

# Study of Communication Structural Relationship between Processors of Parallel and Distributed System

Rakesh Kumar Katare , Neha Singh\*, Aarti Pandey , Charvi Katare

Department of Computer Science, A.P.S. University, Rewa, Madhya Pradesh, India

## ABSTRACT

### Article Info

Volume 7, Issue 5

Page Number: 21-27

### Publication Issue :

September-October-2021

### Article History

Accepted : 10 Sep 2021

Published : 19 Sep 2021

Communication Complexity, Data Structures and its structural relationship are the key parameters for study of architecture in any interconnection Network. This paper reveals that the study of data structures in Perfect Difference Interconnection Network have some structural relationship between nodes or processors. In the Perfect Difference Network (PDN) with processors as a nodes and links between the processors or processing elements as an edge represents graphical Model for interconnection networks. The relationship between each processor is connected with some mapping functions to map data according to the topology of the PDN

**Keywords :** PDN, PDS, Interconnection Network, Data Structures, Structural Relationship.

## I. INTRODUCTION

This paper is focus on the study of structural relationship/structural mapping in the PDN. Perfect Difference Network is based on mathematical properties of Perfect difference set. In this we also represent a graphical model for this PDN to explore or analyze the relation of node/vertex in terms of mapping function according to topological behaviour of the Interconnection Network. Perfect difference Sets were first discussed in 1938 by J. Singer .Their formulation was in terms of points and lines in a finite projective Plane [1,2].The Perfect Difference Sets considered for being develop into a Interconnection Network mainly through works of Parahami, Behrooz and Rakov [3,4].They also compared PDNs and some of their derivatives with

hypercube and its other variants of interconnection networks for similar cost and performance. PDN are a robust high Performance interconnection networks for Parallel and distributed systems. A more exhaustive comparative study is done by Katare and et al., 2007, 2009[6,7] based on topological properties of hypercube and PDN. Further Katare and et al. [9] in 2013 study the circuits based architecture of PDN [9] in their research work by studying the link utilization of PDN and hypercube .The study of connectivity using AND and XOR logical operation on PDN for parallel and distributed systems is done by Bhardwaj and Katare,2018[15]. They show that the usefulness of three eloquent and efficient incidence, circuit and path matrix on PDN architecture.

### 1. Processor Matrix-

To represent a graph in a useful and convenient way is a matrix. As we know that Matrices lend themselves for easily calculations and manipulations. Many results of matrix algebra can be readily applied to view the study of structural algebraic property.

For this we define a processor matrix  $P = [p_{ij}]$  where  $n$  rows correspond to the  $i^{th}$  processor and the  $n$  columns correspond to the  $j^{th}$  relationship between processor as follows:

$$[p_{ij}] = \begin{cases} 1, & \text{when } i\text{th processor is directly} \\ & \text{incident on } j\text{th processor} \\ 0, & \text{otherwise} \end{cases}$$

Such a matrix  $P$  is called vertex to vertex Processor Incidence matrix or simply a Processor matrix. The Processor matrix contains only two elements 0 and 1 in Interconnection network. Such a matrix is also called a binary Processor-matrix or (0-1) P-matrix.

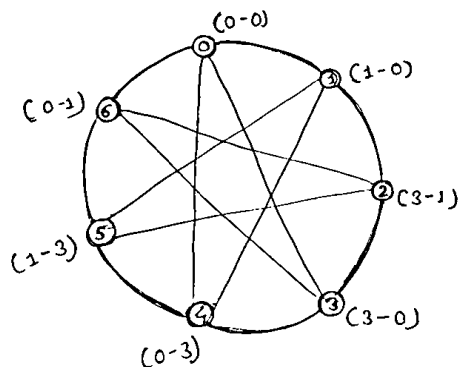


Figure 1 : PDN for  $\Delta= 3$

Table 1 : Processor matrix for PDN  $\Delta= 3$

	P0	P1	P2	P3	P4	P5	P6
P0	1	1	0	1	1	0	1
P1	1	1	1	0	1	1	0
P2	0	1	1	1	0	1	1
P3	1	0	1	1	1	0	1
P4	1	1	0	1	1	1	0
P5	0	1	1	0	1	1	1
P6	1	0	1	1	0	1	1

Table 1 shows the communication connectivity of nodes in a Perfect Difference Network(PDN) where

each column of the matrix in '1' represents the relation of nodes and in '0' represents a relation/connectivity between nodes is a vector used for logical operation for the investigation of the inter node connectivity of the network.

