

# IMPROVED EXTENDED XY ON-CHIP ROUTING IN DIAMETRICAL 2D MESH NoC

Prasun Ghosal and Tuhin Subhra Das

Bengal Engineering and Science University, Shibpur  
Howrah 711103, INDIA

prasun@ieee.org, tuhinbcrec@gmail.com

## ABSTRACT

*Network-on-Chip (NoC) is a new approach for designing the communication subsystem among IP cores in a System-on-Chip (SoC). NoC applies networking theory and related methods to on-chip communication and brings out notable improvements over conventional bus and crossbar interconnections. NoC offers a great improvement over the issues like scalability, productivity, power efficiency and signal integrity challenges of complex SoC design. In an NoC, the communication among different nodes is achieved by routing packets through a pre-designed network fabric according to some routing algorithm. Therefore, architecture and related routing algorithm play an important role to the improvement of overall performance of an NoC. A Diametrical 2D Mesh routing architecture has the facility of having some additional diagonal links with simple 2D Mesh architecture. In this work, we have proposed a Modified Extended 2D routing algorithm for this architecture, which will ensure that a packet always reaches the destination through the possible shortest path, and the path is always deadlock free.*

## KEYWORDS

*NoC routing, Diametrical 2D mesh routing, On-chip communication, Extended XY routing*

## 1. INTRODUCTION

Network-On-Chip (NoC) [2] is an emerging technology for designing future complex system-on-chip (SoC) [4]. It brings a notable improvement over conventional Bus and Crossbar interconnection. A bus based system has lot of limitation. First, a global bus has a large capacity load for the bus drivers. In turn, this implies large delays and huge power consumption [6] as signal propagation in wire across the chip requires multiple clock cycle. Again on-chip bus allows only one communication transaction at a time according to the arbitration result thus the average communication bandwidth of each processing element is in inverse proportion to the total number of IP cores in a system. This character makes a bus-based architecture inherently non-scalable for a complex system.

In NoC architecture traditional bus structure is replaced with a network which is a lot similar to the Internet. Data communications between segments of chip are packetized and transferred through the network. The network consists of wires and routers. Processors, memories and other IP-blocks are connected to routers. A high level of parallelism is achieved in NoC, because all links in the NoC can operate simultaneously on different data packets. Therefore, as the complexity of integrated systems keeps growing, a NoC provides enhanced performance (such as throughput, power consumption) and better scalable architecture in comparison with previous communication architectures (e.g., dedicated point-to-point signal wires, shared buses, or segmented buses with bridges).

### 1.1. ROUTING ON NOC

In the network design of NoC the most essential things are network topology and its related routing algorithm. This on-chip network should have both low latency and high bandwidth communication fabrics that efficiently support a diverse variety of workloads. The algorithms must be designed in such a way that they offer large parallelism and can hence utilize the potential of NoC.

Routing on NoC is similar to routing on any network. A routing algorithm determines how the data is routed from sender end to receiver end. In NoC, routing algorithm should be implemented by simple logic and require size of data buffers should be minimal. Network topologies [8] [9] [10] and properties may be application specific [11].

### 1.2. GENERAL PROBLEM ON NOC ROUTING

There are couples of requirements that every Network on Chip implementation has to achieve. Performance requirements are small latency [5] [12], guaranteed throughput [12], path diversity sufficient transfer capacity and low power consumption [6]. Architectural requirements are scalability, generality and programmability. Fault and distraction tolerance as well as valid operation are major on Quality of Service.

However, selecting an appropriate topology [8] [9] [10] is one of the most critical decisions in the design of an interconnection network because it is related to some performance metrics such as the network's zero load latency and its capacity [7], and directly influences implementation costs, both in terms of the on-chip resources and implementation complexity.

## 2. BACKGROUND AND MOTIVATION

### 2.1. RELATED WORKS

2D Mesh is a very popular well known topology use in Network on Chip due to its facility in implementation, simplicity of the XY routing strategy and the network scalability. But, 2D Mesh has some disadvantages such as long network diameter as well as energy inefficiency because of the extra hops. Due to this, in this work, we use a proposed novel NoC topology called Diametrical 2D Mesh [1]. Proposed architecture reduces the network diameter [8] [12] 50 percent compared to 2D mesh by using 8 extra bypass diametrical channel. It reduces the network average latency [12] and power consumption [6] considerably.

