{1, \dots, N\}$: $(\varphi_u{}^*, \gamma_u{}^*, \theta_u{}^*)$ is the optimal solution of $\bar{P}$.

The equation (3.12) is multiplied by φ_u. Therefore,

$$\forall\, u \in \{1, \dots, N\}: \left(\varphi_u{}^* \sum_{v=1}^{N} \alpha_{uv}\varphi_v{}^*\right) - \varphi_u{}^*\gamma_u{}^* - \varphi_u{}^*\theta_u{}^* = 0 \tag{3.14}$$

The set of equation (3.14) by applying equation (3.13) turns out to,

$$\forall\, u \in \{1, \dots, N\}: \left(\varphi_u{}^* \sum_{v=1}^{N} \alpha_{uv}\varphi_v{}^*\right) = \varphi_u{}^*\theta_u{}^* \tag{3.15}$$

that implies,

$$\sum\nolimits_{u=1}^{N} \varphi_u{}^* \left(\sum_{v=1}^{N} \alpha_{uv}\varphi_v{}^*\right) = \sum\nolimits_{u=1}^{N} \varphi_u{}^*\theta_u{}^* \tag{3.16}$$

If we prove that $\sum_{u=1}^{N} \varphi_u{}^* \theta_u{}^* = \sum_{u=1}^{N} \theta_u{}^*$, then, $\forall\, u \in \{1, \dots, N\}$: $(\varphi_u{}^*, \gamma_u{}^*, \theta_u{}^*)$ is the optimal solution of $\bar{P}$. To prove this equation it is enough to prove that for any u if $\varphi_u{}^* = 0$ then $\theta_u{}^* = 0$. The proof is done by the following contradiction.

Define set A such that $A \subset \{1, \dots, N\}$. Consider $\forall\, t \in A$: $\varphi_t{}^* = 0$ and $\theta_t{}^* > 0$ such that $\forall\, u \in \{1, \dots, N\}$: $\gamma_u{}^*, \theta_u{}^*$minimizes $\sum_{u=1}^{N} \theta_u{}^*$.We also define $\tilde{\gamma}$ and $\tilde{\theta}$ as $\forall\, t \in A$: $\widetilde{\gamma_t} = \gamma_t{}^* + \theta_t{}^*, \widetilde{\theta_t} = 0$ such that $\forall\, v \notin A$: $\widetilde{\gamma_v} = \gamma_v{}^*, \widetilde{\theta_v} = \theta_v{}^*$. Clearly, $\forall\, u \in \{1, \dots, N\}$: $\varphi_u{}^*, \widetilde{\gamma_u}, \widetilde{\theta_u}$ satisfy (3.12) and (3.13) such that,

$$\sum_{u=1}^{N} \widetilde{\theta_u} < \sum_{u=1}^{N} \theta_u{}^*$$

This contradicts with the assumption that $\forall\, u \in \{1, \dots, N\}$: $\gamma_u{}^*, \theta_u{}^*$minimizes $\sum_{u=1}^{N} \theta_u{}^*$.

Sufficiency: Assume that the vectorsφ^*, γ^*, and $\theta^*$are optimal in problem $\bar{P}$ then in a similar way it can be seen that $\sum_{u=1}^{N} \varphi_u{}^* \theta_u{}^* = \sum_{u=1}^{N} \theta_u{}^*$. If φ^* is not optimal in problem P then there is an even better solution of problem $\bar{P}$ what can be seen again in a similar way, and it is a contradiction. □

In the nonlinear constraint $\forall\, u \in \{1, \dots, N\}$: $\gamma_u \varphi_u = 0$, for each u, if $\varphi_u = 1$, then $\gamma_u = 0$, but if $\varphi_u = 0$ the value of γ_u is not depended on this constraint. Based on constraint (3.8) the upper bound of γ_u is calculated as,

$$M = \max_u \sum_{v=1}^{N} |\alpha_{uv}|$$

Therefore, in $\bar{P}$ the constraint $\forall\, u \in \{1, \dots, N\}$: $\gamma_u \varphi_u = 0$ is replaced by the following constraint which has the same restriction.

$$\forall\, u \in \{1, \dots, N\}: \gamma_u \le M(1 - \varphi_u)$$

Finally, $\bar{P}$ is reformulated as,

$$min\sum_{u=1}^{N}\theta_u \tag{3.17}$$

subject to

$$\forall\, k \in \{1,\dots,n\}: \sum_{u=1}^{N}\omega_{ku}\varphi_u = 1 \tag{3.18}$$

$$\forall\, i \in \{1,\dots,n\}: \sum_{u=1}^{N}\beta_{iu}\varphi_u = 1 \tag{3.19}$$

$$\forall\, u \in \{1,\dots,N\}: \left(\sum_{v=1}^{N}\alpha_{uv}\varphi_v\right) - \gamma_u - \theta_u = 0 \tag{3.20}$$

$$\forall\, u \in \{1,\dots,N\}:\ \gamma_u \le M(1-\varphi_u) \tag{3.21}$$

$$\forall\, u \in \{1,\dots,N\}:\ \gamma_u, \theta_u \ge 0 \tag{3.22}$$

$$\forall\, u \in \{1,\dots,N\}:\ \varphi_u \in \{0,1\} \tag{3.23}$$

As a conclusion, the linearized model of (3.17)-(3.23) is solved to assign the tasks to the cores optimally.

3.4.2 Optimizing of the Number of Routers in NoCs

There are two major sources of power consumption in NoCs, physical links (self and coupling capacitances) and routers; it means that the number of routers plays a significant role in the total power dissipation of the NoC.

Due to this fact, by using an optimization method, the minimum required number of routers can be obtained without a significant degradation in the performance. The input of this step is the optimal layout of NoC obtained in the previous part. In this layout, each task is assigned to one core having a router called own router. Besides these own routers, we insert one dummy router in the center of intersections of WSM. These dummy routers improve the flexibility of our algorithm to minimize the number of routers. The router's selection method is based on the gained topology which is found in the mapping step and the connections between the tasks coming from communication trace graph. These dummy routers increase the number of options for Set Covering Problem and as a result, they give this algorithm more flexibility and help it to check all the possibilities in the given topology and find the optimum number of routers. The routers are selected according to the layout with optimum mapping and the connections among the tasks. That is, based on the optimum mapping produced in the previous step the algorithm tries to find the optimum number of routers among the dummy and own routers in such a way that all the connections defined in the bandwidth matrix are met. Apparently, dummy routers may or may not be selected by the proposed method. The notations of the method are mentioned as follows,

- R is the set of prepositioned routers on the NoC. As mentioned above, for each core one own router and some extra dummy routers in the spaces between cores are placed.
- As each positive element of the bandwidth matrix is a connection between a pair of tasks, therefore, a connection of a pair of tasks represents the connection of their associated cores. C is the set of all connections of NoC. If there are n tasks on the NoC, there will be at most n^2 for directed and $\frac{n(n+1)}{2}$ for bidirectional connections (shown by matrix of bandwidths) note that set C contains the positive bandwidth values.

$$C = \{1, 2, \dots, p\} \mid p \leq \binom{n}{2} = \frac{n(n-1)}{2} \tag{3.24}$$

T is the set of all tasks which has n entities.

