

## Experiment 8

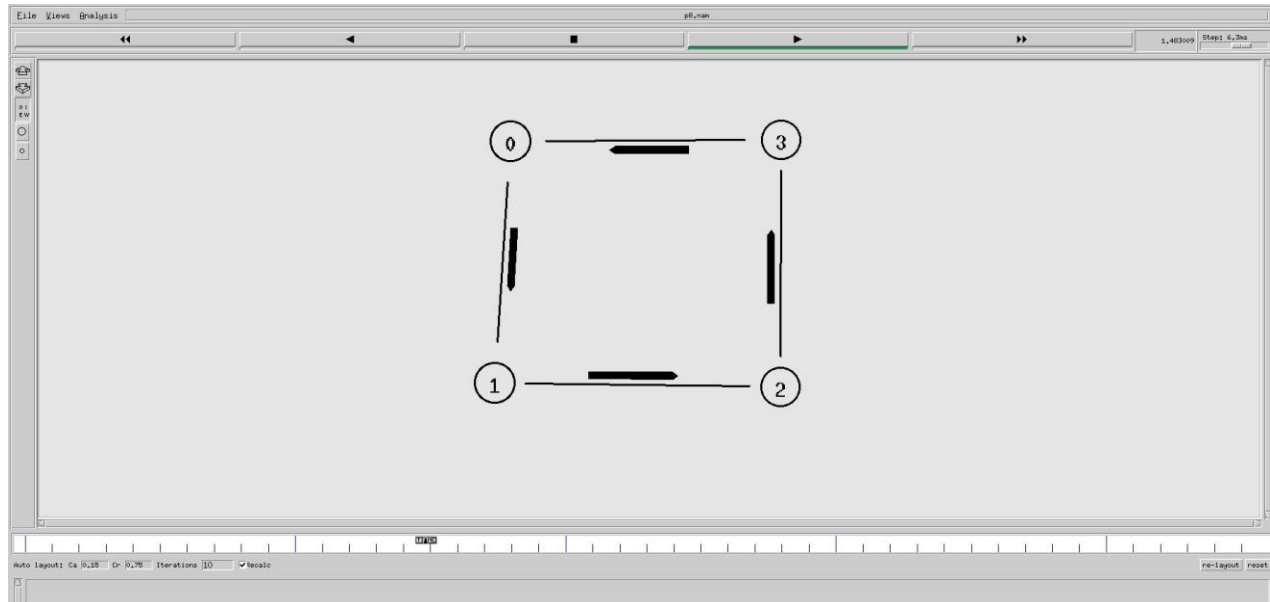
**Aim: “Implement a method of cyclic data transmission using UDP protocol”.**

### TCL code

```
set ns [new Simulator]
set nf [open p8.tr w]
$ns trace-all $nf
set ntrace [open p8.nam w]
$ns namtrace-all $ntrace
for {set i 0} { $i<4 } {incr i} {
set n($i) [$ns node] }
for {set i 0} { $i<4 } {incr i} {
$ns duplex-link $n($i) $n([expr ($i+1)%4]) 1Mb 10ms DropTail }
set udp [new Agent/UDP]
set null [new Agent/Null]
$ns attach-agent $n(0) $udp
$ns attach-agent $n(1) $null
$ns connect $udp $null
set cbr [new Application/Traffic/CBR]
$cbr set interval_ 0.005
$cbr set packetSize_ 500
$cbr attach-agent $udp
set udp1 [new Agent/UDP]
set null1 [new Agent/Null]
$ns attach-agent $n(1) $udp1
$ns attach-agent $n(2) $null1
$ns connect $udp1 $null1
set cbr1 [new Application/Traffic/CBR]
$cbr1 set interval_ 0.005
$cbr1 set packetSize_ 500
$cbr1 attach-agent $udp1
set udp2 [new Agent/UDP]
set null2 [new Agent/Null]
$ns attach-agent $n(2) $udp2
$ns attach-agent $n(3) $null2
$ns connect $udp2 $null2
set cbr2 [new Application/Traffic/CBR]
$cbr2 set interval_ 0.005
```

```
$cbr2 set packetSize_ 500
$cbr2 attach-agent $udp2
set udp3 [new Agent/UDP]
set null3 [new Agent/Null]
$ns attach-agent $n(3) $udp3
$ns attach-agent $n(0) $null3
$ns connect $udp3 $null3
set cbr3 [new Application/Traffic/CBR]
$cbr3 set interval_ 0.005
$cbr3 set packetSize_ 500
$cbr3 attach-agent $udp3
proc Finish { } {
    global ns nf ntrace
    $ns flush-trace
    close $nf
    close $ntrace
    exec nam p8.nam &
    exit 0
}
$ns at 0.5 "$cbr start"
$ns at 4.5 "$cbr stop"
$ns at 0.5 "$cbr1 start"
$ns at 4.5 "$cbr1 stop"
$ns at 0.5 "$cbr2 start"
$ns at 4.5 "$cbr2 stop"
$ns at 0.5 "$cbr3 start"
$ns at 4.5 "$cbr3 stop"
$ns at 5.0 "Finish"
$ns run
```