### 2.2. DIAMETRICAL 2D MESH ARCHITECTURE

The Diametrical 2D Mesh architecture can be considered as a collection of four sub-networks (as show by shaded sky color in below pictures), where opposite corner of each sub-networks is connected by a diametrical link. Number of extra links in Diametrical 2D Mesh do not grow by the growth of IP cores, it is restricted to 8 extra links only .The main purpose of adding 8 extra links to 2D Mesh topology is the reduction of diameter in 2D Mesh when 2D Mesh is expanded by a large number of IP cores.

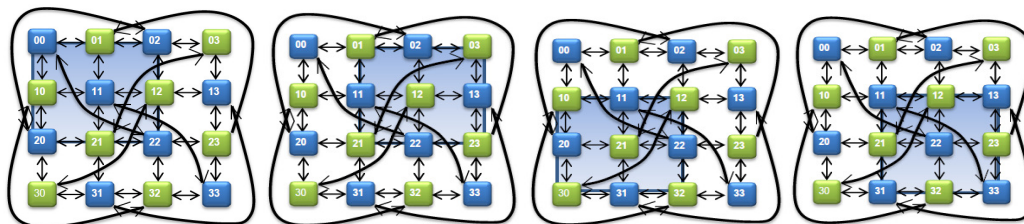


Fig. 1. A 4x4 Diametrical 2D Mesh Network

Some important parameters of an  $M \times M$  Diametrical 2D Mesh are given below

Total number of nodes= $M \times M$

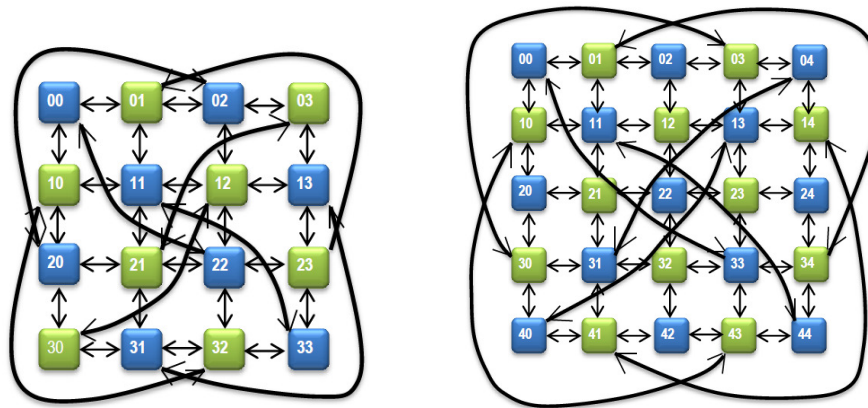
Diameter= $M+1$

Bisection Width= $M+8$

Nodes degree=Maximum 6, Minimum 4

### 2.3. DIAMETRICAL 2D MESH ADDRESSING

A position of a node in Diametrical 2D Mesh is represented by two parts X and Y, where X indicates the number of row position and Y indicates the number of column position in the network. The number of bits in X and Y parts are determined by the number of rows and columns in Diametrical 2D Mesh. For example, in a 16 IP cores, Diametrical 2D Mesh has 4 nodes in each rows and 4 ones in each column, so we need total 4 bits to represent a node position, where 2 bits show the number of row, X, and another 2 bits indicate the number of column, Y. Also, for 25 IP cores, Diametrical 2D Mesh has 5 nodes in each row and column. Therefore, both X part bits and Y part need 3bits to represent each part. This means that total 6 bits addresses are used to label the Diametrical 2D Mesh with 25 IP cores.