- The set of connections that serve each core is collected by,

  $\forall i \in T: CC_i = \{C_j \in C \mid \text{connection } j \text{ connects core } i \text{ to any other core}\}$

- The set of connections that can be served by each router is defined by,

  $\forall j \in R: CR_j = \{C_k \in C | \text{ connection } k \text{ can use router } j \text{ to send data}\}$

The method follows the steps described below,

- For each task, the set of its related connections from set C is defined by C_i such that $\bigcup_{i \in T} C_i = C$.
- For each connection of set C, its potential routers from set R is obtained as set CR_i such that $\bigcup_{i \in C} CR_i \subseteq R$.
- For each router of set R, binary variable X is defined. If the value of X is 1, the router should be used on NoC, if it is 0, the router is eliminated from the NoC. Then, the following mathematical model is solved to eliminate the unrequired routers of set R.

$$min \sum_{j \in R} X_j \tag{3.25}$$

subject to

$$\forall\, i \in C: \sum_{j \in CCR_i} X_j \geq 1 \tag{3.26}$$

$$\forall\, j \in R:\ X_j \in \{0,1\} \tag{3.27}$$

- Set CCR for each core is defined as,

$$\forall i \in C: CCR_i = \{k \mid k \in R\ \& CC_i \subseteq CR_k\} \tag{3.28}$$

Constraints (3.26) in existence of the minimization objective function (3.25) guarantee that each core which is a part of at least one connection, is connected to at the maximum one router.

- Considering constraints (3.26), the objective function (3.25) minimizes the number of routers.

The proposed model finds minimum number of routers for the given layout of NoC based on the initial layout of routers of NoC.

3.5 Experimental Results

We evaluate the proposed method for five benchmarks. The optimization problem is divided into two steps. Our approach is compared to non-optimized layout in terms of power dissipation, latency and energy consumption. Finally, we compare OPAIC with MOCA [38].

3.5.1 Benchmarks

Five multimedia benchmarks are selected to perform the computational experiments on the introduced algorithms. Each benchmark contains some tasks and a matrix representing the bandwidth between each pair of the tasks. In Table 3.1, the characteristics of the multimedia benchmarks such as H.263 video encoder, H.263 video decoder, MP3 audio encoder [40], Video object plane decoder (VOPD) and Multi-window display (MWD) [55] are depicted. The number of different tasks varies from 8 nodes to 15 which are connected by edges ranging from 11 to 19.

Table 3.1. Benchmarks' characteristics

Graph	Graph ID	Number of Tasks	Number of Edges
H.263 encoder	G1	8	11
MWD	G2	12	13
VOPD	G3	12	15
H.263 enc MP3 enc	G4	14	19
H.263 enc H.263 dec	G5	15	19

3.5.2 Experimental Results for Mapping Step

In the mapping step of the optimization section, QAP and its linearized version are introduced. QAP is linearized by the previous section. Generally, the linearized versions of QAPs are more effective than QAP to be solved, thus, the linearized version of QAP is used to formulate task to core mapping problem. The communication task graph (CTG) of a specific application and weighted super mesh (WSM) topology are inputs and proposed linearized formulation of QAP tries to map the tasks of CTG to the best cores in WSM in such a way that the two tasks with higher bandwidth communication mapped to the closer cores . The number of the cores in WSM is considered to be more than the number of the tasks in CTG, to have more flexibility in the mapping procedure. There is a trade-off in number of these dummy cores; the more number of dummy cores, the more options software has to choose and better result would be gained but run time goes up. As an instance for VOPD with 12 tasks, we consider 5*5 WSM topology. Therefore, there will be 13 dummy cores. The optimization problem of task to core mapping was coded in Xpress optimizer [56]. Xpress solved the linearized model optimally. Figure 3.13 and Table 3.2 illustrate the communication trace graph and node description of VOPD benchmark, respectively. In Figure 3.13, the edges of the graph are denoted with bandwidth demand in Mb per second.

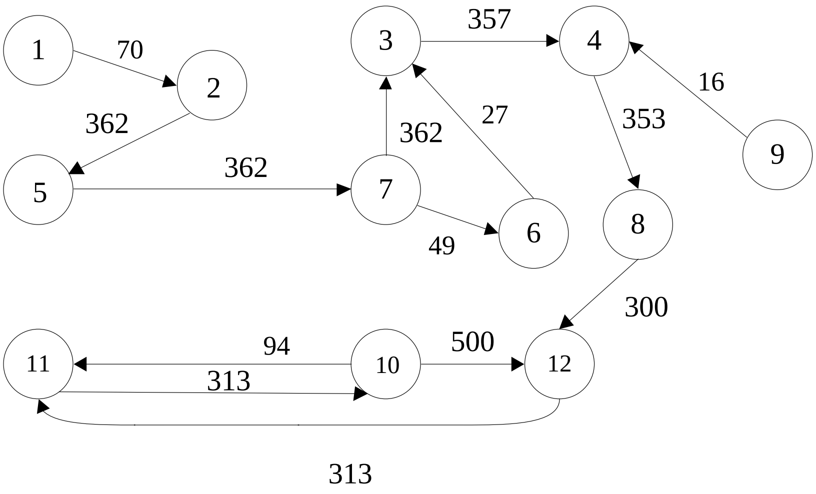


Figure 3.13. Communication Trace Graph for VOPD

Table 3.2. Node descriptions for VOPD

Node	Description
1	VLD
2	RUN LEN DEC
3	IQUANT
4	IDCT
5	INV SCAN
6	STRIPE MEM
7	ACDC PRED
8	UP SAMP
9	ARM
10	VOP MEM
11	PAD
12	VOP REC

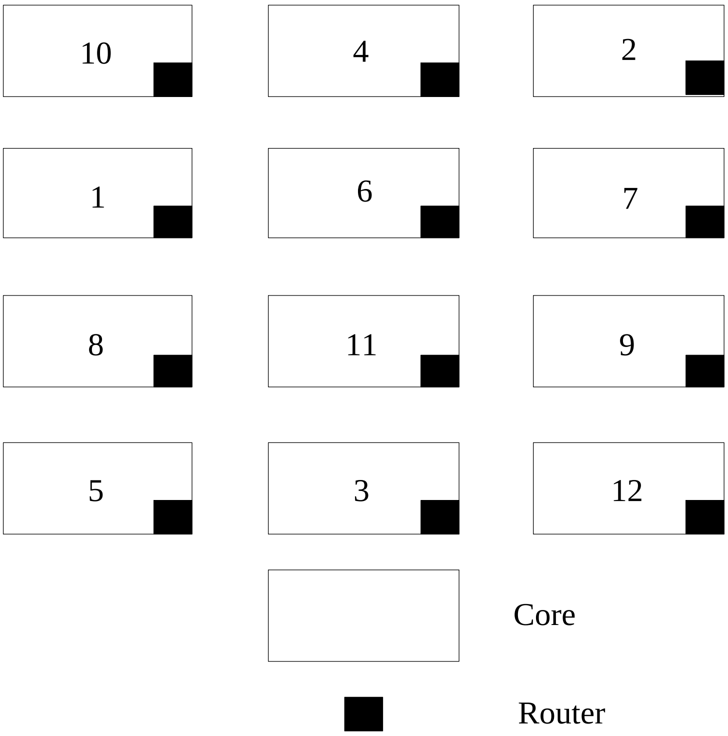


Figure 3.14. Non-optimized layout for VOPD

In Figure 3.14, the non-optimized layout of NoC for VOPD benchmark is depicted and after applying our method for solving the problem of task to core mapping, optimized layout obtained by the proposed method is presented by Figure 3.15. In Figures 3.14 and 3.15 the rectangular represents core and darkened squares indicate the routers in NoC.

