

SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

EDIA

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 11, Issue 5, May 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

## Impact Factor: 8.118

9940 572 462

6381 907 438

 $\bigcirc$ 

🛛 ijirset@gmail.com





Volume 11, Issue 5, May 2022

DOI:10.15680/IJIRSET.2022.1105047

### Implementation of an 8-Hypercube Network of 256 nodes by using an emerging Technology based multiple-input one-output logic gate

#### **Anup Kumar Biswas**

Assistant Professor, Department of Computer Science and Engineering, Kalyani Govt. Engineering College, Kalyani,

#### Nadia, West Bengal, India

ABSTRACT: In this work, an 8-dimentional network is presented, though a figure beyond 3-dimension can't be realized physically but realized virtually. We have tried to draw a figure of a network containing 256host points and it can be realized as an8-dimensional figure, because any host can be represented by a number of 8-bits and this 8-bits number can be treated as an address of a host point. An8-dimensional diagram of a network is traced in a 3-dimensional space. Total number of host points are  $2^8=256$  in this network. These host points are arranged in such a pattern so that this network system becomes free from collision of data packets and there would be no deadlock situation. For the purpose of finding out the predefined routing path for the present network, we should use the concept of deterministic algorithm which provides us the cognition that how the network becomes collision or deadlock free. For finding out the predefined routing path(s), host addresses involving in the path are measured by using XOR operation. Firstly, XOR operation is applied on the two address of hosts and then on the derived addresses according to the algorithm stated in section III. In lieu of classical CMOS based XOR gate, here a new model of XOR gate is used, which is made up of Multiple input threshold logic gate and using the operational concept of this gate, we have been able to make OR gate, AND gate, XOR gate, inverter and more complex combinational and sequential circuits. Among all of the gates, only XOR gate are chosen to overcome our problem. We have tried to find out the node addresses for predefining the deadlock free routing path for any set of (source, destination). A total number of (72) Exclusive-OR gates are arranged in a pattern discussed in section V and the desired circuit is implemented to develop a "Collision free network with a Routing path having at most 8 host nodes". The implemented circuit can help us to supply the predefined nodes through which the packets /messages/data can travel. For input-output values in some circuits, some truth tables are listed in substantial places. When applying the algorithm, it has been observed that for our network, the maximum path length are confined to 7, i.e. whenever any set of source and destination points are provided, then 8-node points are produced (including source and destination points). Among these 8-pionts, some of them may have identical addresses, as a result host/node numbers belonging to a path would be at most 8. The network designed and implemented must be a collision free one, because when a truth table is listed, somebody can observed that there are not any two or more number of nodes identical in a column.

KEYWORDS: 8-dimensional, Routing-path, collision free, Threshold, multi-input-one-output

#### I. INTRODUCTION

In this work, we have focused on an 8-hypercube network which is discussed elaborately in the following sections. By using the Algorithm-1 cited in section III, a large number of travelling paths containing some intermediate host points is given in tabular form. The distance between two host points in a link in the network can be determined by Hamming weight measured by *the number of* 1s in the Exclusive-OR operation for two binary numbers representing the source hostand destination host. The distance/length between host host P (represented as '11101001') and thehost Q (represented as '10001110') in the network given by:  $(11101001) \oplus (10001110) = 01100111$  is 5 (total number of 1s). To draw the 8-dimensional hypercube network is a troublesome job yet it is depicted in section III. For a pair of (source, destination) addresses the routing path is traced in deep and thick color for realizing the link.

The algorithm given in section III can be realized and represented by a combinational circuit which can be implemented by using an emerging technology based logic- called multi-input threshold logic. Based on this technology, three logic gates like AND, OR and XOR are designed and implemented. After implementing the XOR gate, a more complex circuit is depicted in section III. This circuit is able to play the role of tracing the routing path, calculating the path distance, and predicting the successive hostaddresses involving in the routing path. The circuit

L IJIRSET | e-ISSN: 2319-8753, p-ISSN: 2320-6710| <u>www.ijirset.com</u> | Impact Factor: 8.118|

Volume 11, Issue 5, May 2022

#### | DOI:10.15680/IJIRSET.2022.1105047 |

implemented using multiple-input threshold logic gate, which requires a small amount of energy in the scale of millielectron volt (meV) to execute, gives its high processing speed, in comparison with the CMOS based circuit.

#### **II. LITERATURE SURVEY**

An emerging technology on the basis of a single electron tunneling, which consumes very small amount of power, produces higher throughput and concentrates in a small space, will attracts scientists and researchers. Based on the movement of an electron a device can play the role of logic decision- high or low. Relying on these two logic levels, a circuit developed can predict the nodes involving in a link of a computer network. Theoretical aspect of multiple input single output logical circuit is analyzed and the input-output relationships of different circuits made from it are simulated and square results are obtained. The main fault for these circuits is "maintaining the atmosphere at very low temperature". At present, scientists and researchers are trying to make such electron tunneling so as to perform operation at  $0^{\circ}$ C - $10^{\circ}$ C. When succeeded this problem, a new horizon will open to the field of Technology and science so that manufacturer will be able to produce and provide the electronic goods/gadgets which consumes very low power, gives higher order of fan-out with small amount of time.

#### III. AN n-DIMENSIONAL HYPERCUBE FOR 2<sup>n</sup>NODES

In general, we consider an n-dimensional hypercube having  $2^n$  nodes (peaks). Each node having its node number is coded in a binary number like  $p = p_{n-1}p_{n-2}p_{n-3} \dots p_1p_0$ . The starting node also called source node ( $s = s_{n-1}s_{n-2}s_{n-3}\dots s_1s_0$ ) is the point from which routing is started. And a goal or destination node ( $d = d_{n-1}d_{n-2}d_{n-3}\dots d_1d_0$ ) to which the travelling is ended. How many nodes with their node numbers either in decimal or in binary required in a routing path are to be measured. We must find out the order of nodes involved in a routing path. Along with the node numbers and their order sequentially, the path length is measured also.

In general situation, we indicate the n-dimensions as i = 0, 1, 2, 3, ..., n - 1; here, the  $i^{\text{th}}$  dimension shows the  $(i - 1)^{\text{th}}$  bit in a node number  $q = q_{n-1}q_{n-2}q_{n-3} ... q_1q_0$ . The travelling route from starting /source node to the goal/destination (d) node can be determined by the following algorithm [2].

#### Algorithm-1:

- (i) Calculate the direction bit  $k_i = s_{i-1} \oplus d_{i-1}$  where i = 1, 2, 3, ..., n; Commence with first dimension
- i.e., i = 1 and q = s then follow the following.
- (ii) Start routing from the node q to the next node =  $(q \oplus 2^{i-1})$  if  $k_i = 1$ . If  $k_i = 0$ , skip the step.
- (iii) Move to the next dimension (i + 1), that is  $i \leftarrow i + 1$ . If  $i \le n$ , go to the previous step (ii), else done.

By taking an example, the algorithm written above can be explained. Consider an E-cube routing over a 7-dimensional hypercube given in Fig. 1.

In our present work, the dimension n = 8 in considered, so the total number of nodes in the network will be equal to  $2^8$ =256, the nodes are represented by the node numbers as 00000000 [0], 00000001 [1], 00000010 [2], ...,1111111 [255]. For the case of routing, the source and destination nodes are selected as, for example, S=00000101 and D=11001000 respectively. So, $k_i = k_8 k_7 k_6 k_5 k_4 k_3 k_2 k_1 = (S \oplus D) = =11001101$ . We must skip the dimension for i = 2, 5 and 6 as  $k_2=0\oplus0=0, k_5=0\oplus0=0, and k_6=0\oplus0=0$  respectively. Now our routing start from source node 00000101to ( $S \oplus (0000000k_1)$ ) = (00000101) $\oplus$ (00000001)= 00000100, since  $k_1 = 1\oplus0=1$ . Next route from q = (00000000) to  $(q \oplus (00000k_300)) = 00000000$ , since  $k_3=1\oplus0=1$ . Next route from q = (00000000) to  $(q \oplus (00000k_4000)) = 00001000$ , since  $k_4=0\oplus1=1$ . Next route from q = (0000100) to  $(q \oplus (0k_7000000)) = 01001000$ , since  $k_7=0\oplus1=1$ . Next route from q = (01001000) to  $(q \oplus (0k_7000000)) = 11001000$ , since  $k_8=0\oplus1=1$ . So, the routing path, arranging the sequence of order, should be: 00000101→00000100→00000000→0000000000)

Similarly, If we take another set of two nodes/hosts, i.e. source and destination nodes like S=1 1 1 1 1 1 0 1 and D=0 0 0 1 0 0 0 0 we obtain-

Source =  $(S_7S_6S_5S_4S_3S_2S_1S_0) = 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1$ Destination =  $(d_7d_6d_5d_4d_3d_2d_1d_0) = 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0$  $k_8k_7k_6k_5k_4k_3k_2k_1 = 11101101$ , in this case  $k_2 = 0$  and  $k_5 = 0$ 

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 8.118



Routing path:

Volume 11, Issue 5, May 2022

| DOI:10.15680/IJIRSET.2022.1105047 |

 $11111101 [253] \rightarrow 1111100 [252] \rightarrow 11111000 [248] \rightarrow 11110000 [240] \rightarrow 11010000 [208] \rightarrow 00010000 [16]$ 

#### IV. TRACKIG OF ROUTING PATH NODES AND NETWORK REPRESENTAION

As we are concentrated in 8-hypercube, so each node in a routing path is represented by 8-bit word. In the present work, the network system must have  $2^8 = 256$  nodes.Since there are 256 distinct nodes belong to the Fig.1, one have to find out the total number of source-destination pairs combination which can be calculated as (255 + 127 + 126 + 126) $125 + \dots + 2 + 1$  × 2 = 255 × 256 = 65280; here, the multiplication factor 2 is taken to be bidirectional communication. Among all these source-destination pairs, a small number of pairs are chosen and tested with respect to the circuit shown in Fig.2 to determine the number of nodes involved for the routing path made by any two source and destination points. The numbers of nodes involving in such situation is given in the last column in the Table-1 below. According to the algorithm-1, a maximum number of nodes connected to a routing path is 8. Note that when a pair of nodes/hosts are selected for the purpose of communication or data transferring, these two points may be even numbers or odd numbers or mixture of them. If they are even or odd, they follow the even (or odd) number nodes only. If the source point is even (or odd) number and the destination point is odd (or even), on the routing path the second point and the successive points up to destination must be odd (or even) number points. For instance, in serial numbers 1 and 8 in Table-1 both the source and destination points are of the same kind, i.e., in the case of Sl. No, 1, source and destination points are even numbers so all the nodes in this routing path should be of even numbers. Similarly for the case of Sl. No. 8 the node identity numbers must be odd. For the Sl. No. 16 source and the destination points are 46 and 35, here the second number, third etc. node numbers will be of odd numbers. If we consider the SI. No. 12, the  $2^{nd}$  and the remaining point up to destination will be of even numbers. If somebody expands the table up to Sl. No. 65280, he/she will not find any pair of common numbers in any column of the full-fledged Table-1. This is the beauty of the algorithm. As no two or more number of nodes in column are not being the same, there would be no possibility of conflicting, as a result it would be a deadlock free network system [19].