(a)Network 16 IP cores

(b) Network 25 IP cores

Fig. 2. Diametrical 2D Mesh network

### 2.4. DRAWBACKS IN EXISTING WORKS

But the existing algorithm called Extended XY Routing algorithm1 have some drawbacks like sometimes the packet never reaches to its destination (because of live-lock) and sometime packet follows XY path though there exists a diametrical path. So, existing algorithm is not able to totally exploit the benefit of diametrical path. But our proposed deterministic routing ensures that the packet will always reach the destination through the shortest path and it is deadlock free[13] as no circular waiting path is generated from the proposed routing algorithm.

### 3. PROBLEM DESCRIPTION

Assume a Diametrical 2D Mesh consisting of 25 IP cores (see Fig. 3(a)) having 5 nodes in each row and each column. We illustrate two cases, where in case1 message never reaches to its destination(shown by red color path of Fig3.(a)) and in case2 message is not able to use the shortest path though there exists a shortest path(shown by yellow color path of Fig.3(a)).

Case 1: If we consider a packet whose current node is (3,0) and destination node is (4,4),then according to the existing Extended XY algorithm [1] packet will move through diametrical path

not follow simple XY routing path and it will reach to node (0,3). As  $X_{offset} = (4-3) = 1$  (using  $X_{offset} = X_{dest} - X_{current}$ ) and  $Y_{offset} = (4-0) = 4$  (using  $Y_{offset} = Y_{dest} - Y_{current}$ ) and  $d=5$  (using  $d = \text{square root of } 25$ ) and so it does not meet the necessary criteria for XY Routing as described in [1], which is  $X_{offset} + Y_{offset} \leq (d-1)$ , rather it will follow diametric path and will reach to node (0,3). At the node (0,3),  $X_{offset} = (4-0) = 4$  and  $Y_{offset} = (4-3) = 1$ , so packet at this node will again follow diametric path as condition  $X_{offset} + Y_{offset} \leq (d-1)$  becomes false and will reach to node (3,0). Thus a live-lock will appear and the packet will never reach to destination node (4,4).

Case 2: Again, if we consider the current node is (3,0) and destination node is (0,4), then  $X_{offset} = (0-3) = -3$  and  $Y_{offset} = (4-0) = 4$  and  $d = 5$  (using  $d = \text{square root of } 25$ ) and according to the existing extended XY algorithm [1] packet will move through XY path as  $X_{offset} + Y_{offset} = 1$  which is less than  $(d-1)$ , where  $d=5$ . But it should follow the existing diametrical path which will carry the packet to the node (0,3) in terms of utilizing proper facility of diametrical path and to reduce the number of hop count.