## NAM Output



## Experiment 9

**Aim: “Implement using C, the error detecting code CRC for 16 bits”.**

### C Program for CRC for 5 bits

```
#include<stdio.h>
#include<string.h>
#define N strlen(g)
//declare the header libraries
char t[50], cs[50], g[50];
int a,e,c;
void xor()
{
    for(c=1;c<N;c++)
        //
        cs[c]=((cs[c]==g[c])?'0':'1');
        //Checking the XOR operation. If both operands are same, then output will be "0"
        otherwise its "1".
}

void crc()
{
    for(e=0;e<N;e++)
        //Consider only first FIVE bits from the modified data
        cs[e]=t[e];
        //Copy those first FIVE bits to CHECKSUM cs[e] from t[e]
    do{
        if(cs[0]=='1')
            //If first leftmost bit is 1 then perform XOR operation
            xor();
            //Calling XOR function
        for(c=0;c<N-1;c++)
            //Performing XOR operation at the first iteration for FIVE bits (0 to N-1)
            cs[c]=cs[c+1];
            //Perform the same for all the data by right shift by 1
        cs[c]=t[e++];
    } while(e<=a+N-1);
    //Continue the operation for the entire data.
}

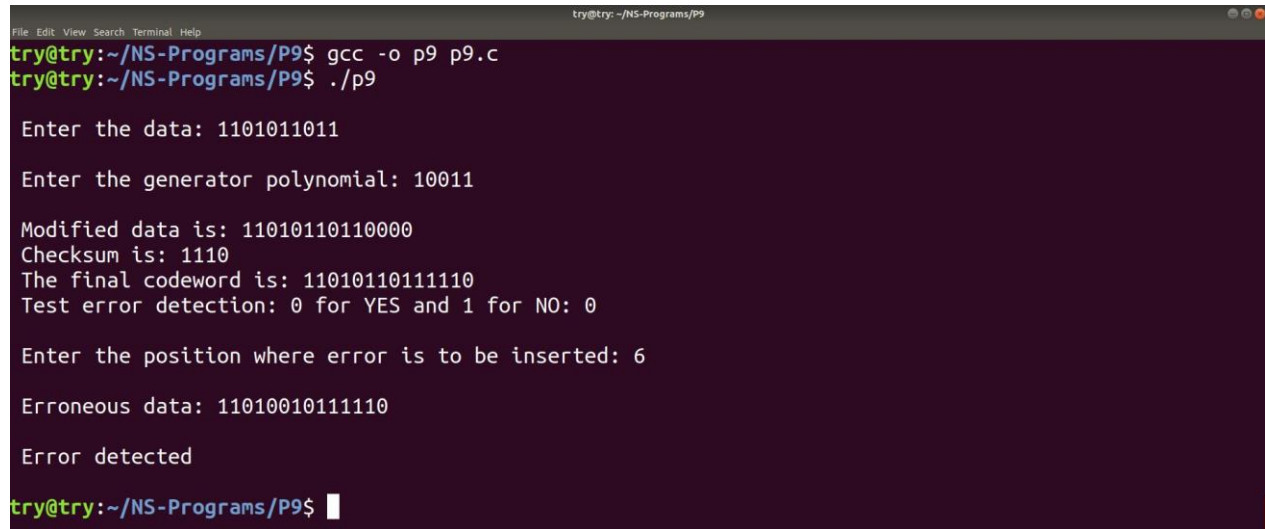
int main(){
    printf("\n Enter the data: ");
    //Enter the data as 1101011011
    scanf("%s", t);
```

```
// Data stored in a string t
printf("\n Enter the generator polynomial: ");
scanf("%s", g);
//Enter the generator polynomial: Since we have hard coded the GP as 10011
a=strlen(t);
// "a" defines the total length of the data
for(e=a;e<a+N-1;e++)
//Appending N-1 zeros to the data where N is the length of the GP
    t[e]='0';
    //t[e] defines appending zeros from e=a;e<a+N-1;e++
printf("\n Modified data is: %s", t);
//MODified data is 11010110110000
crc();
//Call CRC function
printf("\n Checksum is: %s", cs);
//Print the checksum after XOR operation
for(e=a;e<a+N-1;e++)
//To append the checksum value instead of N-1 zeros in total length of the data
    t[e]=cs[e-a];
    //The remodified data with checksum (FINAL CODEWORD)
printf("\n The final codeword is: %s", t);
//Print the final codeword
printf("\n Test error detection: 0 for YES and 1 for NO: ");
//To check for error detection
scanf("%d", &e);
if(e==0)
//If the value of "e" is 0
{
do {
    printf("\n Enter the position where error is to be inserted: ");
    //Specify the position
    scanf("%d", &e);
    //Say for example, e=6
} while(e==0||e>a+N-1);
//WHILE states the boundary, means ranging for 0 to a+N-1

t[e-1] = (t[e-1]=='0')?'1':'0';
//Changing the bit from 0 to 1 and vice versa for error detection
printf("\n Erroneous data: %s\n",t);
}
crc();
for(e=0; (e<N-1)&&(cs[e]!='1'); e++);
//If CHECKSUM is not equal to 1 then error is detected else no error
if(e<N-1)
    printf("\n Error detected \n \n");
```

```
        else
            printf("\n No error detected \n \n");
return 0;
}
```