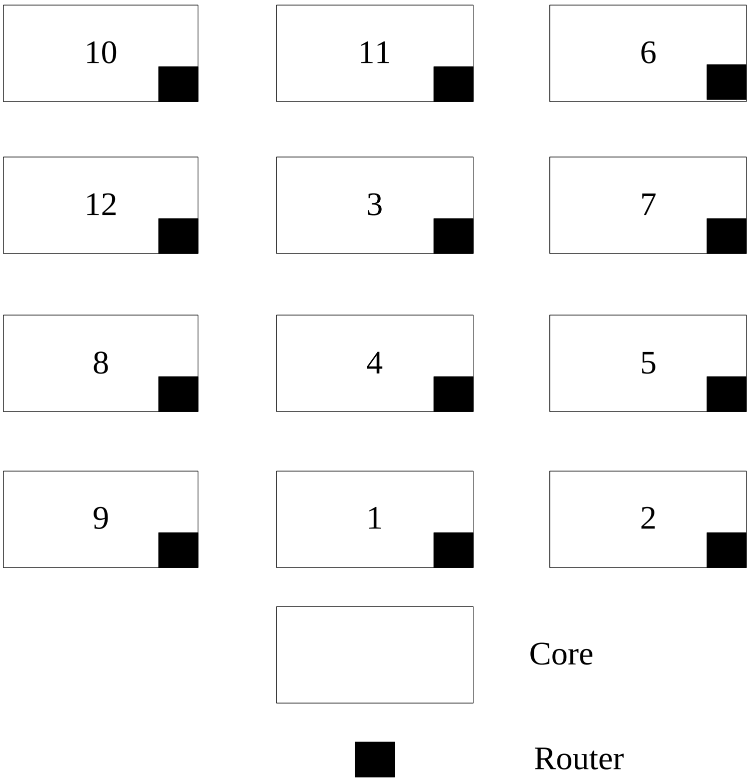


Figure 3.15. Optimized layout for VOPD with proposed method

3.5.3 Experimental Results for the Least Number of Routers

In part 3.4.2, an algorithm based on a linear optimization model is introduced to reduce the number of routers on the NoC. Set of dummy and own routers of the NoC in the optimum layout obtained in the last step are determined and considered as inputs of this step as shown in Figure 3.16. A linear optimization model minimizes the total number of routers on the NoC conditioning that each task is served by one router of its set of assignable routers. We use Xpress optimizer [56] to solve the problem of minimization of router resources. Due to the key contribution of routers in the power consumption, router reduction has a great impact in decreasing the power dissipation. The optimized layout for VOPD benchmark with the optimum number of routers is depicted in Figure 3.17.