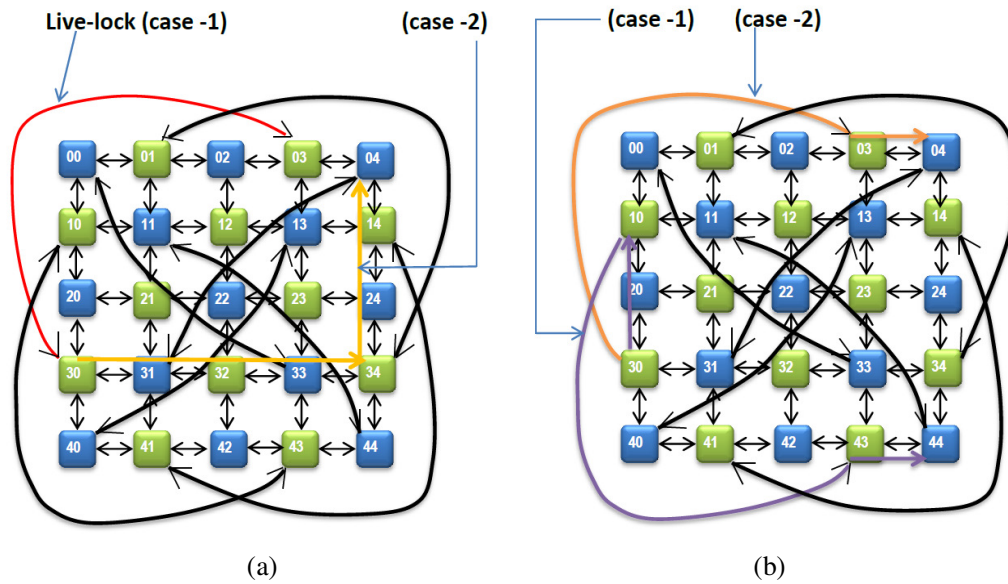


Fig. 3. Diametrical 2D Mesh network of 25 IP cores

#### 4. PROPOSED MODIFIED EXTENDED XY ROUTING

We propose a Modified extended XY routing for Diametrical 2D mesh Topology. This routing algorithm is very similar to Extended XY routing and XY routing, Modified Extended XY is very simple due to simple addressing scheme and structural topology. Algorithm 1 shows the details of the routing pseudo code. We define  $X_{diff}$  and  $Y_{diff}$  values in the pseudo code, which are calculated as follows respectively:

$$X_{diff} = \text{difference between } x_{dest}, y_{curr} \quad (1)$$

$$Y_{diff} = \text{difference between } y_{dest}, y_{curr} \quad (2)$$

$$d = \sqrt{\text{number of nodes}} \quad (3)$$

where

$x_{dest}$  = X coordinate of destination node

$y_{dest}$  = Y coordinate of destination node

$x_{curr}$  = X coordinate of current node

$y_{curr}$  = Y coordinate of current node

$X_{diff}$  and  $Y_{diff}$  are the values indicating the number of rows and columns between a current and a destination node respectively. If both  $X_{diff}$  and  $Y_{diff}$  values are zero, it means that the current node

is the destination and the packet reaches the destination node. The flow of Modified Extended XY routing is described as given below:

- ❖ If a current node and a destination node are located in the same row or column or  $(X_{diff}+Y_{diff}) < d-1$ 
  - Then conventional XY routing is performed (condition 3 of pseudo code).
- ❖ Else
  - If  $X_{diff}$  and  $Y_{diff}$  both are greater than diff (condition 5 of pseudo code)
    - Then flits are forwarded by diameter channel.
  - Else
    - If  $X_{diff}$  is less than diff (condition 7 of pseudo code)
      - Then depending on some conditions as stated in pseudo code under condition 7 either  $X_{shift}$  routing or  $Y_{shift}$  routing or diameter channel is used for routing.
    - Else (condition 8 of pseudo code)
      - Then depending on some conditions as stated in pseudo code under condition 8 either  $X_{shift}$  routing or diameter channel is used for routing.

In the proposed algorithms, the following notations have been used.

- x+ indicates movement of it in positive direction of x axis (i.e. in eastward direction)
- x- indicates movement of it in negative direction of x axis (i.e. in westward direction)
- y+ indicates movement of it in positive direction of y axis (i.e. in northward direction)
- y- indicates movement of it in negative direction of y axis (i.e. in southward direction).

Also, in the pseudo code represented in Algorithm 1, C< # > represents the condition numbers. For example Condition (1) will be represented as C1.

#### 4.1. ILLUSTRATION OF ALGORITHM WITH EXAMPLE

We are discussing here the follow of the proposed algorithm with few cases.

**Case1:** If current node of packet is (3,0) and destination node is (4,4) of Fig3.(b). Then according to our proposed algorithm  $X_{diff} = 4-3 = 1$  and  $Y_{diff} = 4-0 = 4$ . Now as condition 3 (i.e.  $X_{diff}+Y_{diff} < (d-1)$  for  $d=5$ ) in pseudo code becomes false. It check for condition 5 which is not true as  $X_{diff} < diff$  (where  $diff=2+\text{number of rows not having any diametrical connection} = 2+1=3$ ). Now condition 7 becomes true as  $X_{diff} < diff$  is true. As  $x_{dest}$  is equal to  $d-1$ , so condition 7a becomes true and a  $Y_{shift}$  routing will take place which will move packet to node (2,0). At this node again a  $Y_{shift}$  routing will take place because of above reason and packet will move to node (1,0). Now  $X_{diff} = 4-1 = 3$  and  $Y_{diff} = 4 - 0 = 4$ . So condition 5 becomes true as value of diff is 3 and packet will follow the diametrical path to move node (4,3). At node (4,3) packet will follow simple XY routing and will reach to node (4,4) (see Table 2 Sl. no 8 and violet colour path in Fig.3(b)).