## Terminal Output

A terminal window titled 'try@try: ~/NS-Programs/P9' showing the execution of a C program. The user enters the data '1101011011' and the generator polynomial '10011'. The program outputs the modified data '11010110110000', the checksum '1110', and the final codeword '1101011011110'. It then asks for the position of an error, and the user enters '6'. The program outputs the erroneous data '1101001011110' and reports 'Error detected'.

```
try@try:~/NS-Programs/P9$ gcc -o p9 p9.c
try@try:~/NS-Programs/P9$ ./p9

Enter the data: 1101011011

Enter the generator polynomial: 10011

Modified data is: 11010110110000
Checksum is: 1110
The final codeword is: 1101011011110
Test error detection: 0 for YES and 1 for NO: 0

Enter the position where error is to be inserted: 6

Erroneous data: 1101001011110

Error detected

try@try:~/NS-Programs/P9$
```

**Note: The above code gives CRC output for 5bits generator polynomial. Students need to develop C code for 16bits generator polynomial CRC.**

## Experiment 10

**Aim: “Implement using C, Hamming Code generation for error detection and correction”**

### C Program

```
#include<stdio.h>
int data[4],encoded[7],edata[7],syn[3];
int gmatrix[4][7]={ { 0,1,1,1,0,0,0},{ 1,0,1,0,1,0,0},{ 1,1,0,0,0,1,0},{ 1,1,1,0,0,0,1 } };
int hmatrix[3][7]={ { 1,0,0,0,1,1,1},{ 0,1,0,1,0,1,1},{ 0,0,1,1,1,0,1 } };
int main(){
int i,j;
printf("Hamming Code encoding\n");
printf("Enter the 4 bit data (one by one): \n");
for(i=0;i<4;i++)scanf("%d",&data[i]);
printf("Generator Matrix\n");
for(i=0;i<4;i++){
for(j=0;j<7;j++){
printf("%d",gmatrix[i][j]);
printf("\n");
}
printf("\n\nEncoded data : ");
for(i=0;i<7;i++){
for(j=0;j<4;j++)encoded[i]^=(data[j]*gmatrix[j][i]);
printf("%d",encoded[i]);
}

printf("\n\nHamming Code Decoding \n\n");
printf("Enter the encoded bit received (one by one) :\n");
for(i=0;i<7;i++)scanf("%d",&edata[i]);

printf("Syndrome = ");
for(i=0;i<3;i++){
for(j=0;j<7;j++)syn[i]^=(edata[j]*hmatrix[i][j]);
printf("%d",syn[i]);
}

for(j=0;j<=7;j++)
if(syn[0]==hmatrix[0][j]&&syn[1]==hmatrix[1][j]&&syn[2]==hmatrix[2][j])break;

if(j==7)printf("\n\nThe code is error free\n");
else{
printf("\n\nError Received at bit no %d of the data\n\n",j+1);
edata[j]!=edata[j];
}
```

```
printf("The correct data should be : ");  
for(i=0;i<7;i++)printf("%d",edata[i]);  
}  
printf("\n\n");  
return 0;  
}
```