The following properties about the processor matrix are -

- 1) Since every vertex is connected to exactly four vertices, each rows and columns of  $P$  (including itself) has exactly five 1's.
- 2) In Processor matrix 1's represents the preserve of relation between processors.
- 3) In the Processor matrix 0's represents the absence of direct relation between processors.

Rows and columns vectors are same which shows that the symmetry of connectivity between processors. Here we are assuming & considering the vector of a connectivity matrix as the value of a node. For example the vector of  $p_0$  processor is (1101101) so this value is the value of processor  $p_0$ .

**Theorem:** In an Interconnection networks the number of bits of edges in connectivity matrix shows number of vertices. For Complete graph in processor matrix number of bits of vertices in connectivity matrix shows the number of edges in connectivity matrix of vertices and edges.

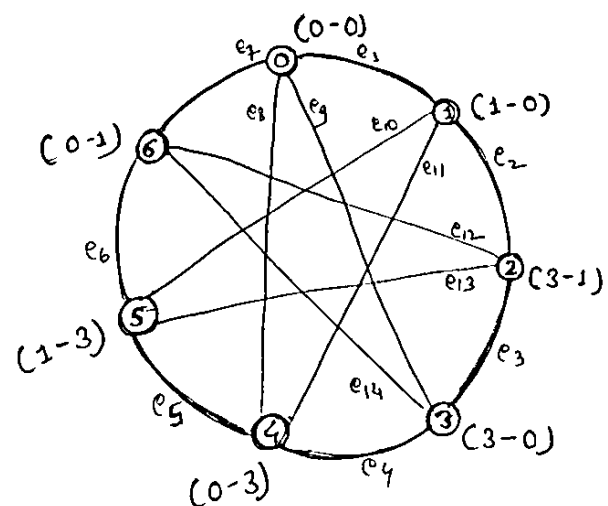


Figure 2.PDN for  $\Delta= 3$

**Proof:**

We proof this theorem by Edges to vertices Connectivity Matrix of processor. In the above figure2 is made for 7 node connectivity of PDN where each vector of edges has seven values that represent 7 nodes in a PDN.

	P0	P1	P2	P3	P4	P5	P6
E1	1	1	0	0	0	0	0
E2	0	1	1	0	0	0	0
E3	0	0	1	1	0	0	0
E4	0	0	0	1	1	0	0
E5	0	0	0	0	1	1	0
E6	0	0	0	0	0	1	1
E7	1	0	0	0	0	0	1
E8	1	0	0	0	1	0	0
E9	1	0	0	1	0	0	0
E10	0	1	0	0	0	1	0
E11	0	1	0	0	1	0	0
E12	0	0	1	1	0	1	0
E13	0	0	1	1	0	1	0
E14	0	0	0	1	0	0	1

**Table 2 :** Connectivity matrix of processor for 7 nodes

Here E1 has vector (1 1 0 0 0 0 0) that can be represent 7 nodes of PDN. Similarly E2,E3,E4,E5,E6,E7,E8,E9,E10,E11,E12,E13 and E14 edges vectors contains only 7 values.

Similarly values of vertices vector is represent the number of edges in a PDN. For example P1 has vector ( 1 0 0 0 0 1 1 1 0 0 0 0 0) contains 14 values that represents 14 edges of PDN. Same as we can proof it for all processor has total 14 values that represent 14 edges in a PDN

**Theorem:** The vector of connectivity matrix shows the topological or mathematical properties of the Interconnection Network.

**Proof:**

This theorem can be proved from the defined PDS ( $\pm 0, \pm 1, \pm 3$ ). For this we can derive all connectivity matrix from processor to processor.

	P0	P1	P2	P3	P4	P5	P6
P0	1	1	0	1	1	0	1
P1	1	1	1	0	1	1	0
P2	0	1	1	1	0	1	1
P3	1	0	1	1	1	0	1
P4	1	1	0	1	1	1	0
P5	0	1	1	0	1	1	1
P6	1	0	1	1	0	1	1

**Table 3 :** Connectivity Matrix of 7 Processors

Where the vectors of processors P0, P1, P2, P3, P4, P5, and P6 are represented PDS elements.

For example P0 has vector (1 1 0 1 1 0 1) derived from PDS (0-0, 1-0, 3-0, 0-3, 0-1). Similarly we can derive all processor vectors from given PDS.