**Case2:** If current node of packet is (3,0) and destination node is (0,4) of Fig3.(b). Then according to our proposed algorithm  $X_{diff} = 3- 0 = 3$  and  $Y_{diff} = 4-0 = 4$ . So condition 3 (i.e.  $X_{diff} + Y_{diff} < (d - 1)$  for  $d = 5$ ) in pseudo code becomes false. Now condition 5 is true as both  $X_{diff}$  and  $Y_{diff}$  have value which is greater than or equal to diff (as  $diff=3$  here). So packet will follow diametrical path and hence move to node (0,3). At this node  $X_{diff} = 0-0 = 0$  and  $Y_{diff} = 4-3 = 1$ . So condition 3 becomes true and packet will follow simple XY routing and will reach to node (0,4) (see Table 2 Sl. no 7 and orange colour path in Fig.3(b)).

```

Input : Diametric 2D matrix of nodes, xdest, ydest, xcurr, ycurr
Output: Optimum routing path between source and destination nodes

Xdiff =difference between(xdest,xcurr);
Ydiff =difference between(ydest,ycurr);
Xoffset = xdest - xcurr;
Yoffset = ydest - ycurr;
d=size of diametric matrix having same X and Y size. /*(e.g. for 5 × 5 matrix size is 5)*;/

if(d>=4)
    diff=2+no of rows not having diametrical connection;
else
    diff=1;

C1: If(Xdiff == 0 and Ydiff == 0)
{
    channel=internal;
}
C2:Else
{
    C3: if(Xdiff == 0 or Ydiff == 0 or Xdiff+Ydiff<d-1)
    {
        XY routing;
    }
    C4: else
    {
        C5: If(Xdiff>= diff and Ydiff>= diff)
        {
            channel =diameter;
        }
        C6: Else
        {
            C7:If(Xdiff<diff){
                C7a: If(xdest == 0 or xdest == d-1)
                Yshift Routing;
                C7b: Elseif ((xcurr == 0 or xcurr == 1 or xcurr == d-2 or xcurr == d-1) and
                (ycurr == 0 or ycurr == 1 or ycurr == d-2 or ycurr == d-1))
                channel=diameter;
                C7c: Elseif (xcurr == 0 || xcurr == d-1)
                Xshift Routing;
                C7d: Else
                Yshift Routing ;
            }
            C8:Else
            {
                C8a: If(ydest == 0 || ydest == d-1)
                Xshift Routing;
                C8b: Elseif ((xcurr == 0 || xcurr == 1 || xcurr == d-2 || xcurr == d-1) &&
                (ycurr == 0 || ycurr == 1 || ycurr == d-2 || ycurr == d-1))
                channel=diameter;
                C8c: Else
                Xshift Routing ;
            }
        }
    }
}

```

Algorithm 1: Algorithm for proposed routing in diametrical 2D mesh NoC Algorithm

```
XY_Routing():  
  
    if(Ydiff !=0){  
        if(Yoffset>0) then  
            channel=x+;  
        else  
            channel=x-;  
    }else{  
        if(Xoffset>0) then  
            channel=y-;  
        else  
            channel=y+;  
    }  
}
```

Algorithm 2: Procedure for XY routing

```
Xshift_Route():  
  
    if(Yoffset>0) then  
        channel=x-;  
    else  
        channel=x+;
```

Algorithm 3: Procedure for X shift routing

```
Yshift_Route():  
  
    if(Xoffset>0) then  
        channel=y+;  
    else  
        channel=y-;
```

Algorithm 4: Procedure for Y shift routing

## 5. EXPERIMENTAL RESULTS

The proposed algorithm is implemented in a standard desktop environment running Linux operating system on a chipset with Intel Pentium processor running at 3 GHz using GNU GCC compiler. The important section of a sample run is shown below.

```
Give size of diametric matrix  
5  
give value of xdest 4  
give value of ydest 4  
give value of xcurr 0  
give value of ycurr 0  
current node is(0 0)  
current Xdiff is 4  
current Ydiff is 4  
After diametric routing current node is (3 3)  
current Xdiff is 1  
current Ydiff is 1
```