**Terminal Output**

```
try@try:~/NS-Programs/P10$ gcc -o p10 p10.c  
try@try:~/NS-Programs/P10$ ./p10  
Hamming Code encoding  
Enter the 4 bit data (one by one):  
1  
0  
0  
1  
Generator Matrix  
0111000  
1010100  
1100010  
1110001  
  
Encoded data : 1001001  
  
Hamming Code Decoding  
  
Enter the encoded bit received (one by one) :  
1  
0  
0  
1  
0  
0  
1  
Syndrome = 000  
  
The code is error free  
  
try@try:~/NS-Programs/P10$
```



## Experiment 12

**Aim: “Simulate a 7-node network to verify Link State routing protocol”.**

### TCL Code

```
set ns [new Simulator]
set namfile [open p12.nam w]
$ns namtrace-all $namfile
set tracefile [open p12.tr w]
$ns trace-all $tracefile
proc finish { } {
    global ns namfile tracefile
    $ns flush-trace
    close $namfile
    close $tracefile
    exec nam p12.nam &
    exit 0
}
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
$ns duplex-link $n0 $n2 1Mb 10ms DropTail
$ns duplex-link $n0 $n3 1Mb 10ms DropTail
$ns duplex-link $n1 $n2 1Mb 10ms DropTail
$ns duplex-link $n1 $n4 1Mb 10ms DropTail
$ns duplex-link $n2 $n4 1Mb 10ms DropTail
$ns duplex-link-op $n0 $n1 orient right
$ns duplex-link-op $n0 $n2 orient right-down
$ns duplex-link-op $n0 $n3 orient down
$ns duplex-link-op $n1 $n2 orient left-down
$ns duplex-link-op $n1 $n4 orient down
$ns duplex-link-op $n2 $n4 orient right-down
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
set null0 [new Agent/Null]
$ns attach-agent $n4 $null0
$ns connect $udp0 $null0
set udp1 [new Agent/UDP]
```

```

$ns attach-agent $n2 $udp1
set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.005
$cbr1 attach-agent $udp1
set null0 [new Agent/Null]
$ns attach-agent $n4 $null0
$ns connect $udp1 $null0

```

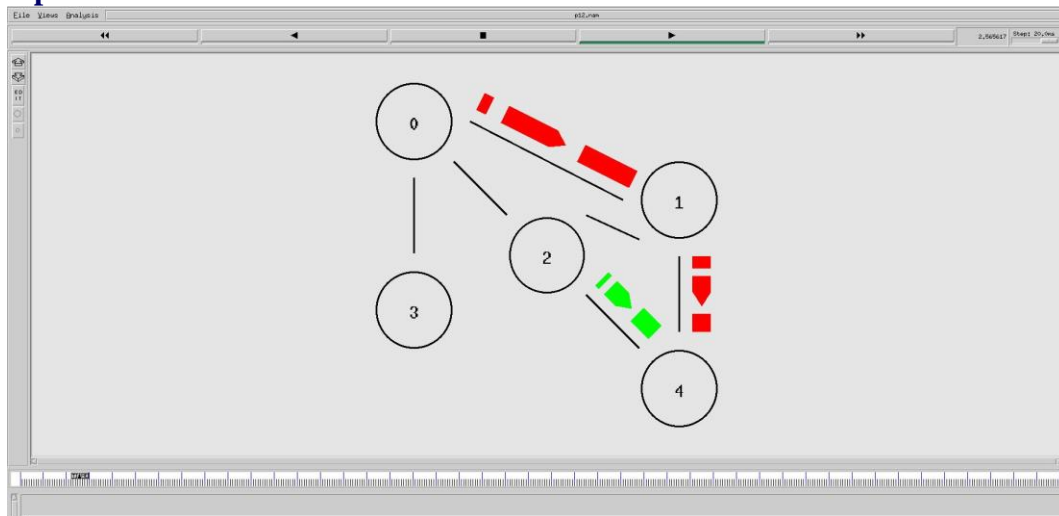
#The Link state routing algorithm is also known as Dijkstra's algorithm which is used to find the shortest path from one node to every other node in the network.

```

$ns rtproto LS
$ns rtmodel-at 20.0 down $n1 $n4
$ns rtmodel-at 23.0 up $n1 $n4
$ns rtmodel-at 25.0 down $n2 $n4
$ns rtmodel-at 40.0 up $n2 $n4
$udp0 set class_ 1
$udp1 set class_ 2
$ns color 1 Red
$ns color 2 Green
$ns at 1.0 "$cbr0 start"
$ns at 2.0 "$cbr1 start"
$ns at 45 "finish"
$ns run