P1 has Vector (1 1 1 0 1 1 0) = (0-0,1-0,3-1,0-3,1-3)

P2 has vector (0 1 1 1 0 1 1) = (1-0,3-1,3-0,1-3,0-1)

P3 has vector (1 0 1 1 1 0 1) = (0-0,3-1,3-0,0-3,0-1)

P4 has vector (1 1 0 1 1 1 0) = (0-0, 1-0, 3-0,0-3,1-3)

P5 has vector (0 1 1 0 1 1 1) = (1-0, 3-1,0-3,1-3,0-1)

P6 has vector (1 0 1 1 0 1 1) = (0-0,3-1, 3-0,1-3,0-1)

**3. Switching Functions:**

Switching Functions play an important step in the logical design of a modern digital machine (to minimize the Boolean function before implementing them). Suppose we build a logical circuit that gives the following function as follows:

$$P0 (P0+P1+P3+P4+P6) = P0P0+P0P1+P0P3+P0P4+P0P6$$

$$P1(P0+P1+P2+P4+P5) = P1P0+P1P1+P1P2+P1P4+P1P5$$

$$P2(P1+P2+P3+P5+P6) = P2P1+P2P2+P2P3+P2P5+P2P6$$

$$P3(P0+P1+P3+P4+P6) = P3P0+P3P2+P3P3+P3P4+P3P6$$

$$P4(P0+P1+P3+P4+P5) = P4P0+P4P1+P4P3+P4P4+P4P5$$

$$P5(P1+P2+P4+P5+P6) = P5P1+P5P2+P5P4+P5P5+P5P6$$

$$P6(P0+P2+P3+P5+P6) = P6P0+P6P2+P6P3+P6P5+P6P6$$

We implement an architecture that has minimal connectivity using the sum of product function. Functions are writing for all processors so that it can be known that the minimum number of switches required for connecting single processor. These relations is equivalent we have proved by using logical relations in the form of lemmas.

**Lemma 1:** For the processor P0 in Perfect Difference architecture the product of sum is equivalent to its product of sum that is operations on node in architecture are dual in nature.

$$P0(P0+P1+P3+P4+P6) = P0P0+P0P1+P0P3+P0P4+P0P6$$

Proof :

1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P0	P0	P0	P0	P0	6	1n
0	1	3	4	6	∩	∩	∩	∩	∩	∪7	(1∪
					P0	P1	P3	P4	P6	∪8	2∪
										∪9	3∪
										∪10	4∪
											5)
1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	1	0	1	1	0	1	0	1	1
0	1	1	0	1	0	0	0	0	0	0	0
1	0	1	1	1	1	0	0	1	0	1	1
1	1	1	1	0	1	1	1	1	1	1	1
0	1	0	1	1	0	0	0	0	0	0	0
1	0	1	0	1	1	0	0	0	1	1	1

$$\text{Lemma 2: } P1(P0+P1+P2+P4+P5) = P1P0+P1P1+P1P2+P1P4+P1P5$$

1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P1	P1	P1	P1	P1	6	1n
0	1	2	4	5	∩	∩	∩	∩	∩	∪7	(1∪
					P0	P1	P2	P4	P5	∪8	2∪
										∪9	3∪
										∪10	4∪
											5)
1	1	0	1	0	1	1	0	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1
0	1	1	0	1	0	1	1	0	1	1	1
1	0	1	1	0	0	0	0	0	0	0	0
1	1	0	1	1	1	1	0	1	1	1	1
0	1	1	1	1	0	1	1	1	1	1	1
1	0	1	0	1	0	0	0	0	0	0	0

$$\text{Lemma 3: } P2(P1+P2+P3+P5+P6) = P2P1+P2P2+P2P3+P2P5+P2P6$$

1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P2	P2	P2	P2	P2	6	1n
1	2	3	5	6	∩	∩	∩	∩	∩	∪7	(1∪
					P1	P2	P3	P5	P6	∪8	2∪
										∪9	3∪
										∪10	4∪
											5)
1	0	1	0	1	0	0	0	0	0	0	0
1	1	0	1	0	1	1	0	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1
0	1	1	0	1	0	1	1	0	1	1	1
1	0	1	1	0	0	0	0	0	0	0	0
1	1	0	1	1	1	1	0	1	1	1	1
0	1	1	1	1	0	1	1	1	1	1	1