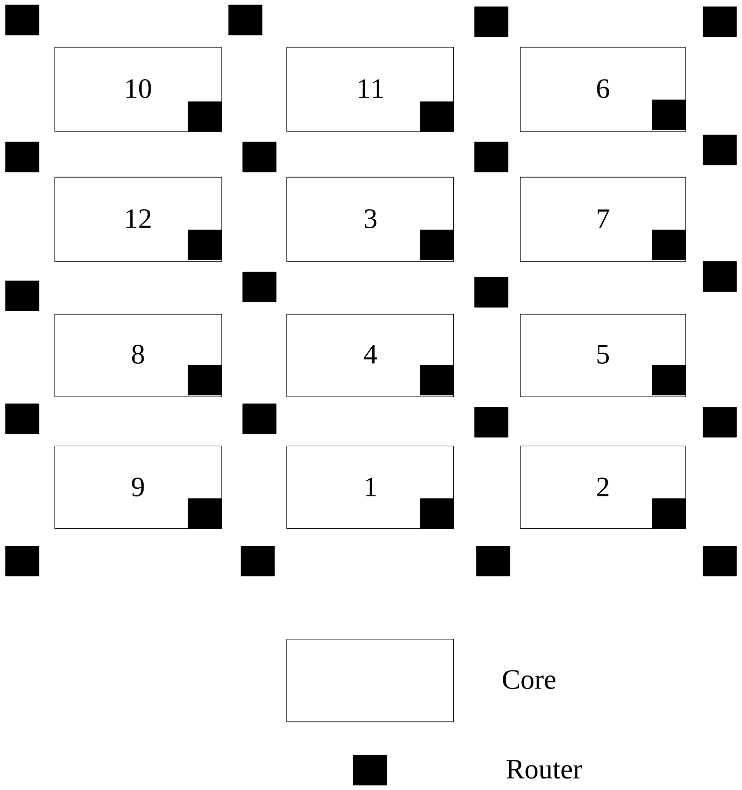


Figure 3.16. Optimized VOPD layout with dummy and own routers

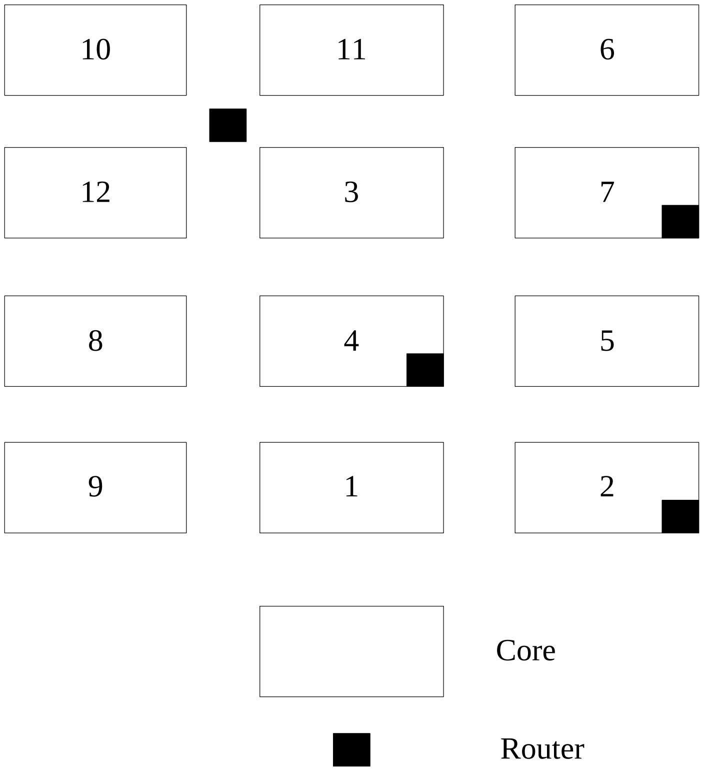


Figure 3.17. Optimized VOPD layout with optimum number of routers

3.5.4 Experimental Results of Implementation

In our implementation, topology characterized in 65 nm technology, $V_{dd}$is equal to 1 Volt and according to the critical path of the system the clock frequency is 500 MHz. Length of the metal wires is 2 mm. The capacitance of the wire links and coupling capacitance are selected as 0.2 pF/mm and 0.6 pF/mm respectively [1]. Modelsim is used to evaluate the number of the transitions of wires under the uniform distribution for sending the packets. Power of physical link and coupling power consumption is considered as a total power of links in NoC. We compare the results obtained by applying of the proposed method on five multimedia benchmarks with non-optimized NoC. Figure 3.18 depicts the power consumption, the horizontal axis of the Figure indicates the graph ID and the vertical axis indicates the normalized value of power dissipation. The values in Figure 3.18 are normalized to the corresponding power consumption by the non-optimized layout. In comparison to the non-optimized the proposed algorithm consumes, on average, 67.7% lower power dissipation.

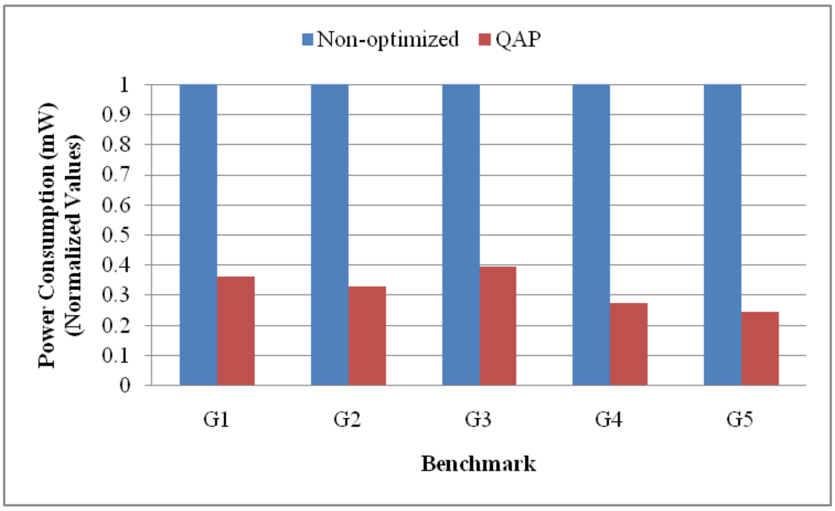


Figure 3.18. Power comparison of non-optimized and QAP

Latency comparison is shown in Table 3.3. As illustrated in this Table, the proposed method improves the latency, on an average, 33.8% compared to the non-optimized layout.

Table 3.3 Comparison of latency between non-optimized and QAP

Latency (Cycles)	Graph ID	Non-optimized	QAP	Improvement (%)
H.263 encoder	G1	5015	2995	40.27
MWD	G2	69	51	26.08
VOPD	G3	456	297	34.86
H.263 enc MP3 enc	G4	319	289	9.40
H.263 enc H.263 dec	G5	5202	2153	58.61

Table 3.4 presents the energy consumption. As shown, the energy of the proposed method compares to the non-optimized NoC, on average, 67.7% is improved.

Table 3.4 Comparison of energy consumption between non-optimized and QAP

Energy (μJ)	Graph ID	Non-optimized	QAP	Improvement (%)
H.263 encoder	G1	34.00	12.35	63.67
MWD	G2	52.72	17.45	66.90
VOPD	G3	59.51	23.50	60.51
H.263 enc MP3 enc	G4	7.82	2.15	72.50
H.263 enc H.263 dec	G5	89.38	22.00	75.38

As mentioned, researchers in [39] proposed a mesh based method subject to the latency by the name of MOCA. In [39] researchers compared their method with two previous state of the art works which are called NMAP [36] and MILP [44]. NMAP does not consider latency constraint and as a result NMAP fails for most of the benchmarks [39]. They compare MILP and MOCA in two cases: with and without latency constraints. The results of [39] show that MOCA can improve power consumption in comparison with MILP in both cases. We compare the proposed method with a non-optimized layout of NoC with the same number of cores and also with MOCA in two cases, with latency and without latency constraints, as best approach comparatively to the others.

Table 3.5 compare the results obtained for OPAIC, non-optimized NoC architecture and MOCA for VOPD benchmark. MOCA does not optimize the number of routers; thus, it is not able to decrease the area consumption compared with non-optimized NoC architecture. In the other word, Non-optimized NoC and MOCA algorithm have same number of routers because this algorithm does not have any router reduction method. The proposed algorithm intends to optimum power consumption with trivial effect on performance of system in such a way that the energy consumed in NoC would be dwindled. The results show that even though performance of OPAIC compared with the previous methods has a trivial increase, due to optimal mapping and number of router reduction, the energy of the system is improved drastically. The connections, locations and number of routers in both the non-optimized NoC topology and MOCA impose the data to pass through more routers rather than proposed method, therefore, energy dissipation is increased.

As shown in Table 3.5, the proposed method consumes on average 76.2% and 76.1% less power and energy dissipation, respectively while the pervious methods require 3 times as many routers as OPAIC topology has. The area consumption of OPAIC, on an average, is 35.3% less than non-optimized NoC architecture and MOCA approach.

Table 3.5 Comparison of OPAIC

Layout	Total Power (mW)	Latency (Cycles)	Total Energy (μJ)	No. of Router	Area (μ$\boldsymbol{m^2}$)
Non-optimized NoC	24.72	456	376.03	12	48834
MOCA with latency[39]	22.43	273	341.08	12	48834
MOCA without latency[39]	22.38	290	340.34	12	48834
OPAIC	5.33	303	81.11	4	31550

3.6 Summary

Network on Chip (NoC) is an appropriate and scalable solution for today's System on Chips (SoCs) with the high communication demands. Application specific NoCs is preferable since they can be customized to optimize all requirements of the specific applications. This chapter presents an OPtimization technique for Application specifIC NoCs (OPAIC) that addressed the mapping and obtaining the optimum number of routers for application specific NoC architectures which aims not only to decrease the energy consumption but also to improve the area of NoCs. OPAIC is composed of two stages to find the optimum NoC; in the first stage, it uses a linearized form of a Quadratic Assignment Problem (QAP) to map tasks on cores to minimize the energy. In the second stage, a Mixed Integer Linear Problem (MILP) is proposed to find the optimum number of the routers for the layout earned in previous stage.

The most important effective factors on energy and performance of NoC are used comprehensively; these factors contain bandwidth, link length, latency, number of the routers which are considered as different constraints in OPAIC. In comparison with the previous work, OPAIC is able to reduce on average 76.1% of the energy consumption as well as 35.3% of area of implementation.

Chapter 4

CONCLUSION AND FUTURE WORK

4.1 Conclusion

This dissertation has covered low power encoding and different optimization approaches to design Network on Chip architectures. The low power encoding algorithm presented in this work improves the power dissipation without any significant degradation in the performance of NoCs compared to the state of the art low power encoding algorithms. The presented methods for optimization tackle power and performance by core mapping problem and router reduction approach. The proposed approaches allow NoC designers to achieve power and performance improvements.