After xy routing current node is(3 4)  
 current Xdiff is 1  
 current Ydiff is 0  
 After xy routing current node is(4 4)  
 current Xdiff is 0  
 current Ydiff is 0  
 Packet has reached the destination

Table1 shows a comparison of few important parameters between Diametrical 2D Mesh and simple 2D Mesh and from the above table it is clear that number of extra links in Diametrical 2D Mesh does not grow by the growth of IP cores. Number of extra links constantly equals to 8. This link redundancy is decreasing gradually by the growth of IP cores. For example, the link redundancy in 16 IP cores 2D Mesh is 1/3 and in 25 IP cores 2D Mesh, this ratio is 1/5. Furthermore, the network diameter is decreased to 50% with any number of IP cores in comparison to 2D Mesh. This leads to less average hop count and network average latency and power consumption considerably.

No of IP cores		4 IP cores d=2	9 IP cores d=3	16 IP cores d=4	25 IP cores d=5	36 IP cores d=6	49 IP cores d=7
Number of links	D-2D Mesh	6	20	32	48	68	92
	2D Mesh	4	12	24	40	60	84
D-2D Mesh link Redundancy		33%	30%	25%	16%	11%	8%
Diameter	D-2D Mesh	1	2	3	4	5	6
	2D Mesh	2	4	6	8	10	12
D-2D Mesh Diameter Improvements		50%	50%	50%	50%	50%	50%

Table 1. A comparative study of few Interconnection parameters between D-2D Mesh and 2D Mesh. d=No of nodes in each side of Diametrical 2D Mesh, D-2D= Diametrical 2D Mesh

SL No.	Total Cores	Matrix Size(d)	Source Node	Destination Node	Routing Path	Hop Counts For	
						Modified Extended XY	Simple XY
1	25	5	00	23	00--d->33--xy->23	2	5
2	25	5	00	04	00--xy-->01--xy-->02--xy-->03--xy-->04	4	4
3	25	5	00	40	00--xy-->10--xy-->20--xy-->30--xy-->40	4	4
4	25	5	00	14	00--d->33--xy-->34--xy-->24--xy-->14	4	5
5	25	5	21	04	21--yshift-->31--d-->04	2	5
6	25	5	21	44	21--yshift-->11--d-->44	2	5
7	25	5	30	04	30--d-->03--xy-->04	2	7
8	25	5	30	44	30--yshift-->20--yshift-->10--d-->43--xy-->44	4	5
9	25	5	44	00	44--d-->11--xy-->10--xy-->00	3	8
10	25	5	44	20	44--d-->11--xy-->10--xy-->20	3	6

Table 2. Routing path with required hope count of a matrix size 5. d=diametrical routing, xy=simple xy routing, y<sub>shift</sub>=Y<sub>shift</sub> routing of pseudo code.



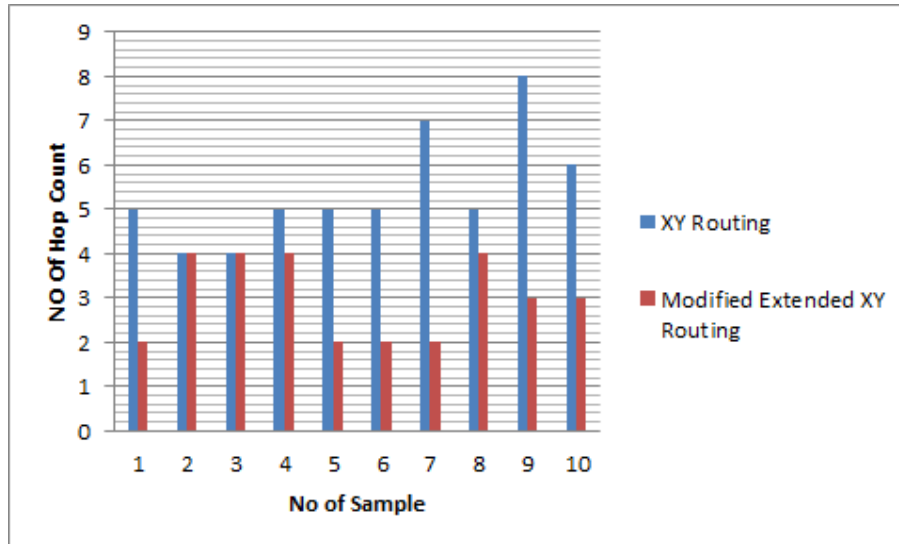


Fig4. A comparison of required number of hop count for XY and Modified Extended XY routing of a 5x5 network.