$$\text{Lemma 4: } P3(P0+P2+P3+P4+P6) = P3P0+P3P2+P3P3+P3P4+P3P6$$

1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P3	P3	P3	P3	P3	6	1n
0	2	3	4	6	∩	∩	∩	∩	∩	∪7	(1∪
					P0	P2	P3	P4	P6	∪8	2∪
										∪9	3∪
										∪10	4∪
											5)
1	0	1	0	1	1	0	1	0	1	1	1
1	1	0	1	0	0	0	0	0	0	0	0
0	1	1	1	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	0	1	1	1
1	0	1	1	0	1	0	1	1	0	1	1
0	1	0	1	1	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1

**Lemma 5:**  $P_4(P_0+P_1+P_3+P_4+P_5) = P_4P_0+P_4P_1+P_4P_3+P_4P_4+P_4P_5$

1	1	1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---

1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P <sub>4</sub>	P <sub>4</sub>	P <sub>4</sub>	P <sub>4</sub>	P <sub>4</sub>	6	1n
0	1	3	4	5	n	n	n	n	n	U 7	(1 U
					P <sub>0</sub>	P <sub>1</sub>	P <sub>3</sub>	P <sub>4</sub>	P <sub>5</sub>	U 8	2 U
										U 9	3 U
										U 10	4 U
											5)
1	1	1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1	1	1
0	1	1	0	1	0	0	0	0	0	0	0
1	0	1	1	0	1	0	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	0	1	0	1	1	1	1
1	0	1	0	1	0	0	0	0	0	0	0

**Lemma 6:**  $P_5(P_1+P_2+P_4+P_5+P_6) = P_5P_1+P_5P_2+P_5P_4+P_5P_5+P_5P_6$

1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P <sub>5</sub>	P <sub>5</sub>	P <sub>5</sub>	P	P <sub>5</sub>	6	1n
1	2	4	5	6	n	n	n	5n	n	U 7	(1 U
					P <sub>1</sub>	P <sub>2</sub>	P <sub>4</sub>	P <sub>5</sub>	P <sub>6</sub>	U 8	2 U
										U 9	3 U
										U 10	4 U
											5)
1	0	1	0	1	0	0	0	0	0	0	0
1	1	1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1	1	1
0	1	1	0	1	0	0	0	0	0	0	0
1	0	1	1	0	1	0	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	0	1	0	1	0	1	1

**Lemma 7:**  $P_6(P_0+P_2+P_3+P_5+P_6) = P_6P_0+P_6P_2+P_6P_3+P_6P_5+P_6P_6$

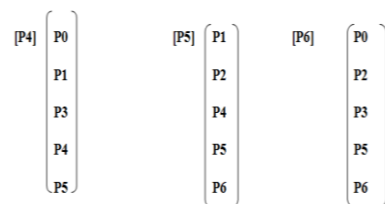
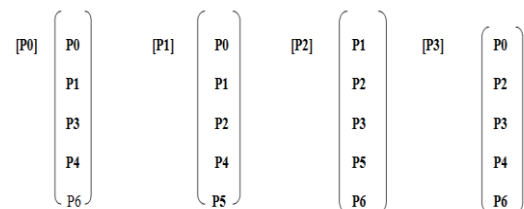
1	2	3	4	5	6	7	8	9	10		
P	P	P	P	P	P <sub>6</sub>	P <sub>6</sub>	P <sub>6</sub>	P <sub>6</sub>	P <sub>6</sub>	6	1n
0	2	3	5	6	n	n	n	n	n	U 7	(1 U
					P <sub>0</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>5</sub>	P <sub>6</sub>	U 8	2 U
										U 9	3 U
										U 10	4 U
											5)
1	0	1	0	1	1	0	1	0	1	1	1
1	1	0	1	0	0	0	0	0	0	0	0
0	1	1	1	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	0	1	1	1
1	0	1	1	0	0	0	0	0	0	0	0
0	1	0	1	1	0	1	0	1	1	1	1

**II. Communication Matrix of a Processor**

In an Interconnection network, by communication matrix we can derived how many processors are connected to and what patterns are used to connect them.