In the first part of this dissertation, presented in Chapter 2, a novel low power encoding algorithm is proposed. The proposed encoding method reduces the power consumption in NoCs compared to the state of the art low power encoding algorithms. We also assess the impact of the network parameters on effectiveness of our method.

In the second part of this dissertation, presented in Chapter 3, we proposed an optimization technique which is composed of two contributions to design the optimum NoC. In the first section, we introduce a linearized form of a Quadratic Assignment Problem (QAP) to map tasks on cores to reduce the energy dissipation.

In the second section, an Integer Linear Problem is proposed to obtain the optimum number of the routers for the layout earned in previous section.

4.2 Future Work

Although the energy, power and latency improvement in the proposed methods are promising, we believe that QAP as a layout problem to find the best layout for NoCs can be solved with the other approaches as follows:

4.2.1 Meta-heuristic Algorithms

Mapping problem is classified as NP-hard problem [47]. Thus, due to the running time of mapping algorithm, the key factors such as objective function, constraints and evaluation should be considered more precisely.

Murali et al. [57] introduce a heuristic approach for mapping the cores onto a mesh-based NoC with bandwidth constraint such that the average communication delay is minimized. Researchers in [58] investigate a multi objective mapping method for mesh-based NoC architecture. They use a genetic algorithm to find Pareto mapping which is able to optimize power dissipation and performance in NoC. The authors in [59] propose a genetic algorithm based method that is used for application specific NoC. The objective function of this approach is power and router resources reduction. In [60], an Integer Linear Programming is presented that is followed by a heuristic algorithm according to the Particle Swarm Optimization to find the best position for the router with communication cost minimization objective. They focus on the application specific NoC to obtain an appropriate position for the routers from the list of available router’s position. In [61], a mapping algorithm is introduced with multi objective genetic algorithm to find Pareto front solutions for different network topology with respect to power dissipation and latency.

In this section, meta-heuristic algorithms are suggested to design the layout of NoCs. Genetic algorithm (GA) is one of the population based evolutionary algorithm that is widely used to solve combinatorial optimization problems [62, 63] as well as simulated annealing (SA) algorithm which is a single solution based meta-heuristic method that is also widely used to solve combinatorial optimization problems [64].

We suggest meta-heuristic algorithms to assign the tasks with higher bandwidth to the cores with lower distance and vice versa to minimize the objective function of the QAP. GA, SA and a GA which is hybridized by a SA while the objective function is calculated using crisp values of bandwidth and distance are suggested to solve QAP. GA starts with a population of solutions (chromosomes). First $p\%$ (randomly selected) of the solutions (worst solutions of the population) are randomly reproduced and evaluated. Then, one/some of the remaining $(1-p)\%$ of the solutions are selected as parents to generate a new solution (named child or offspring) using crossover, mutation and other operators. The new solution is evaluated and the worst solution among the population is replaced by the new solution if the new one has better objective function than the worst solution. Selection of parents, generation of the child, evaluation of the child and its replacement with the worst solution of the population (if happens) are together considered as one iteration of the GA. Then, the GA is terminated after a limited number of iterations and the best solution among the final population is introduced as the solution of the algorithm.

SA algorithm is a single solution based meta-heuristic method which is also widely used to solve combinatorial optimization problems [64]. The SA algorithm starts with a single initial solution (which is also considered as best solution) and an initial temperature. Then, for a limited number of iterations a neighbor of the initial solution is generated. In the SA algorithm, swap operator is used to generate the neighbor solution. In each iteration the initial solution and the best one are replaced by the generated neighbor solution if the neighbor solution has better objective value than the best one. If not, the neighbor may be accepted by a random generated probability as just the initial solution. Otherwise, the initial and the best solutions are unchanged. Whether it is changed or not, the initial solution is the input of the next iteration. After all iterations of the initial temperature are done, the procedure continues by cooling down the initial temperature until a predefined final temperature is met. In each subsequent temperature, the same limited number of iterations as the number of iterations of the initial temperature, are performed to improve the initial and best solutions. At the end, the output of the SA algorithm is the last saved best solution.

Now, the GA is hybridized by the SA algorithm. In each iteration of the GA a child is generated by a crossover operator. The generated child is considered as the initial solution of the SA algorithm to be improved if possible. Then the improved solution is sent back to the population of the GA to be replaced with the worst solution among the population. The algorithm continues until all iterations of the GA are done. Note that in each iteration of the GA, the SA algorithm is performed. In Figure 4.1 the flowchart of the suggested approach is depicted.