From Table-1, it is clear that there would be a maximum number of path length is =8, where a single path length is considered to be the segment between any two consecutive node points in the routing path concerned. In 15<sup>th</sup> row in the Table-1, the path length is 7 since every node number in the routing path is distinct. But in some rows, each node identification numbers/addresses are not the distinct. Some are distinct and some are not. This is why path lengths are varied. Path lengths are calculated in two ways:

(i) Total number of nodes (involved in a routing path) -1

(ii) Number of 1s in the vector  $k = \{k_8k_7k_6k_5k_4k_3k_2k_1\}$ 

For instance, in the row 5, source and destination points are 52 and 31. From source to destination, the routing path are traced with its nodes. The nodes involved in the path are 52,53,55,63 and 31. Since there are 5 nodes, so the path length is =5-1=4. The value of the vector kin row 5 is  $\{k_i\} = \{k_8k_7k_6k_5k_4k_3k_2k_1\} = \{00101011\}$ . Here, total number of 1s in this vector  $\{00101011\}$  is 4, hence the path length = 4.

SI. No.	Starting /	Goal / Destination	Original Source I	Source 2	Source 3	Source 4	Source 5	Source 6	Source 7	Source 8	goal	Routing Path Length = No. of 1s in $\{k_i\}$ =
1	24	228	111101	111101	111101	111101	111101	111001	111001	111001	111001	000100
	4		00	00	00	00	00	00	00	00	00	00
			=244	=244	=244	=244	=244	=228	=228	=228	=228	(1)
2	11	237	011101	011101	011101	011101	011111	011011	011011	011011	111011	100110
	8		10=11	11 =	01 =	01 =	01 =	01 =	01 =	01 =	01 =	11
			8	119	117	117	125	109	109	109	237	(5)
3	12	132	011110	011110	011110	011111	011101	011001	010001	000001	100001	111111
	3		11 =	10 =	00 =	00 =	00 =	00 =	00 =	00 = 4	00 =	11