Here processor p0 in communication matrix shows that each processor connected to which processor. Processor p0 is connected to itself and also connected to p1, p3, p4, p6. Similarly

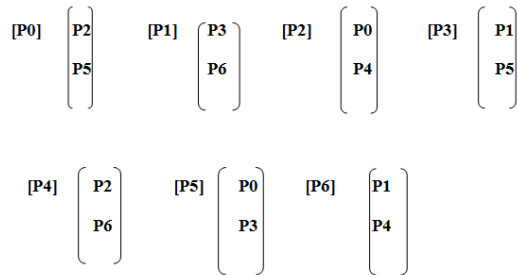
- processor p1 is connected with processors (p0, p1, p2, p4, p5)
- processor p2 is connected with processors (p1, p2, p3, p5, p6)
- processor p3 is connected with processors ( p0, p2, p3, p4, p6)
- processor p4 is connected with processors ( p0, p1, p3, p4, p5)
- processor p5 is connected with processors ( p1, p2, p4, p5, p6)
- processor p6 is connected with processors ( p0, p2, p3 , p5, p6)



**III. Processor Connectivity Residual Link**

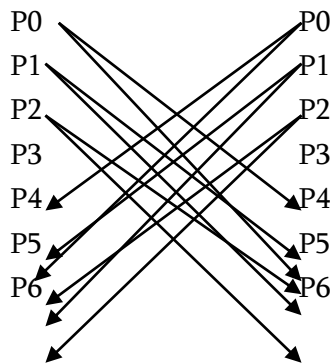
The chordal rings made in connectivity of processor is the links connecting consecutive nodes from i-1 to i+1 also known as Residual Ring links. While for non-consecutive nodes of processor connectivity from i to i ±s<sub>j</sub>(mod n) for 2 ≤ j ≤ Δ , are skip links or residual

link or residual chords. The residual link in connected nodes for  $i$  and  $i \pm s_j \pmod n$  is shown in figure: In the figure we see that  $p_0$  is not directly connected to  $p_2, p_5$  as their diameter of PDN is 2 and to make a connectivity in these processor we need a bi-relation. Similarly for the processor  $p_1, p_2, p_3, p_4, p_5, p_6$  of PDN with diameter 2 and its residual processors  $\{p_3, p_5\}, \{p_0, p_4\}, \{p_1, p_5\}$  and so on.



**IV. Bipartite Connectivity of PDN**

Here we are representing the diagonal connectivity of processors in a PDN.



Since each processor in this PDN is connected with each four processor so the degree of this PDN is 28, that is twice of the edges in PDN ( $2 \times 14$ ) and total no Processor in this PDN is 7 so total no of communication edges is twice of the processor  $= 2 \times 7 = 14$  out of which there are 7 chordal. So no of diagonal edges in PDN is the difference between communication edges in processor and chordal of PDN i.e.  $14 - 7 = 7$  (diagonal).  $PDS = \{\pm 0, \pm 1, \pm 3\}$

$$\begin{matrix} [0-0] & \begin{pmatrix} 0-0 \\ 1-0 \\ 3-0 \\ 0-3 \\ 0-1 \end{pmatrix} & [1-0] & \begin{pmatrix} 0-0 \\ 1-0 \\ 3-1 \\ 0-3 \\ 1-3 \end{pmatrix} & [3-1] & \begin{pmatrix} 1-0 \\ 3-1 \\ 3-0 \\ 1-3 \\ 0-1 \end{pmatrix} & [3-0] & \begin{pmatrix} 0-0 \\ 3-1 \\ 3-0 \\ 0-3 \\ 0-1 \end{pmatrix}
 \end{matrix}$$

$$\begin{matrix} [0-3] & \begin{pmatrix} 0-0 \\ 1-0 \\ 3-0 \\ 0-3 \\ 0-1 \end{pmatrix} & [1-3] & \begin{pmatrix} 1-0 \\ 3-1 \\ 0-3 \\ 1-3 \\ 0-1 \end{pmatrix} & [0-1] & \begin{pmatrix} 0-0 \\ 3-1 \\ 3-0 \\ 1-3 \\ 0-1 \end{pmatrix}
 \end{matrix}$$

The connectivity of node is generated by remainder theorem ( $R=N-D*Q$ ) in PDN which is formulised as:

$$\begin{aligned}
 p_i - p_j &= (\Delta^2 + \Delta) \pmod{(\Delta^2 + \Delta + 1)} \\
 p_i - p_j &= (0, 1, \dots, \Delta^2 + \Delta)
 \end{aligned}$$

$\pmod{(\Delta^2 + \Delta + 1)}$

Where  $R = N \pmod D$ , where  $Q=1$  is a constant value in processors,  $N = (0, 1, \dots, \Delta^2 + \Delta)$  and  $D$  is  $(\Delta^2 + \Delta + 1)$ .