A comparison on number of hop count for different combination of source and destination node (which is given in Table2) for two different routing technique is showing in above plot (see Fig4). From above comparison it is clear that average hop count for diametrical xy routing is very much less(i.e. in general case (d-1)) compared to xy routing technique and require user routing time is also negligible (user routing time<0.005sec).

## 6. CONCLUSION

Diametrical 2D Mesh was proposed to overcome with the disadvantages of 2D Mesh architecture such as large network diameter and high power consumption. But the algorithm proposed in [1] was not able to adequately utilize the benefit of Diametrical 2D Mesh and in some cases the packet is not able to reach to its destination node. In our proposed algorithm those issues have been solved and the proposed deterministic algorithm is able to fully exploit the benefit of Diametrical 2D Mesh architecture. Our proposed algorithm is a deterministic one. So, future works may be to make the algorithm adaptive in order to address the issues like network congestion and faulty network issues.

SL No.	Total Cores	Matrix Size(d)	Source Node	Destination Node	Hop Count
1	4	2	00	11	1
2	4	2	00	10	1
3	9	3	00	11	1
4	9	3	00	20	2
5	9	3	00	22	2
6	16	4	00	11	2
7	16	4	00	03	3
8	16	4	00	13	3
9	16	4	00	33	3
10	25	5	00	22	2
11	25	5	00	04	4
12	25	5	00	44	3

13	36	6	00	23	4
14	36	6	00	33	3
15	36	6	00	05	5
16	36	6	00	55	3
17	49	7	00	34	2
18	49	7	00	44	3
19	49	7	00	06	6
20	49	7	00	66	3
21	64	8	00	77	3
22	64	8	00	34	6
23	64	8	00	07	7
24	81	9	00	88	3
25	81	9	00	03	3
26	81	9	00	43	7
27	81	9	00	08	8

Table 3. Shows required no of hop count to route from source 00 to different destination node of matrix size  $d=2,3, \dots, 9$ .

## REFERENCES

### Authors

**Dr. Prasun Ghosal** is currently associated with Bengal Engineering and Science University as an Assistant Professor. He has completed PhD (2011) from Bengal Engineering and Science University, and M.Tech. (2005) as well as B.Tech. (2002) in Radio Physics and Electronics from Institute of Radio Physics and Electronics, University of Calcutta, India. He is also an Honours Graduate (major in Physics) under University of Calcutta. He has received Young Scientist Research Award for the year 2010-11 from Indian Science Congress Association. He is also recipient of several Best paper awards. His research interests include VLSI Physical Design: Algorithms and Applications, Network on Chip: Architectures and Algorithms, Embedded Systems: Architectures and Applications, Quantum Circuits and Computing, Communication, Teleportation, and Cryptography, Application and development of different optimization algorithms for Social and Environmental Computing. He has contributed around 10 research articles in several peer reviewed international journals and more than 40 in peer reviewed international conferences. Besides a copyright application he has also contributed towards several book chapters. He has carried out several sponsored research projects funded by AICTE, DIT, MCIT, Govt. of India, IET etc.



**Tuhin Subhra Das** has received his M. Tech. in Information Technology from Bengal Engineering and Science University, Shibpur, India in 2012 and B. Tech. in Information Technology from Dr. B.C. Roy Engineering College, Durgapur, India in 2005. He is currently pursuing his research works towards the doctorate degree under the guidance of Dr. Prasun Ghosal in Bengal Engineering and Science University, India. His research interests include optimization of Network-On-Chip(NoC) architectures and related routing algorithms.