#### Table-1 Minimum routing path(s)



Volume 11, Issue 5, May 2022

#### | DOI:10.15680/IJIRSET.2022.1105047 |

			123	122	120	124	116	100	68		132	(8)
4	13	123	100001	100001	100001	100000	100010	100110	101110	111110	011110	111111
	2		00 =	01 =	11 =	11 =	11 =	11 =	11 =	11 =	11 =	11
			132	133	135	131	139	155	187	251	123	(8)
5	52	31	001101	001101	001101	001101	001111	001111	000111	000111	000111	001010
			00 =	01 =	11 =	11 =	11 =	11 =	11 =	11 = 31	11 = 31	11
			52	53	55	55	63	63	31			(4)
6	14	129	000011	000011	000011	000010	000000	000000	000000	000000	100000	100011
			10 =	11 =	01 =	01 = 9	01 = 1	01 = 1	01 = 1	01 = 1	01 =	11
			14	15	13						129	(5)
7	3	199	000000	000000	000000	000001	000001	000001	000001	010001	110001	110001
			11 = 3	11 = 3	11 = 3	11 = 7	11 = 7	11 = 7	11 = 7	11 = 71	11 =	00
											199	(3)
8	19	95	000100	000100	000100	000101	000111	000111	000111	010111	010111	010011
			11 =	11 =	11 =	11 =	11 =	11 =	11 =	11 = 95	11 = 95	00
			19	19	19	23	31	31	31			(3)
9	7	253	000001	000001	000001	000001	000011	000111	001111	011111	111111	111110
			11 = 7	11 = 7	01 = 5	01 = 5	01 =	01 =	01 =	01 =	01 =	10
							13	29	61	125	253	(6)
10	41	89	001010	001010	001010	001010	001010	001110	000110	010110	010110	011100
			01 =	01 =	01 =	01 =	01 =	01 =	01 =	01 = 89	01 = 89	00
			41	41	41	41	41	57	25			(3)
11	0	255	000000	000000	000000	000001	000011	000111	001111	011111	111111	111111
			00 = 0	01 = 1	11 = 3	11 = 7	11 =	11 =	11 =	11 =	11 =	11
							15	31	63	127	255	(8)
12	25	0	111111	111111	111111	111110	111100	111000	110000	100000	000000	111111
	5		111-	10 -	00 -	00 -	00 -	00 -	00 -	00 -	00 - 0	11
	5		11 -	10 -	00 =	00 =	00 =	00 =	00 =	00 -	00 - 0	11
	5		255	254	252	248	240	224	192	128	00-0	(8)
13	22	222	255 000101	254 000101	252 000101	248 000101	240 000111	224 000111	192 000111	128 010111	110111	(8) 110010
13	22	222	255 000101 10 =	$     \begin{array}{r}       10 \\       254 \\       000101 \\       10 \\       22     \end{array} $				00 = 224 000111 10 = 20	192 000111 10 =	$     \begin{array}{r}       128 \\       0101111 \\       10 = 94     \end{array} $	110111 10 = 222	(8) 110010 00
13	22	222	255 000101 10 = 22	10 = 254 000101 10 = 22		248 000101 10 = 22	$     \begin{array}{r}       00 = \\       240 \\       000111 \\       10 = \\       30 \\       010110     \end{array} $	00 = 224 000111 10 = 30	$     \begin{array}{r}       00 = \\       192 \\       000111 \\       10 = \\       30 \\       011010     \end{array} $	$     \begin{array}{r}       128 \\       010111 \\       10 = 94     \end{array}   $	110111 10 = 222	(8) 110010 00 (3)
13	22 85	222 170	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 01 \end{array}$	10 = 254 000101 10 = 22 010101	00 = 252 000101 10 = 22 010101	$\begin{array}{c} 000 = \\ 248 \\ 000101 \\ 10 = \\ 22 \\ 010100 \\ 10 \end{array}$	$ \begin{array}{r} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \end{array} $	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ \end{array}$	$ \begin{array}{r} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \end{array} $	$\begin{array}{c} 0.0 \\ 128 \\ 0.001111 \\ 10 = 94 \\ 0.001010 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 1$	$ \begin{array}{c} 110111\\ 10 = \\ 222\\ 101010\\ 10 \end{array} $	(8) 110010 00 (3) 111111
13	22 85	222 170	11 = 255 000101 10 = 22 010101 01 = 25	10 = 254 000101 10 = 22 010101 00 = 24	000 = 252 000101 10 = 22 010101 10 = 26	$\begin{array}{c} 00 \\ 248 \\ 000101 \\ 10 \\ 22 \\ 010100 \\ 10 \\ 82 \end{array}$	$\begin{array}{r} 000 = \\ 240 \\ 000111 \\ 10 = \\ 30 \\ 010110 \\ 10 = \\ 00 \\ \end{array}$	000 = 224 000111 10 = 30 010010 10 = 74	$ \begin{array}{r} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 100 \end{array} $	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \end{array}$	$ \begin{array}{r} 110111\\ 10 = \\ 222\\ 101010\\ 10 = \\ 170\\ \end{array} $	(8) 110010 00 (3) 111111 11 (0)
13	22 85	222	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 01 \\ 85 \\ 111011 \end{array}$	$\begin{array}{c} 10 \\ 254 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 00 \\ 84 \\ 111011 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 10 \\ 86 \\ 111011 \end{array}$	$\begin{array}{c} 000 \\ 248 \\ 000101 \\ 10 \\ 22 \\ 010100 \\ 10 \\ 82 \\ 111011 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \end{array}$	$ \begin{array}{r} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 106 \\ 110101 \end{array} $	$\begin{array}{c} 000 \\ 128 \\ 010111 \\ 10 = 94 \\ 001010 \\ 10 = 42 \\ 100101 \\ \end{array}$	$ \begin{array}{c} 110111\\ 10 = \\ 222\\ 101010\\ 10 = \\ 170\\ 000101 \end{array} $	(8) 110010 00 (3) 111111 11 (8) 111110
13 14 15	22 85 23	222 170 21	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 01 \\ 85 \\ 111011 \\ 10 \\ - \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ - \\ 22 \\ 010101 \\ 00 \\ - \\ 84 \\ 111011 \\ 11 \\ - \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 10 \\ 86 \\ 111011 \\ 01 \\ - \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ - \\ 22 \\ 010100 \\ 10 \\ - \\ 82 \\ 111011 \\ 01 \\ - \end{array}$	$\begin{array}{c} 000 = \\ 240 \\ 000111 \\ 10 = \\ 30 \\ 010110 \\ 10 = \\ 90 \\ 111001 \\ 01 = \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ - \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 106 \\ 110101 \\ 01 \\ - \end{array}$	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ 001010 \\ 10 = 42 \\ 100101 \\ 01 - \end{array}$	$ \begin{array}{c} 110111\\ 10 = \\ 222\\ 101010\\ 10 = \\ 170\\ 000101\\ 01 = 21\\ \end{array} $	(8) 110010 00 (3) 111111 11 (8) 111110 11
13 14 15	22 85 23 8	222 170 21	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 01 \\ 85 \\ 111011 \\ 10 \\ 238 \end{array}$	$\begin{array}{c} 10 \\ -254 \\ 000101 \\ 10 \\ -22 \\ 010101 \\ 00 \\ -84 \\ 111011 \\ 11 \\ -239 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 10 \\ 86 \\ 111011 \\ 01 \\ 237 \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ - \\ 22 \\ 010100 \\ 10 \\ - \\ 82 \\ 111011 \\ 01 \\ - \\ 237 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 220 \\ \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 106 \\ 110101 \\ 01 \\ 213 \end{array}$	$\begin{array}{c} 0.0 \\ 128 \\ 0.10111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ \end{array}$	$ \begin{array}{c} 11\\ (8)\\ 110010\\ 00\\ (3)\\ 111111\\ 11\\ (8)\\ 111110\\ 11\\ (7)\\ \end{array} $
13 14 15	22 85 23 8	222 170 21	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ - \\ 22 \\ 010101 \\ 00 \\ - \\ 84 \\ 111011 \\ 11 \\ - \\ 239 \\ 011111 \end{array}$	$\begin{array}{c} 000 \\ 252 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 10 \\ 86 \\ 111011 \\ 01 \\ 237 \\ 011111 \end{array}$	$\begin{array}{c} 000 \\ 248 \\ 000101 \\ 10 \\ 22 \\ 010100 \\ 10 \\ 82 \\ 111011 \\ 01 \\ 237 \\ 011110 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 106 \\ 110101 \\ 01 \\ 213 \\ 010010 \\ \end{array}$	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ 000010 \\ \hline \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ \end{array}$	(8) 110010 00 (3) 111111 11 (8) 111110 11 (7) 111101
13 14 15 16	22 85 23 8 12 6	222 170 21 139	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ - \\ 22 \\ 010101 \\ 00 \\ - \\ 84 \\ 111011 \\ 11 \\ - \\ 239 \\ 011111 \\ 11 \\ - \\ \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ 22 \\ 010101 \\ 10 \\ 86 \\ 111011 \\ 01 \\ 237 \\ 011111 \\ 11 \\ - \end{array}$	$\begin{array}{c} 00 \\ 248 \\ 000101 \\ 10 \\ 22 \\ 010100 \\ 10 \\ 82 \\ 111011 \\ 01 \\ 237 \\ 011110 \\ 11 \\ - \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ - \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ - \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 100 \\ 11010 \\ 01 \\ 213 \\ 010010 \\ 11 \\ - \end{array}$	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ 001010 \\ 10 = 42 \\ 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 = 0 \end{array}$	(8) 110010 00 (3) 111111 11 (8) 111110 11 (7) 111101 01
13 14 15 16	22 85 23 8 12 6	222 170 21 139	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ - \\ 22 \\ 010101 \\ 00 \\ - \\ 84 \\ 111011 \\ 11 \\ - \\ 239 \\ 011111 \\ 11 \\ - \\ 127 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ \hline 111011 \\ 01 \\ = \\ 237 \\ \hline 011111 \\ 11 \\ = \\ 127 \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ - \\ 22 \\ 010100 \\ 10 \\ - \\ 82 \\ 111011 \\ 01 \\ - \\ 237 \\ 011110 \\ 11 \\ - \\ 123 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 11 \\ 123 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ 107 \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 100 \\ 1001 \\ 01 \\ $	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ \hline 000010 \\ 11 = 11 \\ \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139 \end{array}$	$(8) \\(110010) \\(00) \\(3) \\(111111) \\(11) \\(8) \\(111110) \\(11) \\(7) \\(111101) \\(01) \\(6) \\(6) \\(8) \\(11001) \\($
13 14 15 16	22 85 23 8 12 6	222 170 21 139	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ - \\ 22 \\ 010101 \\ 00 \\ - \\ 84 \\ 111011 \\ 11 \\ - \\ 239 \\ 011111 \\ 11 \\ - \\ 127 \\ 001100 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ - \\ 22 \\ 010100 \\ 10 \\ - \\ 82 \\ 111011 \\ 01 \\ - \\ 237 \\ 011110 \\ 11 \\ - \\ 123 \\ 001101 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ 107 \\ 001011 \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	$\begin{array}{c} 0.0 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline 001011 \\ \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011 \end{array}$	$(8) \\(110010) \\(00) \\(3) \\(111111) \\(11) \\(8) \\(111110) \\(11) \\(7) \\(111101) \\(01) \\(6) \\(000111) \\($
13 14 15 16 17	22 85 23 8 12 6 48	222 170 21 139 47	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \end{array}$	$\begin{array}{c} 10 \\ -254 \\ \hline 000101 \\ 10 \\ = \\ 22 \\ \hline 010101 \\ 00 \\ = \\ 84 \\ \hline 111011 \\ 11 \\ = \\ 239 \\ \hline 011111 \\ 11 \\ = \\ 127 \\ \hline 001100 \\ 01 \\ = \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010101 \\ 10 \\ = \\ 86 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \\ 11 \\ = \end{array}$	$\begin{array}{c} 000 \\ 248 \\ 000101 \\ 10 \\ 22 \\ 010100 \\ 10 \\ 82 \\ 111011 \\ 01 \\ 237 \\ 011110 \\ 11 \\ 123 \\ 001101 \\ 11 \\ \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \\ 11 \\ \end{array}$	$\begin{array}{c} 00 \\ 224 \\ \hline 000111 \\ 10 \\ 30 \\ \hline 010010 \\ 10 \\ \hline 10 \\ 74 \\ \hline 111101 \\ 01 \\ 245 \\ \hline 011010 \\ 11 \\ 1 \\ 107 \\ \hline 001011 \\ 11 \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline 001011 \\ 11 = 47 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ \end{array}$	(8) 110010 00 (3) 111111 11 (8) 111110 11 (7) 111101 01 (6) 000111 11
13 14 15 16 17	22 85 23 8 12 6 48	222 170 21 139 47	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 00 \\ = \\ 84 \\ 111011 \\ 11 \\ = \\ 239 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 01 \\ = \\ 49 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ 111011 \\ 01 \\ = \\ 237 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 11 \\ = \\ 51 \end{array}$	$\begin{array}{c} 00 \\ 00 \\ 248 \\ 000101 \\ 10 \\ 22 \\ 010100 \\ 10 \\ 82 \\ 111011 \\ 01 \\ 82 \\ 111011 \\ 01 \\ 237 \\ 011110 \\ 11 \\ 123 \\ 001101 \\ 11 \\ 55 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \\ 11 \\ 11 \\ 63 \end{array}$	$\begin{array}{c} 00 \\ \hline 224 \\ \hline 000111 \\ 10 \\ \hline 30 \\ \hline 010010 \\ 10 \\ \hline 74 \\ \hline 111101 \\ 01 \\ \hline 245 \\ \hline 011010 \\ 11 \\ \hline 107 \\ \hline 001011 \\ 11 \\ \hline 1 \\ 47 \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline 001011 \\ 11 = 47 \\ \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ \end{array}$	$(8) \\ 110010 \\ 00 \\ (3) \\ 111111 \\ 11 \\ (8) \\ 111110 \\ 11 \\ (7) \\ 111101 \\ 01 \\ (6) \\ 000111 \\ 11 \\ (5) \\ (8) \\ (9) \\ $
13 14 15 16 17 18	22 85 23 8 12 6 48	222 170 21 139 47 217	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 00 \\ = \\ 84 \\ 111011 \\ 11 \\ = \\ 239 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 01 \\ = \\ 49 \\ 001110 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ 111011 \\ 01 \\ = \\ 237 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 11 \\ = \\ 51 \\ 001110 \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ - \\ 22 \\ 010100 \\ 10 \\ - \\ 82 \\ 111011 \\ 01 \\ - \\ 237 \\ 011110 \\ 11 \\ - \\ 123 \\ 001101 \\ 11 \\ - \\ 55 \\ 001110 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \\ 11 \\ 63 \\ 001110 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ 107 \\ 001011 \\ 11 \\ 47 \\ 001110 \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	128 010111 10 = 94 001010 10 = 42 100101 01 = 149 000010 11 = 11 001011 11 = 47 010110	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ \end{array}$	$(8) \\(110010) \\(00) \\(3) \\(111111) \\(11111) \\(11110) \\(1110) \\(1$