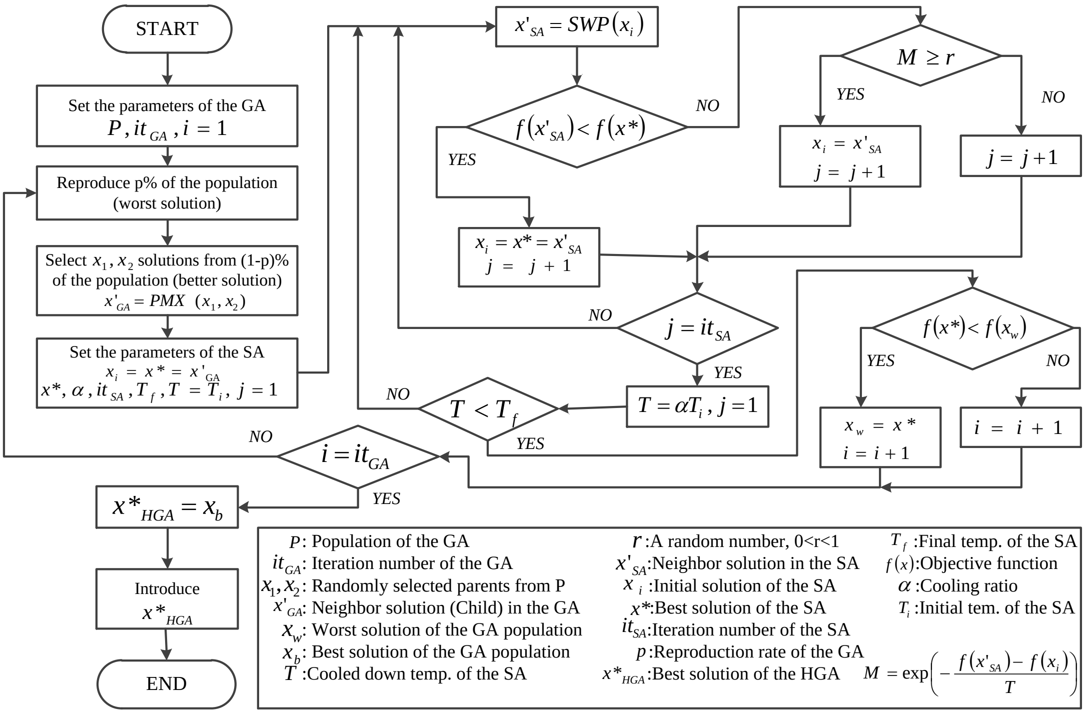

START
Set the parameters of the GA
$P, it_{GA}, i = 1$
Reproduce p% of the population (worst solution)
Select x_1, x_2 solutions from (1-p)% of the population (better solution)
$x'_{GA} = PMX\ (x_1, x_2)$
Set the parameters of the SA
$x_i = x^* = x'_{GA}$
$x^*, \alpha, it_{SA}, T_f, T = T_i, j = 1$
$x'_{SA} = SWP(x_i)$
$f(x'_{SA}) < f(x^*)$
YES
NO
$x_i = x^* = x'_{SA}$
$j = j + 1$
$M \geq r$
YES
NO
$x_i = x'_{SA}$
$j = j + 1$
$j = j + 1$
$j = it_{SA}$
NO
YES
$T = \alpha T_i, j = 1$
$T < T_f$
NO
YES
$f(x^*) < f(x_w)$
YES
NO
$x_w = x^*$
$i = i + 1$
$i = i + 1$
$i = it_{GA}$
NO
YES
$x^*_{HGA} = x_b$
Introduce
x^*_{HGA}
END
P: Population of the GA
it_{GA}: Iteration number of the GA
x_1, x_2: Randomly selected parents from P
x'_{GA}: Neighbor solution (Child) in the GA
x_w: Worst solution of the GA population
x_b: Best solution of the GA population
T :Cooled down temp. of the SA
r :A random number, 0<r<1
x'_{SA}:Neighbor solution in the SA
x_i :Initial solution of the SA
x^*:Best solution of the SA
it_{SA}:Iteration number of the SA
p :Reproduction rate of the GA
x^*_{HGA} :Best solution of the HGA
T_f :Final temp. of the SA
$f(x)$:Objective function
α :Cooling ratio
T_i :Initial tem. of the SA
$M = \exp\left(-\frac{f(x'_{SA}) - f(x_i)}{T}\right)$

Figure 4.1. Flowchart of the suggested approach

4.2.2 Fuzzy-based Meta-heuristic Algorithm

Decision making affected by uncertainty and fuzzy controller is the best choice for dealing with uncertainty. Fuzzy controller has an advantage of avoidance in rigid boundaries by providing a level of confidence to the data which are really important to remove ambiguities and solve the problem that are difficult from mathematical perspective. Since Quadratic Assignment Problem (QAP) in optimization algorithms is complicated (QAP is combinatorial optimization problem known as a NP-hard problem), the utilization of fuzzy concepts can be helpful to improve the efficiency of our evaluation.

After the algorithm is applied to certain data of NoCs, to make a better decision based on the objective function, fuzzy logic rules are proposed. According to these rules we would be able to find the optimum solution for the tasks to cores mapping.

The fuzzy system is employed to make an optimum decision in terms of bandwidth and distance between the cores as inputs and output is the cost which is the total weighted distance of the tasks. The use of proposed fuzzy algorithm leads to improvement in power consumption and the performance of system.

Fuzzy systems have already been utilized in NoC [65-67]. In [65], the authors present an adaptive routing algorithm where fuzzy logic system is employed to avoid congestion in NoC. They use fuzzy system to generate uniform traffic over the routers which have extra capacity. In the proposed method path decision making is based on the fuzzy system to make links of NoC less congested through distributing the traffic uniformly. Investigators in [66], propose an adaptive routing method to select the router's output port based on the fuzzy system in NoC. The goal of this research is delay and power reduction. Fuzzy logic is used for flow regulation in NoC according to the chip multiprocessors to improve the performance by using the network resources better [67]. In this method, decision making is done completely dynamically based on the traffic as well as the condition of interconnection network.

To the best of our knowledge the fuzzy logic system has not been used in heuristic mapping problem in NoC. The advantages of fuzzy rules in decision making motivated us to suggest a heuristic mapping algorithm based on fuzzy logic rules. The suggested approach takes advantage of fuzzy system to obtain the optimum mapping of the tasks onto the cores with optimum number of routers in NoC architectures as well as takes advantage of the fuzzy concept to have more efficient evaluation in NP-hard problems. The suggested method in Section 4.1 is modified by a fuzzy-based objective function. All steps of this algorithm are the same as the previous one except for the evaluation of solutions. In this method, the solutions are evaluated by fuzzy sets and logics. This method of evaluation considers fuzzy sets for bandwidth and distance and assigns a membership function to each of them. Then fuzzy rules are defined to determine fuzzy cost of mapping based on the fuzzified bandwidth and distance values of each pair of cores. Then the obtained fuzzy mapping cost is defuzzificated. The defuzzification is done for all pairs of cores of the NoC.

The following relations are considered for a pair of cores in a NoC [5],

- Relation 1: Considering a fixed value of bandwidth between a pair of cores, if the distance of cores is increased, then, power consumption and latency of the NoC are increased.
- Relation 2: Considering a fixed distance between a pair of cores, if the bandwidth of cores is increased, then, power consumption of the NoC is increased.
- Relation 3: Considering a fixed distance between a pair of cores, if the bandwidth of cores is increased, then, latency of the NoC is decreased.

Based on the above relations, a new criterion named “Cost of Mapping” (CoM) is defined for each pair of cores in NoC which is depended on distance and bandwidth of the cores. Conceptually, CoM reflects power consumption and latency together. From the above-mentioned relations it is concluded that if a pair of tasks with high bandwidth are mapped to the cores having less distances, the tasks will have less CoM.

4.2.3 Bi-objective Fuzzy-based Meta-heuristic Algorithm

Besides links, a significant portion of power of NoCs is consumed in routers. Another contribution of this thesis is to find the optimum number of routers using fuzzy logic algorithm. This approach along with optimal layout gained in previous step can reduce the power of NoCs. Conclusively, proposed fuzzy solution is used to minimize the power dissipation by tasks to cores assigning and finding the optimum number of routers in NoC. Thus, the physical links power consumption, router's power dissipation and consequently, the total power of NoC are reduced.

As mentioned, number of router resources has great impact on power dissipation. Thus, suggested approach reduces the power consumption due to number of router reduction in NoCs. This algorithm uses GA and SA same as the previous algorithms with a bi-criteria objective function. This function targets to minimize the sum of weighted distances of tasks and the number of routers using fuzzy logic and fuzzy sets for bandwidth, distance and number of routers. In the other word, the pervious methods, which are described in 4.1 and 4.2, are modified in this section by a fuzzy-based bi-criteria objective function. All steps of this method are the same as the meta-heuristic algorithm except for evaluation of solution. Each solution generated by the bi-objective fuzzy-based algorithm, is evaluated by a fuzzy bi-criteria objective function which consists of two criterion, cost of mapping (CoM) and cost of router (CoR). These cost functions are calculated by fuzzy linguistic IF-THEN rules and then the weighted sum of CoM and CoR is considered as the bi-criteria objective function of the solution.