```

### NAM Output



## Viva Questions

1. What are 10Base2, 10Base5 and 10BaseT Ethernet LANs?
2. What is the difference between an unspecified passive open and a fully specified passive open?
3. Explain the function of Transmission Control Block.
4. What is a Management Information Base (MIB)?
5. What is anonymous FTP and why would you use it?
6. What is the front end and back end languages used in NS2
7. Which layer of the 7-layer model provides services to the Application layer over the Session layer connection?
8. What is full form OTCL?
9. What is Point to Point Communication.
10. Which OSI Reference Layer controls application to application communication?
11. What is a DNS resource record?
12. What is the meaning of NAM.
13. What protocol is used by DNS name servers?
14. What is the difference between interior and exterior neighbor gateways?
15. What is the HELLO protocol used for?
16. What are the advantages and disadvantages of the three types of routing? tables
18. What is source route?
19. What is RIP (Routing Information Protocol)?
20. What is SLIP (Serial Line Interface Protocol)?
21. What is Proxy ARP?
22. What is OSPF?
23. What is Kerberos?
24. What is a Multi-homed Host?
25. What is NVT (Network Virtual Terminal)?
26. What is Gateway-to-Gateway protocol?
27. What is BGP (Border Gateway Protocol)?
28. What is autonomous system?
29. What is EGP (Exterior Gateway Protocol)?
30. What is IGP (Interior Gateway Protocol)?
31. What is Mail Gateway?
32. What is wide-mouth frog?
34. What is silly window syndrome?
36. What is multicast routing?
37. What is traffic shaping?
38. What is packet filter?
39. What is virtual path?
40. What is virtual channel?
41. What is logical link control?
42. Why should you care about the OSI Reference Model?
43. What is the difference between routable and non- routable protocols?
44. Name the OS used in your lab to support NS2

45. Explain 5-4-3 rule
46. What is the difference between TFTP and FTP application layer protocols
47. What is the range of addresses in the classes of internet addresses
48. What is the minimum and maximum length of the header in the TCP segment and IP datagram
49. What is difference between ARP and RARP?.
50. What is ICMP?
51. What are the data units at different layers of the TCP / IP protocol suite
52. What is Project 802?
53. What is Bandwidth?
54. Difference between bit rate and baud rate?
55. What is MAC address?
56. What is attenuation?
57. What is cladding?
58. Explain the five components of NS2
59. What is post processing in NS2
60. What is the command used to filter in trace file
61. What is Beaconsing?
62. What is terminal emulation, in which layer it comes?
63. What is frame relay, in which layer it comes?
64. What do you meant by “triple X” in Networks?
65. What is SAP?
66. What is subnet?
67. What is Brouter?
68. How Gateway is different from Routers?
69. What are the different type of networking / internetworking devices?
70. What is mesh network?
71. What is passive topology?
72. What are the important topologies for networks?
73. What are major types of networks and explain?
74. What is Protocol Data Unit?
75. What is difference between baseband and broadband transmission?
76. What are the possible ways of data exchange?
77. What are the types of Transmission media?
78. Difference between the communication and transmission.
79. The Internet Control Message Protocol occurs at what layer of the seven layer model?
80. Which protocol resolves an IP address to a MAC address?
81. MPEG are examples of what layer of the OSI seven-layer model?
82. What is the protocol number for UDP?
83. Which protocol is used for booting diskless workstations?
84. Which layer is responsible for putting 1s and 0s into a logical group?
85. What does ‘P’ mean when running a Trace?
86. UDP works at which layer of the DOD model?
87. What is the default encapsulation of Netware 3.12?
88. Ping uses which Internet layer protocol?

89. Which switching technology can reduce the size of a broadcast domain?
90. What is the first step in data encapsulation?
91. What is the protocol number for TCP?
92. What is the use of Xgraph plotting in NS2
93. Repeaters work at which layer of the OSI model?
94. WAN stands for which of the following?
95. LAN stands for which of the following?
96. DHCP stands for
97. What does the acronym ARP stand for?
98. Which layer is responsible for identifying and establishing the availability of the intended communication partner?
99. Which OSI layer provides mechanical, electrical, procedural for activating maintaining physical link?
100. Define Network?
101. What is a Link?
102. What is a node?
103. What is a gateway or Router?
104. What is point-point link?
105. What is Multiple Access?
106. What is the essence of RSVP ? Explain the suitable example
107. What is the need of scheduling and policing techniques in multimedia networking?
108. What is the need of RTCP protocol along with RTP protocol in multimedia communication?
109. Explain WAN architecture in detail.
110. Explain email architecture and its services.
112. Explain Bluetooth architecture with diagram.
113. Discuss various layers used in ATM architecture.