13 14 15 16 17 18	22 85 23 8 12 6 48 59	222 170 21 139 47 217	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 00 \\ = \\ 84 \\ 111011 \\ 11 \\ = \\ 239 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 01 \\ = \\ 49 \\ 001110 \\ 11 \\ = \end{array}$	$\begin{array}{c} 00 \\ 252 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ 111011 \\ 01 \\ = \\ 237 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 11 \\ = \\ 51 \\ 001110 \\ 01 \\ = \end{array}$	$\begin{array}{c} 00 \\ 248 \\ 000101 \\ 10 \\ = \\ 22 \\ 010100 \\ 10 \\ = \\ 82 \\ 111011 \\ 01 \\ = \\ 237 \\ 011110 \\ 11 \\ = \\ 123 \\ 001101 \\ 11 \\ = \\ 55 \\ 001110 \\ 01 \\ = \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \\ 11 \\ 63 \\ 001110 \\ 01 \\ = \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ 107 \\ 001011 \\ 11 \\ 47 \\ 001110 \\ 01 \\ = \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 1 \\ 106 \\ 110101 \\ 01 \\ $	$\begin{array}{c} 0.0 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline 001011 \\ 11 = 47 \\ \hline 010110 \\ 01 = 89 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ \end{array}$	$(8) \\(110010) \\(00) \\(3) \\(111111) \\(11111) \\(11110) \\(11110) \\(11110) \\(11110) \\(11110) \\(6) \\(000111) \\(11100) \\(10)$
13 14 15 16 17 18	22 85 23 8 12 6 48 59	222 170 21 139 47 217	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \\ 59 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ \hline 000101 \\ 10 \\ = \\ 22 \\ \hline 010101 \\ 00 \\ = \\ 84 \\ \hline 111011 \\ 11 \\ = \\ 239 \\ \hline 011111 \\ 11 \\ = \\ 127 \\ \hline 001100 \\ 01 \\ = \\ 49 \\ \hline 001110 \\ 11 \\ = \\ 59 \\ \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010101 \\ 10 \\ = \\ 86 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \\ 11 \\ = \\ 51 \\ \hline \\ 001110 \\ 01 \\ = \\ 57 \end{array}$	$\begin{array}{c} 00 \\ 248 \\ \hline 000101 \\ 10 \\ = \\ 22 \\ \hline 010100 \\ 10 \\ = \\ 82 \\ \hline 111011 \\ 01 \\ = \\ 237 \\ \hline 011110 \\ 11 \\ = \\ 123 \\ \hline 001101 \\ 11 \\ = \\ 55 \\ \hline 001110 \\ 01 \\ = \\ 57 \\ \end{array}$	$\begin{array}{c} 0.0 \\ \hline 240 \\ \hline 000111 \\ 10 \\ = \\ 30 \\ \hline 010110 \\ 10 \\ = \\ 90 \\ \hline 111001 \\ 01 \\ = \\ 229 \\ \hline 011110 \\ 11 \\ = \\ 123 \\ \hline 001111 \\ 11 \\ = \\ 63 \\ \hline 001110 \\ 01 \\ = \\ 57 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ \hline 000111 \\ 10 \\ = \\ 30 \\ \hline 010010 \\ 10 \\ = \\ 74 \\ \hline 111101 \\ 01 \\ = \\ 245 \\ \hline 011010 \\ 11 \\ = \\ 107 \\ \hline 001011 \\ 11 \\ = \\ 47 \\ \hline 001110 \\ 01 \\ = \\ 57 \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	$\begin{array}{c} 00 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline 001010 \\ 10 = 42 \\ \hline 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline 001011 \\ 11 = 47 \\ \hline 010110 \\ 01 = 89 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ 217\\ \end{array}$	$(8) \\ 110010 \\ 00 \\ (3) \\ 111111 \\ 11 \\ (8) \\ 111110 \\ 11 \\ (7) \\ 111101 \\ 01 \\ (6) \\ 000111 \\ 11 \\ (5) \\ 111000 \\ 10 \\ (4) \\ (4) \\ (8) $
13 14 15 16 17 18 19	22 85 23 8 12 6 48 59 24	222 170 21 139 47 217 255	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 00 \\ = \\ 84 \\ 111011 \\ 11 \\ = \\ 239 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 01 \\ = \\ 49 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \end{array}$	$\begin{array}{c} 000 \\ 252 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ 111011 \\ 01 \\ = \\ 237 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 11 \\ = \\ 51 \\ 001110 \\ 01 \\ = \\ 57 \\ 111110 \end{array}$	$\begin{array}{c} 000 \\ 248 \\ 000101 \\ 10 \\ = \\ 22 \\ 010100 \\ 10 \\ = \\ 82 \\ 111011 \\ 01 \\ = \\ 237 \\ 011110 \\ 11 \\ = \\ 123 \\ 001101 \\ 11 \\ = \\ 55 \\ 001110 \\ 01 \\ = \\ 57 \\ 111111 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \\ 11 \\ 63 \\ 001110 \\ 01 \\ 57 \\ 111111 \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ 245 \\ 011010 \\ 11 \\ 107 \\ 001011 \\ 11 \\ 47 \\ 001110 \\ 01 \\ 57 \\ 111111 \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	128 $010111$ $10 = 94$ $001010$ $10 = 42$ $100101$ $01 =$ $149$ $000010$ $11 = 11$ $001011$ $11 = 47$ $010110$ $01 = 89$ $111111$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ 217\\ 111111\\ \end{array}$	(8)         110010         00         (3)         111111         11         (8)         111110         11         (7)         111101         01         (6)         000111         11         (5)         111000         10         (4)
13 14 15 16 17 18 19	22 85 23 8 12 6 48 59 24 8	222 170 21 139 47 217 255	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \\ 00 \\ = \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 00 \\ = \\ 84 \\ 111011 \\ 11 \\ = \\ 239 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 01 \\ = \\ 49 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \\ 01 \\ = \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 10 \\ = \\ 86 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \\ 11 \\ = \\ 57 \\ \hline \\ 111110 \\ 11 \\ = \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ - \\ 22 \\ 010100 \\ 10 \\ - \\ 82 \\ 111011 \\ 01 \\ - \\ 237 \\ 011110 \\ 11 \\ - \\ 123 \\ 001101 \\ 11 \\ - \\ 55 \\ 001110 \\ 01 \\ - \\ 57 \\ 111111 \\ 11 \\ - \end{array}$	$\begin{array}{c} 000 \\ 240 \\ 000111 \\ 10 \\ 30 \\ 010110 \\ 10 \\ 90 \\ 111001 \\ 01 \\ 229 \\ 011110 \\ 11 \\ 123 \\ 001111 \\ 11 \\ 63 \\ 001111 \\ 01 \\ 57 \\ 111111 \\ 11 \\ \end{array}$	$\begin{array}{c} 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 74 \\ 111101 \\ 01 \\ 245 \\ 011010 \\ 11 \\ 245 \\ 011010 \\ 11 \\ 1 \\ 47 \\ 001011 \\ 11 \\ 47 \\ 001110 \\ 01 \\ 57 \\ 111111 \\ 11 \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	$\begin{array}{c} 0.0 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline \\ 001010 \\ 10 = 42 \\ \hline \\ 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline \\ 001011 \\ 11 = 47 \\ \hline \\ 010110 \\ 01 = 89 \\ \hline \\ 111111 \\ 11 = \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ 217\\ 111111\\ 11 =\\ \end{array}$	(8)         110010         00         (3)         111111         11         (8)         111110         11         (7)         111101         01         (6)         000111         11         (5)         111000         10         (4)         000001         11
13 14 15 16 17 18 19	22 85 23 8 12 6 48 59 24 8	222 170 21 139 47 217 255	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \\ 00 \\ = \\ 248 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 00 \\ = \\ 84 \\ 111011 \\ 11 \\ = \\ 239 \\ 011111 \\ 11 \\ = \\ 127 \\ 001100 \\ 01 \\ = \\ 49 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \\ 01 \\ = \\ 249 \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010101 \\ 10 \\ = \\ 86 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \\ 11 \\ = \\ 51 \\ \hline \\ 001110 \\ 01 \\ = \\ 57 \\ \hline \\ 111110 \\ 11 \\ = \\ 251 \end{array}$	$\begin{array}{c} 00 \\ - \\ 248 \\ 000101 \\ 10 \\ = \\ 22 \\ 010100 \\ 10 \\ = \\ 82 \\ 111011 \\ 01 \\ = \\ 237 \\ 011110 \\ 11 \\ = \\ 123 \\ 001101 \\ 11 \\ = \\ 55 \\ 001110 \\ 01 \\ = \\ 57 \\ 111111 \\ 11 \\ = \\ 255 \end{array}$	$\begin{array}{c} 000 \\ 240 \\ \hline \\ 000111 \\ 10 \\ = \\ 30 \\ \hline \\ 010110 \\ 10 \\ = \\ 90 \\ \hline \\ 111001 \\ 01 \\ = \\ 229 \\ \hline \\ 01110 \\ 11 \\ = \\ 123 \\ \hline \\ 001111 \\ 11 \\ = \\ 57 \\ \hline \\ 111111 \\ 11 \\ = \\ 255 \\ \end{array}$	$\begin{array}{c} 00 \\ \hline \\ 224 \\ \hline \\ 000111 \\ 10 \\ \hline \\ 30 \\ \hline \\ 010010 \\ 10 \\ \hline \\ 74 \\ \hline \\ 111101 \\ 01 \\ \hline \\ 245 \\ \hline \\ 011010 \\ 11 \\ \hline \\ 11 \\ \hline \\ 47 \\ \hline \\ 001011 \\ 11 \\ \hline \\ 47 \\ \hline \\ 001110 \\ 01 \\ \hline \\ 57 \\ \hline \\ 111111 \\ 11 \\ \hline \\ 255 \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ \hline 000111 \\ 10 \\ = \\ 30 \\ \hline 011010 \\ 10 \\ = \\ 106 \\ \hline 110101 \\ 01 \\ = \\ 213 \\ \hline 010010 \\ 11 \\ = \\ 75 \\ \hline 001011 \\ 11 \\ = \\ 47 \\ \hline 000110 \\ 01 \\ = \\ 25 \\ \hline 111111 \\ 11 \\ = \\ 255 \\ \end{array}$	$\begin{array}{c} 0.0 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline \\ 001010 \\ 10 = 42 \\ \hline \\ 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline \\ 001011 \\ 11 = 47 \\ \hline \\ 010110 \\ 01 = 89 \\ \hline \\ 111111 \\ 11 = \\ 255 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ 217\\ 111111\\ 11 =\\ 255 \end{array}$	$(8) \\ 110010 \\ 00 \\ (3) \\ 111111 \\ 11 \\ (8) \\ 111110 \\ 11 \\ (7) \\ 111101 \\ 01 \\ (6) \\ 000111 \\ 11 \\ (5) \\ 111000 \\ 10 \\ (4) \\ 000001 \\ 11 \\ (3) \\ (8) \\ (8) \\ (8) \\ (10) \\ (10) \\ (10) \\ (11) \\ (3) \\ (10) \\ (10) \\ (10) \\ (10) \\ (11) \\ (3) \\ (10) \\ (10) \\ (10) \\ (10) \\ (11) \\ (3) \\ (10) \\ (10) \\ (10) \\ (10) \\ (10) \\ (11) \\ $
13 14 15 16 17 18 19 20	22 85 23 8 12 6 48 59 24 8 19	222 170 21 139 47 217 255 185	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \\ 00 \\ = \\ 248 \\ 101111 \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ \hline 000101 \\ 10 \\ = \\ 22 \\ \hline 010101 \\ 00 \\ = \\ 84 \\ \hline 111011 \\ 11 \\ = \\ 239 \\ \hline 011111 \\ 11 \\ = \\ 127 \\ \hline 001100 \\ 01 \\ = \\ 49 \\ \hline 001110 \\ 11 \\ = \\ 59 \\ \hline 111110 \\ 01 \\ = \\ 249 \\ \hline 101111 \\ \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010101 \\ 10 \\ = \\ 86 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \\ 11 \\ = \\ 51 \\ \hline \\ 001110 \\ 01 \\ = \\ 57 \\ \hline \\ 111110 \\ 11 \\ = \\ 251 \\ \hline \\ 101111 \end{array}$	$\begin{array}{c} 00 \\ 00 \\ 248 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010100 \\ 10 \\ = \\ 82 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011110 \\ 11 \\ = \\ 55 \\ \hline \\ 001101 \\ 11 \\ = \\ 57 \\ \hline \\ 111111 \\ 11 \\ = \\ 255 \\ \hline \\ 101110 \\ \end{array}$	$\begin{array}{c} 0.0 \\ 240 \\ \hline 240 \\ 000111 \\ 10 \\ = \\ 30 \\ \hline 010110 \\ 10 \\ = \\ 90 \\ \hline 111001 \\ 01 \\ = \\ 229 \\ \hline 011110 \\ 11 \\ = \\ 123 \\ \hline 001111 \\ 11 \\ = \\ 57 \\ \hline 111111 \\ 11 \\ = \\ 255 \\ \hline 101110 \\ \end{array}$	$\begin{array}{c} 00 \\ 00 \\ 224 \\ 000111 \\ 10 \\ 30 \\ 010010 \\ 10 \\ 10 \\$	$\begin{array}{c} 000 \\ 192 \\ \hline 000111 \\ 10 \\ = \\ 30 \\ \hline 011010 \\ 10 \\ = \\ 106 \\ \hline 110101 \\ 01 \\ = \\ 213 \\ \hline 010010 \\ 11 \\ = \\ 75 \\ \hline 001011 \\ 11 \\ = \\ 47 \\ \hline 000110 \\ 01 \\ = \\ 25 \\ \hline 111111 \\ 11 \\ = \\ 255 \\ \hline 101110 \\ \end{array}$	$\begin{array}{c} 0.0 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline \\ 001010 \\ 10 = 42 \\ \hline \\ 100101 \\ 01 = \\ 149 \\ 000010 \\ 11 = 11 \\ \hline \\ 001011 \\ 11 = 47 \\ \hline \\ 010110 \\ 01 = 89 \\ \hline \\ 111111 \\ 11 = \\ 255 \\ 101110 \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ 217\\ 111111\\ 11 =\\ 255\\ 101110\\ \end{array}$	11         (8)         110010         00         (3)         111111         11         (8)         111110         11         (7)         111101         01         (6)         000111         11         (5)         111000         10         (4)         000001         11         (3)         000001
13         14         15         16         17         18         19         20	22 85 23 8 12 6 48 59 24 8 19 1	222 170 21 139 47 217 255 185	$\begin{array}{c} 11 \\ 255 \\ 000101 \\ 10 \\ = \\ 22 \\ 010101 \\ 01 \\ = \\ 85 \\ 111011 \\ 10 \\ = \\ 238 \\ 011111 \\ 10 \\ = \\ 126 \\ 001100 \\ 00 \\ = \\ 48 \\ 001110 \\ 11 \\ = \\ 59 \\ 111110 \\ 00 \\ = \\ 248 \\ 101111 \\ 11 \\ = \end{array}$	$\begin{array}{c} 10 \\ - \\ 254 \\ \hline 000101 \\ 10 \\ = \\ 22 \\ \hline 010101 \\ 00 \\ = \\ 84 \\ \hline 111011 \\ 11 \\ = \\ 239 \\ \hline 011111 \\ 11 \\ = \\ 127 \\ \hline 001100 \\ 01 \\ = \\ 49 \\ \hline 001110 \\ 11 \\ = \\ 59 \\ \hline 111110 \\ 01 \\ = \\ 249 \\ \hline 101111 \\ 11 \\ = \end{array}$	$\begin{array}{c} 00 \\ 252 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010101 \\ 10 \\ = \\ 22 \\ \hline \\ 010101 \\ 01 \\ = \\ 237 \\ \hline \\ 011111 \\ 11 \\ = \\ 127 \\ \hline \\ 001100 \\ 11 \\ = \\ 57 \\ \hline \\ 111110 \\ 11 \\ = \\ 251 \\ \hline \\ 101111 \\ 01 \\ = \end{array}$	$\begin{array}{c} 00 \\ 00 \\ 248 \\ \hline \\ 000101 \\ 10 \\ = \\ 22 \\ \hline \\ 010100 \\ 10 \\ = \\ 82 \\ \hline \\ 111011 \\ 01 \\ = \\ 237 \\ \hline \\ 011110 \\ 11 \\ = \\ 123 \\ \hline \\ 001101 \\ 11 \\ = \\ 55 \\ \hline \\ 001101 \\ 01 \\ = \\ 57 \\ \hline \\ 111111 \\ 11 \\ = \\ 255 \\ \hline \\ 101110 \\ 01 \\ = \\ \end{array}$	$\begin{array}{c} 0.0 \\ 240 \\ \hline 240 \\ 000111 \\ 10 \\ = \\ 30 \\ \hline 010110 \\ 10 \\ = \\ 90 \\ \hline 111001 \\ 01 \\ = \\ 229 \\ \hline 011110 \\ 11 \\ = \\ 123 \\ \hline 001111 \\ 11 \\ = \\ 57 \\ \hline 111111 \\ 11 \\ = \\ 255 \\ \hline 101110 \\ 01 \\ = \\ \end{array}$	$\begin{array}{c} 00 \\ 00 \\ 224 \\ \hline \\ 000111 \\ 10 \\ = \\ 30 \\ \hline \\ 010010 \\ 10 \\ = \\ 74 \\ \hline \\ 111101 \\ 01 \\ = \\ 245 \\ \hline \\ 011010 \\ 11 \\ = \\ 47 \\ \hline \\ 001011 \\ 11 \\ = \\ 47 \\ \hline \\ 001011 \\ 01 \\ = \\ 57 \\ \hline \\ 111111 \\ 11 \\ = \\ 255 \\ \hline \\ 101110 \\ 01 \\ = \\ \end{array}$	$\begin{array}{c} 000 \\ 192 \\ 000111 \\ 10 \\ 30 \\ 011010 \\ 10 \\ 10 \\$	$\begin{array}{c} 0.0 \\ 128 \\ 010111 \\ 10 = 94 \\ \hline \\ 001010 \\ 10 = 42 \\ \hline \\ 100101 \\ 01 = \\ 149 \\ \hline \\ 000010 \\ 11 = 11 \\ \hline \\ 001011 \\ 11 = 47 \\ \hline \\ 010110 \\ 01 = 89 \\ \hline \\ 111111 \\ 11 = \\ 255 \\ 101110 \\ 01 = \end{array}$	$\begin{array}{c} 110111\\ 10 =\\ 222\\ 101010\\ 10 =\\ 170\\ 000101\\ 01 = 21\\ 100010\\ 11 =\\ 139\\ 001011\\ 11 = 47\\ 110110\\ 01 =\\ 217\\ 111111\\ 11 =\\ 255\\ 101110\\ 01 =\\ \end{array}$	11         (8)         110010         00         (3)         111111         11         (8)         111110         11         (7)         111101         01         (6)         000111         11         (5)         111000         10         (4)         000001         11         (3)         0000001         10