As mentioned above, the aim of the meta-heuristic and fuzzy-based meta-heuristic are to reduce CoM to improve the power consumption in NoCs. In other words, these two methods are trying to find the task to core mapping with minimum CoM. On the other hand, the bi-objective fuzzy-based meta-heuristic algorithm considers two objectives named reducing CoM and CoR simultaneously. Due to the fact that this method compromises the reduction of CoM to decrease CoR, its improvement in CoM might not be as good as reduction of CoM in two other methods. This issue arises because the CoR of each solution is not depended to its CoM but it is highly depended to the topology.

REFERENCES

[1] International Technology Roadmap for Semiconductors (ITRS), http://www.itrs.net/, retrieved on 2011.

[2] Marculescu R., Ogras U.Y., Peh L., Jerger N. E., & Hoskote Y., “Outstanding research problems in NoC design: system, microarchitecture, and circuit perspectives”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 1, pp. 3- 21, 2009.

[3] Benini L., & De Micheli G., “Networks on Chips: A new SoC paradigm”, *IEEE Computer*, vol. 35, no. 1, pp. 70-78, 2002.

[4] De Micheli G., Seiculescu C., Murali S., Benini L., Angiolini F., & Pullini A., “Networks on Chips: from research to products”, in Proc. *Design Automation Conference (DAC)*, pp. 300-305, 2010.

[5] Srinivasan K., Chatha K. S., & Konjevod G., “Linear programming based techniques for synthesis of Network-on-Chip architectures”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 4, pp. 407–420, 2006.

[6] Postman J., Krishna T., Edmonds C., Peh L., & Chiang P., “SWIFT: A low-power Network-on-Chip implementing the token flow control router architecture

with swing-reduced interconnects", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 8, pp. 1432–1446, 2013.

[7] Reehal G., & Ismail M., "A systematic design methodology for low-power NoCs", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 12, pp. 2585-2595, 2014.

[8] Kulkarni M., & Agrawal V., "Energy source lifetime optimization for a digital system through power management", in Proc. *43rd Southeastern Symposium on System Theory*, pp. 73-78, 2011.

[9] Svensson C., "Optimum voltage swing on on-chip and off-chip interconnect", *IEEE Journal of Solid-State Circuits*, vol. 36, no. 7, pp. 1108-1112, 2001.

[10] Wei L., Chen Z., Johnson M., Roy K., & De V., "Design and optimization of dual-threshold circuits for low voltage low power applications" , *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 7, no. 1, pp. 6-24, 1999.

[11] Shin D., Kim W., Kwon S., & Han T. H., "Communication-aware VFI partitioning for GALS-based Networks-on-Chip", *Design Automation for Embedded Systems*, vol. 15, no. 2, pp. 89-109, 2011.

[12] Moyer B., "Low power design for embedded processors", Proc. *of the IEEE*, vol. 89, no. 11, pp. 1576-1587, 2001.

[13] Snowdon D. C., Ruocco S., & Heiser G., “Power management and dynamic voltage scaling: myths and facts”, in Proc. *Workshop on Power Aware Real-Time Computing*, pp. 1-7, 2005.

[14] Benini L., Bogliolo A., & De Micheli G., “A survey of design techniques for system level dynamic power management”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 8, no. 3, pp. 299-316. 2000.

[15] Arelakis A., & Stenstrom P., “SC2: A statistical compression cache scheme”, in Proc. *41st Annual International Symposium on Computer Architecture*, pp. 145-156, 2014.

[16] Anagnostopoulos I., Bartzas A., Filippopoulos I., & Soudris D., “High-level customization framework for application-specific NoC architectures”, *Design Automation for Embedded Systems*, vol. 16, no. 4, pp. 339-361, 2012.

[17] Palesi M., Ascia G., Fazzino F., & Catania V., “Data encoding schemes in Network on Chip”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 5, pp. 774-786, 2011.

[18] Stan M. R., & Burleson W. R, “Limited-weight codes for low power I/O”, in Proc. *International Workshop on Low Power Design*, pp. 209-214, 1994.

[19] Stan M. R. & Burleson W. P., “Bus-invert coding for low- power I/O”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 3, pp. 49-59, 1995.

[20] Kim K. W., Baek K. H., Shanbhag N., Liu C. L., & Kang S., “Coupling driven signal encoding scheme for low power interface design”, in Proc. *International Conference on Computer Aided Design (ICCAD),* pp. 318-321, 2000.

[21] Benini L., De Micheli G., Macii E., Poncino M., & Quer S., “System level power optimization of special purpose applications: the beach solution”, in Proc. *International Symposium on Low Power Electronics and Design*, pp. 24-29, 1997.

[22] Taassori M., & Hessabi S., “Low power encoding in NOCs based on coupling transition avoidance”, in Proc. *Digital Systems Design, Architectures, Methods and Tools*, pp. 247-254, 2009.

[23] Mamidipaka M. N., Hirschberg D. S., & Dutt N. D., “Adaptive low power address encoding techniques using self-organizing lists”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 5, pp. 827-834, 2003.

[24] Cheng W. C., & Pedram M., “Power-optimal encoding for a DRAM address bus”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 10, no. 2, pp. 109-118, 2002.

[25] Lee K., & Lee S. J., “SILENT: serialized low energy transmission coding for on-chip interconnection networks”, in Proc. *International Conference on Computer Aided Design (ICCAD)*, pp. 448-451, 2004.

[26] Benini L., Macii A., Macii E., Poncino M., & Scarsi R., “Architectures and synthesis algorithms for power-efficient bus interfaces”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 9, pp. 969-980, 2000.

[27] Lv T., Henkel J., Lekates H., & Wolf W., “A dictionary based en/decoder scheme for low power data buses”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 5, pp. 943-951, 2003.

[28] Brahmbhatt A. R., Zhang J., Wu Q., & Qiu Q., “Low power bus encoding using adaptive hybrid algorithm”, in Proc. *Design Automation Conference (DAC)*, pp. 987-990, 2006.

[29] Benini L., & De Micheli G., “Networks on Chips: technology and tools”, *Murgan Kufmann Publishers*, 2006.

[30] Jafarzadeh N., Palesi M., Khademzadeh A., & Afzali-Kusha A., “Data encoding techniques for reducing energy consumption in Network-on-Chip”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 3, pp. 675-685, 2014.

[31] Jalabert A., Murali S., Benini L., & De Micheli G., “×pipes compiler: A tool for instantiating application specific Networks on Chip”, in Proc. *Design, Automation and Test in Europe (DATE)*, pp. 884–889, 2004.

[32] Chen X., & Peh L., “Leakage power modeling and optimization in interconnection networks”, in Proc. *International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 90–95, 2003.

[33] Chatha K.S., Srinivasan K., & Konjevod G., “Automated techniques for synthesis of application-specific Network-on-Chip architectures”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 8, pp. 1425-1438, 2008.

[34] Hu J., & Marculescu R., “Energy-aware mapping for tile-based NoC architectures under performance constraints”, in Proc. *Asia and South Pacific Design Automation*, pp. 233–239, 2003.

[35] Hu J., & Marculescu R., “Energy and performance aware mapping for regular NoC architectures”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 4, pp. 551-562, 2005.

[36] Murali S., & De Micheli G., “Bandwidth-constrained mapping of cores onto NoC architectures”, in Proc. *Design, Automation and Test in Europe (DATE)*, pp. 896–901, 2004.