By using this formula we can write above structural relation in following format

$$[0 \pmod 7] \begin{pmatrix} 0 \pmod 7 \\ 1 \pmod 7 \\ 3 \pmod 7 \\ 4 \pmod 7 \\ 6 \pmod 7 \end{pmatrix}$$

$$[0-7] \begin{pmatrix} 0-7*1 \\ 1-7*1 \\ 3-7*1 \\ 4-7*1 \\ 6-7*1 \end{pmatrix} = -7 \begin{pmatrix} -7 \\ -6 \\ -4 \\ -3 \\ -1 \end{pmatrix} = 0 \begin{pmatrix} 0 \\ 1 \\ 3 \\ 4 \\ 6 \end{pmatrix}$$

$$\begin{aligned}
 \text{Total links of PDN} &= \text{Chordal} + \text{Diagonal} \\
 &= (\Delta^2 + \Delta + 1) + (\Delta^2 + \Delta + 1) \\
 &= 2(\Delta^2 + \Delta + 1) \\
 \text{Missing Link of PDN} &= \Delta^2 + \Delta + 1 \\
 \text{Complete Graph of } \Delta=2 &= 3(\Delta^2 + \Delta + 1)
 \end{aligned}$$

## V. Conclusion

In this research we have investigate that the communication connectivity between the nodes or processors is a vector which is used for logical operation in an inter node connectivity of network. We have found that minimum three nodes are required to form a circuit to increase the communication efficiency. We also derive that no of bits in an edges of connectivity matrix in a processors represent the number of edges if our network is a complete graph. We also find that the vector matrix shows the topological properties in an inter connection network.

## VI. REFERENCES

- [1]. Singer,J."Perfect Difference Sets", Division of Mathematics ,Brooklyn College , Brooklyn,N.Y,1966
- [2]. Singer,J." A theorem in Finite Projective Geometry and Some Applications to Number Theory ". Trans American Mathematical Society ,Vol.43 ,PP 377-385,1938.
- [3]. Parhami,B., Rakov,M." Application of Perfect Difference Sets to the Design of efficient and Robust Interconnection Networks"
- [4]. Parhami,B., Rakov,M."Perfect Difference Networks and related Interconnection Structures for parallel and distributed Systems" IEEE transactions on Parallel and distributed systems,Vol.16,no. 8, august 2005,pp 714-724.
- [5]. Parhami,B., Rakov,M."Perfect Difference Networks and Graphs and their applications"2005.
- [6]. Katare ,R.K., and Chaudhari ,N.S. " A Comparative Study of hypercube and perfect difference network for parallel and distributed systems and its application to sparse linear system" Vladimir Journal of Computer Sciences,Vol2. Sandipani Academy,Ujjain,India,pp13-30,2007.
- [7]. Katare ,R.K., and Chaudhari ,N.S. " Study of Topological Property of Interconnection Networks and its mapping to Sparse Matrix Model ", Interconnection Journal of Computer Science and Applications,Technomathematics Research Foundation,Vol.6,N0.1,pp 26-39,2009.
- [8]. Deo. Narsingh," Graph theory with applications to engineering and Computer Sciences",PHI.2012
- [9]. Katare ,R.K.,et al." Study of link Utilization of Perfect Difference Network and Hypercube",2013.
- [10].Tiwari,S.,Katare,R.K." Study of Fabric Architecture Using Structural Pattern and Relation" IJTEMAS,Vol.IV,Issue IX,2015.
- [11].Tiwari,S.,Katare,R.K." A study of Intercoonection Network for parallel & distributed System",IJMITER,Vol.5,Issue 06,2017.
- [12].Bhardwaj,M. Katare,R.K.,"Study of connectivity using AND and XOR logical operatiuons on Perfect Difference Network for Parallel and Distributed Systems",IJEECS,Vol. 7,Issue 3,2018.

### Cite this article as :

Rakesh Kumar Katare, Neha Singh, Aarti Pandey, Charvi Katare, "Study of Communication Structural Relationship between Processors of Parallel and Distributed System", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 5, pp. 21-27, September-October 2021. Available at doi : <https://doi.org/10.32628/CSEIT21759>  
Journal URL : <https://ijsrcseit.com/CSEIT21759>