Volume 11, Issue 5, May 2022

#### | DOI:10.15680/IJIRSET.2022.1105047 |

21	15	142	100111	100111	100111	100111	100111	100011	100011	100011	100011	000100
	8		10 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	00
	-		158	158	158	158	158	142	142	142	142	(1)
22	17	170	101010	101010	101010	101010	101010	101010	101010	101010	101010	000000
	0		10 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	00
			170	170	170	170	170	170	170	170	170	(0)
23	13	130	100001	100001	100001	100000	100000	100000	100000	100000	100000	000001
	2		00 =	00 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	10
			132	132	134	130	130	130	130	130	130	(2)
24	13	5	100001	100001	100001	100001	100001	100001	100001	100001	000001	100000
	5		11 =	11 =	01 =	01 =	01 =	01 =	01 =	01 =	01 = 5	10
			135	135	133	133	133	133	133	133		(2)
25	12	234	011110	011110	011110	011110	011110	011010	011010	011010	111010	100100
	3		11 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	10 =	01
			123	122	122	122	122	106	106	106	234	(3)
26	11	000	011011	011011	011011	011010	011000	011000	010000	000000	000000	011011
	1		11 =	10 =	00 =	00 =	00 =	00 =	00 =	00 = 0	00 = 0	11
			111	110	108	104	96	96	64			(6)
27	25	4	111111	111111	111111	111111	111101	111001	110001	100001	000001	111110
	4		10 =	10 =	00 =	00 =	00 =	00 =	00 =	00 =	00 = 4	10
			254	254	252	252	244	228	196	132		(6)
28	11	35	011100	011100	011100	011100	011100	011000	011000	001000	001000	010100
	5		11 =	11 =	11 =	11 =	11 =	11 =	11 =	11 = 35	11 = 35	00
			115	115	115	115	115	99	99			(2)
29	10	10	011001	011001	011001	011000	011010	011010	010010	000010	000010	011011
	0		00 =	00 =	10 =	10 =	10 =	10 =	10 =	10 = 10	10 = 10	10
			100	100	102	98	106	106	74			(5)
30	61	161	001111	001111	001111	001110	001100	001000	001000	001000	101000	100111
			01 =	01 =	01 =	01 =	01 =	01 =	01 =	01 = 33	01 =	00
			61	61	61	57	49	33	33		161	(4)
31	59	3	001110	001110	001110	001110	001100	001000	000000	000000	000000	001110
			11 =	11 =	11 =	11 =	11 =	11 =	11 = 3	11 = 3	11 = 3	00
			59	59	59	59	51	35				(3)
32	20	225	110100	110100	110100	110100	110100	110000	111000	111000	111000	001100
	9		01 =	01 =	01 =	01 =	01 =	01 =	01 =	01 =	01 =	00
			209	209	209	209	209	193	225	225	225	(2)
33	17	76	101100	101100	101100	101101	101111	101011	100011	110011	010011	111111
	9		11 =	10 =	00 =	00 =	00 =	00 =	00 =	00 =	00 = 76	11
			179	178	176	180	188	172	140	204		(8)

As the 8-hypercube is considered in this study, total number of nodes/hosts  $2^8=256$  will be explored for transparency of our understanding. The arrangement of 256 nodes being presented by 8-bits each is impractical to represent in eight dimensions. One way accepted to arrange the nodes in three dimensional case is shown in Fig.1. It is troublesome to draw this figure because the sequence of numbers arranged for nodes are not sequential and in good.

According to the Algorithm 1, some routing paths are found out between some pairs of source and destination and they are listed in the Table-1. A full-fledged interconnection among the 256 nodes with their node numbers is shown in the Fig.1. We can take any two points as source and destination and when these two node point are applied in the algorithm we get the corresponding path specifying the nodes being involved. For showing the paths in the figure, two pairs of points are taken as (179, 76) and (59, 217). The corresponding path of (179, 76) is indicated by number and arrow as below.