[37] Murali S., Coenen M., Radulescu A., Goossens K., & De Micheli G., "A methodology for mapping multiple use-cases onto Networks on Chips", in Proc. *Design, Automation and Test in Europe (DATE)*, pp. 118–123, 2006.

[38] Hu J., & Marculescu R., "Exploiting the routing flexibility for energy/performance aware mapping of regular NoC architectures", in Proc. *Design, Automation and Test in Europe (DATE)*, pp. 688-693, 2003.

[39] Srinivasan K., & Chatha K. S., "A technique for low energy mapping and routing in Network-on-Chip architectures", in Proc. *International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 387-392, 2005.

[40] Sahu P. K., Shah T., Manna K., & Chattopadhyay S., "Application mapping onto mesh-based Network-on-Chip using discrete particle swarm optimization", *IEEE Transactions on Very Large Scale Integration (VLSI)* Systems, vol. 22, no. 2, pp. 300-312, 2014.

[41] Benini L., "Application specific Network-on-Chip design", in Proc. *Design, Automation and Test in Europe (DATE)*, pp. 491-495, 2006.

[42] Murali S., Benini L., & De Micheli G., "Mapping and physical planning of Networks-on-Chip architectures with quality-of-service guarantees", in Proc. *Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 27-32, 2005.

[43] Srinivasan K., Chatha K. S., & Konjevod G., "An automated technique for topology and route generation of application specific on-chip interconnection networks", in Proc. *International Conference on Computer Aided Design (ICCAD),* pp. 231-237, 2005.

[44] Srinivasan K., Chatha K. S., & Konjevod G., "Linear programming based techniques for synthesis of Network-on-Chip architectures", in Proc. *International Conference on Computer Design: VLSI in Computers and Processors, (ICCD),* pp. 422-429, 2004.

[45] Chatha K. S., Srinivasan K., & Konjevod G., "Automated techniques for synthesis of application-specific Network-on-Chip architectures", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 8, pp. 1425-1438, 2008.

[46] Murali S., & De Micheli G., "SUNMAP: A tool for automatic topology selection and generation for NoCs", in Proc. *Design Automation Conference (DAC),* pp. 914-919, 2004.

[47] He O., Dong S., Jang W., Bian J., & Pan D., "UNISM: Unified scheduling and mapping for general Network on Chip", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 8, pp. 1496-1509, 2012.

[48] Ogras U. Y., Bogdan P., & Marculescu R., "An analytical approach for Network on Chip performance analysis", *IEEE Transactions on Computer-*

Aided Design of Integrated Circuits and Systems, vol. 29, no. 12, pp. 2001-2013, 2010.

[49] Kiasari A. E., Lu Z., & Jantsch A., “An analytical latency model for Network on Chip”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 1, pp. 113-123, 2013.

[50] Paul G., “An efficient implementation of the robust tabu search heuristic for sparse quadratic assignment problems”, *European Journal of Operational Research*, pp. 215-218, 2011.

[51] Paul G., “Comparative performance of tabu search and simulated annealing heuristics for the quadratic assignment problem”, *Operations Research Letters*, pp. 577-581, 2010.

[52] Czapinski M., “An effective parallel multi start tabu search for quadratic assignment problem on CUDA platform”, *Journal of Parallel and Distributed Computing*, pp. 1461-1468, 2013.

[53] Chaovalitwongse W., Pardalos P. M., & Prokoyev O. A., “A new linearization technique for multi-quadratic 0-1 programming problems”, *Operations Research Letters*, vol. 32, no. 6, pp. 517-522, 2004.

[54] He X., Chen A., Chaovalitwongse W., & Liu H., “An improved linearization technique for a class of quadratic 0-1 programming problems”, *Optimization Letters*, vol. 6, no. 1, pp. 31-41, 2012.

[55] Bertozzi D., Jalabert A., & Murali S., “NoC synthesis flow for customized domain specific multiprocessor systems-on-chip”, *IEEE Transactions on Parallel and Distributed Systems,* vol.16, no. 2, pp.113-129, 2005.

[56] Xpress, Available online at: http://www.fico.com/en/products/fico-xpress-optimization-suite.

[57] Murali S., & De Micheli G., “Bandwidth-constraint mapping of cores onto NoC architectures”, in Proc. *Design, Automation and Test in Europe (DATE),* pp. 896–901, 2004.

[58] Ascia G., Catania V., & Palesi M., “Multi objective mapping for mesh based NoC architectures”, in Proc. *Hardware/Software Codesign and System Synthesis*, pp. 182–187, 2004.

[59] Leary G., Srinivasan K., Mehta K., & Chatha K., “Design of Network-on-Chip architectures with a genetic algorithm-based technique”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 5, pp. 674-687, 2009.

[60] Soumya J., & Chattopadhyay S., “Application specific Network on Chip synthesis with flexible router placement”, *Journal of Systems Architecture*, vol. 59, no. 7, pp. 361–371, 2013.

[61] Arjomand M., Amiri H., & Sarbazi H., “Efficient genetic based topological mapping using analytical models for on-chip networks”, *Journal of Computer and System Sciences*, vol. 79, pp. 492–513, 2013.

[62] Gen M., & Cheng R., “Genetic algorithms and engineering design”, *Wiley*, New York, 1997.

[63] Gen M., & Cheng R., “Genetic algorithms and engineering optimization”, *Wiley*, New York, 2000.

[64] Kirkpatrick S., Gelatt Jr. C. D., & Vecchi M.P., “Optimization by simulated annealing”. *Science*, vol. 220, no. 4598, pp. 671-680, 1983.

[65] Ebrahimi M., Tenhunen H., & Dehyadegari M., “Fuzzy-based adaptive routing algorithm for Networks on-Chip”, *Journal of Systems Architecture*, vol. 59, no. 7, pp. 516–527, 2013.

[66] Ascia G., Palesi M., & Catania V., “An adaptive output selection function based on a fuzzy rule base system for Network on Chip”, in Proc. *Digital System Design (DSD)*, pp. 505-512, 2013.

[67] Yao Y., & Lu Z., “Fuzzy flow regulation for Network on Chip based chip multiprocessors systems”, in Proc. *Design Automation Conference (DAC)*, pp. 343-348, 2014.

APPENDIX

Appendix A: List of Journal Publications

- M. Taassori, et al., “OPAIC: An optimization technique to improve energy consumption and performance in application specific network on chips”, *Measurement*, vol. 74, pp. 208-220, 2015.

- M. Taassori, et al., “MFLP: A Low Power Encoding for on Chip Networks”, *Design Automation for Embedded Systems*, DOI: 10.1007/s10617-015-9170-0, 2015.

- M. Taassori, et al., “Fuzzy-based Mapping Algorithms to Design Networks-on-Chip”, *Journal of Intelligent and Fuzzy Systems*, 2016. *(Accepted)*

- M. Taassori, et al., “Power-aware Meta-heuristic Core Mapping Approaches for Network on Chips”, *International Journal of Scientific & Engineering Research*, vol. 6, no. 9, pp. 834-837, 2015.