 $179(10110011) \rightarrow 178(10110010) \rightarrow 176(10110000) \rightarrow 180(10110100) \rightarrow 188(10111100) \rightarrow 172(10101100) \rightarrow 140(10001100) \rightarrow 204(11001100) \rightarrow 76(01001100)$ 

よう続 IJIRSET | e-ISSN: 2319-8753, p-ISSN: 2320-6710| <u>www.ijirset.com</u> | Impact Factor: 8.118|

Volume 11, Issue 5, May 2022

#### DOI:10.15680/IJIRSET.2022.1105047

The routing path of the source destination pair (179, 76) is traced in thick and deep green in the Fig.1. Similarly, for the source destination pair (59, 217), the corresponding path of it is indicated by numbers and arrows as below.

 $59\ (00111011) \rightarrow 57\ (00111001) \rightarrow 25(00011001) \rightarrow 89(01011001) \rightarrow 217(11011001)$ 

And its corresponding routing path is traced in thick and red in Fig. 1 also.



Fig. 1. 8-hypercube node configuration for 256 nodes

#### V. CIRCUIT IMPLEMENTATION OF ROUTING PATH TRACING BASED THE ALGORITHM 1

The node arrangement of 8-hypercube for a network containing 256 node points indicated by 8-bit words is depicted in Fig. 1. The Fig. 2 is drawn maintaining all the conditions of the Algorithm 1 provided in the section III. The two sets of binary numbers  $(S_7S_6S_5S_4S_3S_2S_1S_0)$  and  $(d_7d_6d_5d_4d_3d_2d_1d_0)$  are considered as the source /starting point from which data are transmitted and the destination to which data are reached respectively. The values of  $(k_8k_7k_6k_5k_4k_3k_2k_1)$  are found out with the help of these two numbers. As soon as the values of  $(k_8k_7k_6k_5k_4k_3k_2k_1)$  are obtained, the main circuit starts calculating the node numbers which belong to the routing path created with respect to the source and destination  $(S_7S_6S_5S_4S_3S_2S_1S_0)$  and  $(d_7d_6d_5d_4d_3d_2d_1d_0)$  points. For any two points treated as source and destination,

e-ISSN: 2319-8753, p-ISSN: 2320-6710 www.ijirset.com | Impact Factor: 8.118



Volume 11, Issue 5, May 2022

#### DOI:10.15680/LJIRSET.2022.1105047

there can be maximum path segments 8 and the range of the segments numbers would be 0 to 8. If the source and destination points are selected as the same binary numbers then the segment numbers must be zero (0). For instance, Taking into account the source  $(s_7s_6s_5s_4s_3s_2s_1s_0) = 10000011 [131]_{10}$  and destination  $(d_7d_6d_5d_4d_3d_2d_1d_0) = 10000010 [130]_{10}$ , then the routing path segment will be 1 described as  $131 \rightarrow 130$ .

Taking another two points  $(s_7s_6s_5s_4s_3s_2s_1s_0) = 00011100 [28]_{10}$  and  $(d_7d_6d_5d_4d_3d_2d_1d_0) = 01101001 = [105]_{10}$ , then the algorithmic path would be  $28 \rightarrow 29 \rightarrow 25 \rightarrow 9 \rightarrow 41 \rightarrow 105$ . So the path segments would be 5.



Fig.2 Fig. 8(a) Circuit calculating  $\{k_1k_2k_3k_4k_5k_6k_7\}$  and all the node values  $(s_6s_5s_4s_3s_2s_1s_0)$  connecting the source and destination points

#### I. MANY-INPUT ONE-OUTPUT THRESHOLD LOGIC

In the present era, power crisis happens in the whole world so when a new device/system are being construct one should have considered about power to be consumed. Along with the power consumption, time delay is to be taken into consideration. It would be better if the materials/components required are managed to minimize. One way we can do it by using a fundamental circuit called "multiple-input but single output threshold circuit". The concept of it is described below.

A Many-input one-output threshold logicgate is a fundamental construction based on a single election tunneling phenomenon [3-8, 14-16]. This logic gate is also said to be a generic Multiple-input threshold logic gate (TLG) depictedbelow in Fig. 3.



Volume 11, Issue 5, May 2022

#### DOI:10.15680/IJIRSET.2022.1105047



Fig. 3 amultiple-input but one-output TLG

Thisgeneric multiple-input one-output based Threshold Logic Gate (LTG) shown above ismade up of a tunnel junction capacitance  $C_j$  and its internal resistance  $R_j$  between the two points p and q; There are two multiple input signals (i) ( $V_k^p$ )s and (ii) ( $V_l^n$ )s which are really joined at two points 'q' and 'p' respectively .On the top side, each input signal  $V_k^p$  is connected to the point "q" through their respective capacitances  $C_k^p$ ; and each input in lower side, signal  $V_l^n$  is tightly coupled to the point "p" through the respective capacitance  $C_l^n$  as well. For smooth assistanceabout the tunneling of electron, the bias *voltageV\_b*, different from input signal, is connected to the point "q" through the real or true capacitor  $C_b$ . Another capacitance  $C_0$  through which point "p" is grounded. Two capacitances  $C_j$  and  $C_0$  play the important role for determining the output condition in the circuit i.e. they are the decision maker. The operation of the multiple-input one-output TLG is unveiled by a function called *signun* function. The *signun* function s(x) regarding p(x) is described by two equations (1) and (2) as follows[9-14].

$$s(x) = sgn\{p(x)\} = \begin{cases} 0, & if \ p(x) < 0 \\ 1, & if \ p(x) \ge 0 \end{cases}$$
(1)  
$$p(x) = \sum_{k=1}^{n} (w_k \times x_k) - \theta$$
(2)

in equation(2),  $x_k$  represents multiple inputs k=1,2,3...n, and  $w_k$  are their corresponding integer weights.

In fact, the multiple-input one-output TLG in Fig. 3plays role as a comparator. The **Threshold Logic Gate (LTG)** compares two values (i) the threshold value $\theta$  and (ii) the weighted sum of the *inputs*  $\sum_{k=1}^{n} (w_k \times x_k)$ . If the numerical value of the sum $\sum_{k=1}^{n} (w_k \times x_k)$  is greater than or equal in comparison with the threshold value  $\theta$  of the circuit, then the output value of the multiple-input one-output circuit shows high (logical "1") at the output terminal  $V_0$ , otherwise the output value is low (logical "0").

#### VI. 2-INPUT AND GATE HAVING ONE COMPLEMENT INPUT

Our intention is to construct different logic gates on the basis of the generic Multiple-input single-output threshold logic to calculate the node address numbers belonging to the routing path in a computer network, For the purpose of implementing the circuit doing the operation of the Network shown in the Fig. 1, we require to be implemented an XOR gate having two inputs A and B. The XOR gate can be expressed by a Boolean function as  $F(A, B) = XOR(A, B) = \overline{AB} + A\overline{B}$ . From this logical expression  $(\overline{AB} + A\overline{B})$ , it is transparent that it is an expression of OR operation indicated by "+" of two AND expressions  $\overline{AB}$  and  $A\overline{B}$  respectively [2-5]. In implementing, we must think of logic expression  $AND(A,\overline{B})=A\overline{B}$ , and its corresponding threshold logic that can be expressed as the threshold logic expression of a AND gate having two input variables A and  $\overline{B}$  as

$$AND(A, \overline{B}) = sgn\{w_A A + w_B \overline{B} - \theta\}^{\dots} (3)$$

In order to obtain the values of the coupling weights  $w_A$  and  $w_B$  of the variables A and  $\overline{B}$ , we assume A = 1 and  $\overline{B} = 1$ . Then the equation (3) turns to



||Volume 11, Issue 5, May 2022||

DOI:10.15680/IJIRSET.2022.1105047

 $AND(A, \overline{B}) = sgn\{w_A + w_B - \theta\}^{\dots} (4).$ 

To obtain exactly a 2-input threshold logic AND gate, a truth table can be listed in Table-2 and compared the sum of the coupling weights  $w_A$  and  $w_B$  of two variables A and  $\overline{B}$  respectively in equation (4) with the threshold value  $\theta$  [1-5,7,8].

Table-2	Truth	table of	fF(A,	B)	=AB
---------	-------	----------	-------	----	-----

А	Ē	$F(A, \overline{B})$	inequality
		$=A\overline{B}$	
0	0	0	0< <b>θ</b>
0	1	0	$w_B < \theta$
1	0	0	$w_A < \theta$
1	1	1	$w_B + w_A \ge \theta$

For a 2-input AND gate, which is a positive logic, we can assume the weights of A and  $\overline{B}$ , i.e.  $w_A$  and  $w_B$  to be 1 each. Then from the four inequalities in the4<sup>th</sup> column in Table-2, if we take positive integers  $w_B=1$ ,  $w_A=1$  and  $\theta=2$ , then all four inequalities in 4<sup>th</sup> column in Table-2 are satisfied. As a result, the linear threshold logic equation can be expressed as

 $AND(A,\bar{B}) = sgn\{A + \bar{B} - 2\}^{\dots}$ (5)

But we are cognizant of the numerical sum of a logic variable and its complement and it is always 1, so,  $B + \overline{B} = 1$  or  $\overline{B} = -B+1$ ,

Substitute this value ( $\overline{B}=-B+1$ ) into equation (5), we get the equation (6)

 $F(A,B) = AND(A,\overline{B}) = sgn\{A - B - 1\}$ = sgn\{A - B - (1)\}^{(1)}(6)

According to the equation (6), we obtain the value of  $\theta = 1$  and put the input capacitances  $C_1^p$  and  $C_1^n$  for one positive logic input (for A) and one negative input (for B) in the Generic Multiple input threshold logic gate in Fig.1. The parameter values we have measured for this AND(A,B) gate and those are as follows: logic input "0"=0V, logic "1" = 16mV, C=1aF,  $C_1^p = C_1^n = \frac{1}{2}C=0.5aF$ ,  $C_b = 9.5aF$ ,  $C_j = 0.25aF$ ,  $C_L = 9aF$ ,  $C_0 = 9.5aF$ ,  $R_j=10^5\Omega$ ,  $V_b = V_s = 16mV$ . After substituting these values in the Fig.3 we get the two input AND gate which is shown in Fig. 3(c) that is really the simulation arrangement of AB. The simulation result is provided in Fig. 3(d). The threshold logic gate of AND(A, B) = AB is given in Fig. 3(b).



Fig. 3(a) symbol of  $A\overline{B}$  3(b) TI

3(b) TLG based node of  $A\overline{B}$ 



Volume 11, Issue 5, May 2022

DOI:10.15680/IJIRSET.2022.1105047



#### VI. THRESHOLD LOGIC EQUATION FOR OR GATE

Boolean Expression of an OR gate of two variables A and B is given by F(A,B)=A+B, where "+" indicates OR operation. For making a threshold OR logic gate, the truth table of an OR gate required. Comparing the weights of variables  $w_A$  and  $w_B$  of two variables A and B respectively with the threshold or intrinsic value $\theta$ , the truth table of OR gate drawn in Table-3. The threshold logic expression of two input OR gate can be expressed as  $F(A,B)=OR(A,B)=sgn\{w_AA+w_BB-\theta\}$ 

		Table-3	
Α	В	F(A,B)	inequality
0	0	0	0< <del>0</del>
0	1	1	$w_B \ge \theta$
1	0	1	$w_A \ge \theta$
1	1	1	$w_B + w_A \ge \theta$

As  $0 < \theta$  in Table-3,  $\theta$  must be positive, If we assume  $w_B = 1$ ,  $w_A = 1$  and  $\theta = 0.5$ , then the other three inequalities in Table-3 are satisfied. Therefore, the Threshold logic equation OR(A,B)= $sgn\{w_AA+w_BB-\theta\}$  for an OR gate is given in equation (7) and its corresponding threshold logic gate is drawn in Fig. 4(a)



Fig. 4(a) A Threshold logic OR gate

For the purpose of implementing the OR gate, we will select the parameters  $C_1^p = C_2^p = 0.5 \text{ aF}$ ,  $C_b = 9.1 aF$ ,  $C_L = 9aF$ ,  $C_0 = 10aF$ ,  $R_j = 10^5 \Omega$ ,  $V_s = 16mV$  and accordingly after running the simulator, the result we get is given in Fig. 4(c).



Volume 11, Issue 5, May 2022

DOI:10.15680/IJIRSET.2022.1105047



Fig. 4(b) simulation set OR gate



Fig. 4(c) simulation result of OR gate

#### VII. EXCLUSIVE OR (XOR)

An Exclusive-OR gate expression of two variables A and is expressed as  $XOR(A, B) = \overline{AB} + A\overline{B}$  which is not linearly separable, as we know that if a variable X and its complement  $\overline{X}$  exist in a Boolean expression, then this expression is declared as linearly not separable. Hence XOR(A, B) is not linearly separable, as a result it can't be represented by a Linear Threshold Gate (LTG).

When we are interested in expressing XOR(A, B) =  $\overline{AB} + A\overline{B}by$  a threshold logic gate, first we express P = (AB) with the help of threshold gate-based equation as given in equation (5). We can write as-

 $P = sgn \{A + \bar{B} - 2\}^{.....(8)}$ 

We know  $\overline{B} + B = 1$  or  $\overline{B} = -B + 1$ , so equation (8) will be

 $P = sgn \{ A - B - (1) \}^{....(9)}$ 

Now the Boolean expression XOR(A, B) =  $\overline{AB} + A\overline{B}$  can be written as

 $Y = XOR(A, B) = P + \overline{A}B^{\dots}$ (10)

For finding out the threshold gate logic of equation (10), we accept the assistance of the truth Table-4. In this table, four inequalities in 5<sup>th</sup> column are written. Note that in equation  $P = (A\overline{B})$ , there are two variables A and B, though it seems that P is a single variable in equation (10), it would not be correct. Only two variables A and Bare the fact as P is fully depending on A and B. The TLG expression of Y can be written as

$$Y = sgn\{ W_1 A + W_2 B + W_3 P - (\theta) \}^{\dots} (11)$$

According to equation, a truth Table-4 is listed below.

	I able	-4		
А	В	Р	Y	inequality
0	0	0	0	0< <b>θ</b>
0	1	0	1	$W_2 \ge \theta$
1	0	1	1	$W_1 + W_3 \ge \theta$
1	1	0	0	$W_1 + W_2 \le \theta$

We have the solved the inequalities in the 5<sup>th</sup> column of Table-4 and we obtained a solution set {  $W_1$ ,  $W_2$ ,  $W_3$ ;  $\theta$  } = {-1, 1, 2 : 1}. Hence the Threshold equation for the Y is

$$Y = sgn\{ -A + B + 2P - (1) \}^{\dots} (12)$$

By using the TLG expression in equation (12) we will be able to draw the threshold logic circuit of depth 2 as shown in Fig. 5(a).

e-ISSN: 2319-8753, p-ISSN: 2320-6710 www.ijirset.com | Impact Factor: 8.118



Volume 11, Issue 5, May 2022

| DOI:10.15680/IJIRSET.2022.1105047 |

With the concept of the threshold logic gate of XOR(A, B) =  $\overline{AB} + A\overline{B}$ , and the combination of Fig. 3(c) and Fig. 4(b), we have implemented a 2-input XOR in Fig. 5(b) and the simulation result of this XOR gate is given in Fig 5(c).



Fig. 5(a) TLG of XOR(A, B)Fig. 5(b) Set of A BFig. 5(c) Simulation result of A BFig6 Solution space

To verify the linearly separable condition of the equation (10),  $Y = XOR(A, B) = P + \overline{A}B$ , three variables like A, B and P are thought of. No variable takes both the forms of a variable and its complement simultaneously in this expression, so the expression is treated as linearly separable. A truth table regarding(P +  $\overline{A}B$ ) is given in Table-5. When the logical value of the expression is high (1) we represent it by a small red circle in Fig. 6 and when the value of that expression is 0 we represent it by a small colorless circle. From the Fig. 6, it is clear that there are two kinds of circles-- one for Red and the other for colorless circles. These two types of circles of a cubein Fig.6 can be separated by a plane. Hence, the expression P +  $\overline{A}B$  is linearly separable. As a result we would be able to draw a linearly threshold logic gate with depth two and we feel free to draw it in Fig. 5(a).

А	В	$P(A\overline{B})$	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

**Table 5** Truth table of  $Y = (P + \overline{AB})$ 

#### VIII. SIMULATION SET OF ROUTING ON A 8-DIMENSIONAL HYPERCUBE

For the purpose of implementing the Simulation set to predict the routing path between any two nodes, the parameter values of all components necessary must be measured. We have measured them and observed that they satisfy our expectation and they are all given below:

For P=AB: logic input "0"=0V, logic "1" = 16mV, C=1aF,  $C_1^p = C_1^n = \frac{1}{2}C$  =0.5aF,  $C_b = 9.5aF$ ,  $C_j = 0.25aF$ ,  $C_L = 9aF$ ,  $C_0 = 9.5aF$ ,  $R_j = 10^5\Omega$ ,  $V_b = V_s = 16mV$ 

For Y = XOR(A, B) = P +  $\bar{A}B$ : 0"=0V, logic "1" = 16mV, C=1aF,  $C_1^P = 0.5C_2^P = C_1^n = \frac{1}{2}C=0.5aF$ ,  $C_b = 8.5aF$ ,  $C_i = 0.25aF$ ,  $C_L = 9aF$ ,  $C_0 = 9.5aF$ ,  $R_i=10^5\Omega$ ,  $V_b = V_s=16mV$ .

When these parameters are placed in proper positions, the E-cube routing circuit is created and this circuit is given in Fig. 7(a). Three sets of simulation results are given in Fig. 7(b) below. Two sets of input data (source, destination) are put in the input terminals  $(S_7S_6S_5S_4S_3S_2S_1S_0)$  at the left side and  $(d_7d_6d_5d_4d_3d_2d_1d_0)$  at the top of the Fig. 7(a). After

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 8.118|



Volume 11, Issue 5, May 2022

| DOI:10.15680/IJIRSET.2022.1105047 |

a small processing delay, different outputs of 8-bits are taken from the different output nodes like--  $2^{nd}$  node,  $3^{rd}$  node, ...,  $8^{th}$  node and goal node with respect to the Fig.7(a). Three pairs of data are used for simulation results which are as follows.

Set 1: (Source = (s7s6s5s4s3s2s1s0) = 00110011 = 51, **Destination**=(d7d6d5d4d3d2d1d0) = 10000100 = 132)Set 2: (Source = (s7s6s5s4s3s2s1s0) = 00011000 = 24, **Destination**=(d7d6d5d4d3d2d1d0) = 01100100 = 100)Set 3: (Source = (s7s6s5s4s3s2s1s0) = 10001110 = 142, **Destination**=(d7d6d5d4d3d2d1d0) = 00111100 = 60)

After putting the inputs (source and goal) the simulation waveforms we got are given in Fig 7(b).



Fig. 7(b) Simulation result of Fig 7(a) for 3 pairs of (source, destination)

Volume 11, Issue 5, May 2022

DOI:10.15680/LJIRSET.2022.1105047

#### IX. CONCENTRATION, DELAY, SWITCHING ENERGY AND FASTNESS

Any electronic circuit processing data must have a processing delay which varies from one circuit to another. The fundamental logic gates like AND, OR, NOT, XOR and Universal gats are being inevitable to make circuits. In the present work, we are concentrated inmaking a network circuit bearing 256 hosts/nodes, and this circuit operation is implemented by using multiple-input one-output like XOR.As multiple number of XOR gates are used in this circuit, so this circuit have a time delay for any input-output connection. When measured performance/ efficiency of a circuit weather it may be conventional or Nano- or Pico-, some factors like- (i) number of components, (ii) time required for an execution, (ii) speed, (iv) power consumption, (v) fan-outs, (vi) costs of materials, (vii) controlling charge, (viii) atmospheric temperature etc. play the role to measure that. At the time of calculating the processing/switching delay in the threshold logic circuit, we must consider the tunnel junction capacitance  $C_i$ , the internal critical voltage  $V_c$  for a multi-input threshold logic gate we are dealing with. The switching/processing delay can be calculated for that multiple-input circuit by using the logarithmic expression of probable error which is given by the approach[3-7, 13-14] given below in equation (13).

 $Delay = -(e | ln(P_{error}) | R_t) / (|V_i| - V_c)^{\dots} (13)$ 

, here,  $V_c$  = the critical voltage,  $V_i$  is the tunnel junction voltage  $andR_t$  = the internal resistance of the tunnel junction, Execution is done when the tunnel junction voltage  $V_i$  of the tunnel capacitance (as in Fig.3) is being a fraction of the bias voltage  $V_b$  is greater than or equal to the internal critical voltage  $V_c$  of the tunnel junction i.e., when  $|V_i| \ge V_c$ , then a tunneling event occurs.

When a 2-input AND gate fulfil this conditional expression  $|V_i| \ge V_c$ , then the junction voltage of the 2-input AND gate becomes  $V_i = 10.80$ . The internal critical voltage  $V_c$  of the tunnel junction is found to be 11.58mV [4-7]. Given that the probability of error  $P_{error} = 10^{-12}$  and the parameter values of tunnel resistance  $R_t = 10^5 \Omega$ .by substituting these values intoequation (13), we would be able to calculate the gate delay =  $0.062 |\ln (P_{error})| = 1.71$ ns. In the similar way, we can calculate the different circuit delays and they are provided in Table-6.

At the moment of tunneling, an electron crosses through the tunnel-junction barrier, by changing the total energy present in the circuit. The changes of energies happened before and after a tunneling event is measured by the assistance of the equation (14) given below.

 $\Delta E = E_{before\ tunnel} - E_{after\ tunnel} = -e(V_c - |V_j|)$ (14)

This is the tunneling energy  $-e(V_c - |V_i|)$  being consumed during a tunnel event in the tunneling circuit. We listed the energy consumptions for different TLG circuits in Table-6.

		Table-6		
Gate/Device	SET-	LTG-based delay ns	Speed-up	Switching
	based		(times)	Energy
	delay			meV
	ns			
2-input NOR	4	$0.060  \ln(P_{error})  = 1.65$	2.42	10.70
2-input OR	4	$0.062  \ln(P_{error})  = 1.71$	2.33	10.80
2-input NAND	4	$0.062  \ln(P_{error})  = 1.71$	2.33	10.80
2-input XOR	4	$0.080  \ln(P_{error})  = 2.21$	1.81	21.20
3-input OR	8	$0.073  \ln(P_{error})  = 2.01$	3.98	11.58
2 <sup>8</sup> host	36	$0.180  \ln(P_{error})  = 4.9735$	7.23	1526.40
network				

Delays for the CMOS, SET-based and LTG-based are to be compared with. We have found out the processing delays 12ns [5-11, 17-18] for each of the case of a CMOS logic gate like NAND, NOR, AND and XOR. The tunneling delay of an electron in the case of a single electron tunneling transistor (SET) [7-9] is approximately 4ns [4-10, 17-18]. The speed-up of LTG-based delays with respect to that of SET-based delays are also listed in Table-6.



Volume 11, Issue 5, May 2022

#### DOI:10.15680/IJIRSET.2022.1105047

#### X. CONCLUSION

When data packets are travelling through a through a network, there should be a deadlock free condition otherwise some buffers must be included in every host, which is already predicted by our routing path circuit, staying in the routing path. Wewant to avoid waiting or deadlocking situation, a proper algorithm is to be selected for packet routing so as to move them through the channels without halt. When a deterministic algorithm used for routing, then the routing path from starting point to destination point is strongly determined by the 8-bit binary addresses of source point node as well as destination point node without deadlocking. To include the deterministic routing algorithm, "E-cube Routing on an 8-dimensional hypercube" based on an emerging nanotechnology can be constructed— on the principle of threshold logic gate, combinational and sequential logic circuits can be made whenever necessary. The concentration/population density of the TLG based circuits will be high because the tunnel junction size is very small (diameter  $\approx 10$ nm). The power being consumed for Routing on an8-dimensional hypercube is 1526.40 meV =1526.40 × 10<sup>-3</sup> × 1.602 × 10<sup>-19</sup>=2445.29 × 10<sup>-22</sup> Joule, it is really, very small amount. From the view point of the processing delays, it is clear that the multi-input TLG-based circuit is 2 or more times faster than the Single electron transistor based circuit and 7-times faster in comparison with the CMOS based circuit. As the maximum delay in the network simulator circuit is 4.9735ns, the speed of the network will be  $f = \frac{1}{4.9735ns/bit} = 0.201 \times 10^9$  bit/sec =0.201 G-

bit/sec. We have seen that after every 4.9735 ns one desired output is fanned out from output terminals of the 8dimensional hypercube network circuit. No two points are found identical in any column in Table-1. This indicates that there will be no conflict in the network. Therefore, no dead lock in the 8- dimensional hypercube routing process. All those circuits drawn in this work are verified by the simulator-SIMON and seen that the simulated results are squared with the theoretical aspects.

#### REFERENCES

- [1] H. Sullivan and T.R. Bashkow,'A Large Scale Homogeneous, Fully Distributed Parallel Machine," Proc. 4<sup>th</sup>Symp. Computer Arch., vol.5 pp105-125. Mar, 1977
- [2] Kai Hwang & Naresh Jotwani, "Advanced Computer Architecture Parallelism, Scalability, Programmability" Second Edition, Mc Graw Hill.
- [3] Chin-Heng Liu, Chia-Chun Lin *et al*, "Threshold Function Identification by Redundancy Removal and Comprehensive Weight Assignments" IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 38, NO. 12, DECEMBER 2019
- [4] Anup Kumar Biswas, "Integrated-Circuit Random Access Memory based on an emerging Technology—electron tunneling through Tunnel Junction", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Vol. 8 (4) pp.409-424 July-August-2021, Print ISSN: 2395-1990 | Online ISSN : 2394-4099
- [5] Anup Kumar Biswas, "An Accumulator using Electron Tunneling Through Tunnel Junction" International Journal of Scientific Research in Science, Engineering and Technology Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com) Volume 8, Issue 5 Page Number : 163-183, September-October-2021
- [6] A.K. Biswas, "Measuring of an unknown voltage by using single electron transistor based voltmeter", Semiconductor Physics, Quantum Electronics & Optoelectronics(SPQEO), 24 (3), P. 277-287 (2021) August 2021; ISSN 1605-6582 (On-line)
- [7] Anup Kumar Biswas, "A High-Speed Bidirectional Register with Parallel Loading using single electron Threshold Logic Technology", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Vol. 8 (4): pp394-408; July-August-2021, Print ISSN: 2395-1990 | Online ISSN: 2394-4099 (www.ijsrset.com)
- [8] Anup Kumar Biswas, "Integrated-Circuit Random Access Memory based on an emerging Technology—electron tunneling through Tunnel Junction", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Vol. 8 (4) pp.409-424 July-August-2021, Print ISSN: 2395-1990 | Online ISSN : 2394-4099
- [9] Anup Kumar Biswas, "A High-Speed Bidirectional Register with Parallel Loading using single electron Threshold Logic Technology", International Journal of Scientific Research in Science, Engineering and Technology Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (<u>www.ijsrset.com</u>),July-August-2021, 8 (4) : pp394-408

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 8.118



Volume 11, Issue 5, May 2022

#### | DOI:10.15680/IJIRSET.2022.1105047 |

- [10] Anup Kumar Biswas, "Application of single electron threshold logic gates and memory elements to an up-down Counter" International Journal of Creative Research Thoughts (IJCRT) | Volume 9, Issue 6 June 2021 | ISSN: 2320-2882
- [11] Anup Kumar Biswas, -"Implementation of A 4n-Bit Comparator based on IC Type 74L85 using Linear Threshold Gate Tunneling Technology" International Journal of Engineering Research & Technology ISSN: 2278-0181, Vol. 10 Issue 05, May-2021 pp.299-310,
- [12] Anup Kumar Biswas, "State Transition Diagram for A Pipeline Unit based on Single Electron Tunneling" International Journal of Engineering Research & Technology (IJERT) Vol. 10 Issue 04, April-2021pp.325-336, ISSN: 2278-0181
- [13] Anup Kumar Biswas, "Design of A Pipeline for A Fixed-Point Multiplication using Single Electron Tunneling Technology", International Journal of Engineering Research & Technology ISSN: 2278-0181, Vol. 10 Issue 04, April-2021 pp. 86-98,
- [14] A. K. Biswas and S. K. Sarkar: "An arithmetic logic unit of a computer based on single electron transport system" (SPQEO) Semiconductor Physics, Quantum Electronics & Opt-Electronics. 2003. Vol 6. pp 91-96 No.1,
- [15] A.K. Biswas and S. K. Sarkar: -"Error Detection and Debugging on Information in Communication System Using Single Electron Circuit Based Binary Decision Diagram." (SPQEO) Semiconductor Physics Quantum electronics and opt electronics, Vol. 6, pp.1-8, 2003
- [16] C. Lageweg, S. Cot,fan'a, and S. Vassiliadis, "A linear threshold gate implementation in single electron technology," IEEE Computer Society VLSI Workshop, Apr. 2001, pp. 93
- [17] Jacob. Millman and C. C. Halkias; "Integrated Electronics- Analog and Digital Circuits and Systems -second edition" McGraw Hill Education;
- [18] Millman's Electronic Devices & Circuits 4th Edition (English, Paperback, Jacob Millman)
- [19] Xiaola Lin; P.K. McKinley; L.M. Ni, "Deadlock-free multicast wormhole routing in 2-D mesh multicomputers" IEEE Transactions on Parallel and Distributed Systems, Volume: 5, Issue: 8, Aug 1994 Page(s): 793 - 804

#### BIOGRAPHY



ANUP KUMAR BISWAS, is an Assistant Professor in Computer Science and Engineering, Kalyani Govt. Engineering College,West Bengal, India. He defended his Ph.D.[Engg.] thesis in 2005 in the stream of Electronics and Telecommunication Endineering at Jadavpur University. He was a Senior Research Fellow(SRF) in Faculty of Engineering and Technology (FET) from 2002 to 2005 at the Department of Electronics &TC. Dr. Biswas has published 27 papers in national, international journals

and conferences. The area of his scientific interests are computer arithmetic, parallel architectures, nanotechnology, and computer-aided design. He is engaged in research activities and teaching last 18 years.





Impact Factor: 8.118





## INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY





www.ijirset.com