**Mohd. Aftab Alam**
**02013302717**

# EXPERIMENT 3

**Aim :**

Study of Wired Network using Different Topologies : Ring Topology, Star Topology, Bus Topology.

**Theory :**

The arrangement of a network which comprises of nodes and connecting lines via sender and receiver is referred as network topology. The various network topologies are :

## Ring Topology

A ring network is a network topology in which each node connects to exactly two other nodes, forming a single continuous pathway for signals through each node - a ring. Data travels from node to node, with each node along the way handling every packet. Rings can be unidirectional, with all traffic travelling either clockwise or anticlockwise around the ring, or bidirectional. Because a unidirectional ring topology provides only one pathway between any two nodes, unidirectional ring networks may be disrupted by the failure of a single link. A node failure or cable break might isolate every node attached to the ring.

## Star Topology

A star network is an implementation of a spoke - hub distribution paradigm in computer networks. In a star network, every host is connected to a central hub. In its simplest form, one central hub acts as a conduit to transmit messages. The star network is one of the most common computer network topologies. The hub and hosts, and the transmission lines between them, form a graph with the topology of a star. Data on a star network passes through the hub before continuing to its destination. The hub manages and controls all functions of the network. It also acts as a repeater for the data flow. The star topology reduces the impact of a transmission line failure by independently connecting each host to the hub. Each host may thus communicate with all others by transmitting to, and receiving from, the hub. The failure of a transmission line linking any host to the hub will result in the isolation of that host from all others, but the rest of the network will be unaffected.

## Bus Topology

A bus network is a network topology in which nodes are directly connected to a common half - duplex link called a bus. A host on a bus network is called a station. In a bus network, every station will receive all network traffic, and the traffic generated by each station has equal transmission priority. A bus network forms a single network segment and collision domain. In order for nodes to share the bus, they use a medium access control technology such as carrier - sense multiple access (CSMA) or a bus master. If any link or segment of the bus is severed, all network transmission may fail due to signal reflection caused by the lack of electrical termination.

**Mohd. Aftab Alam**
**02013302717**

**Aim :**

Write a Program for Ring Topology in a Network.

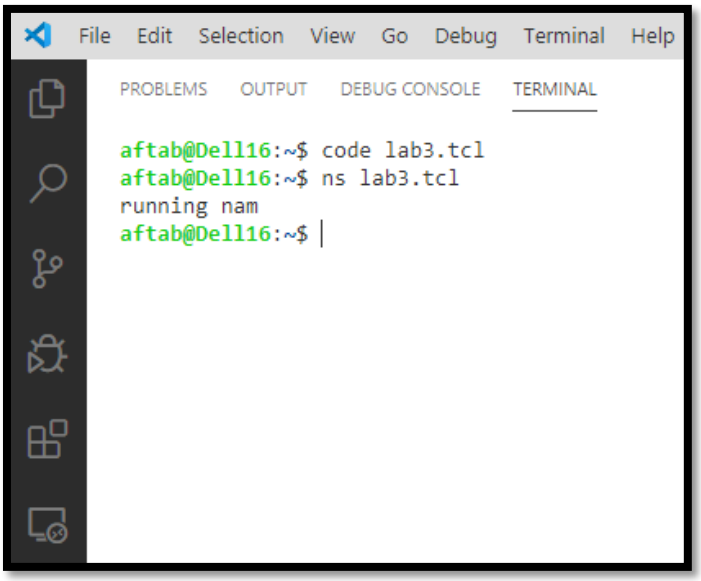
**Code :**

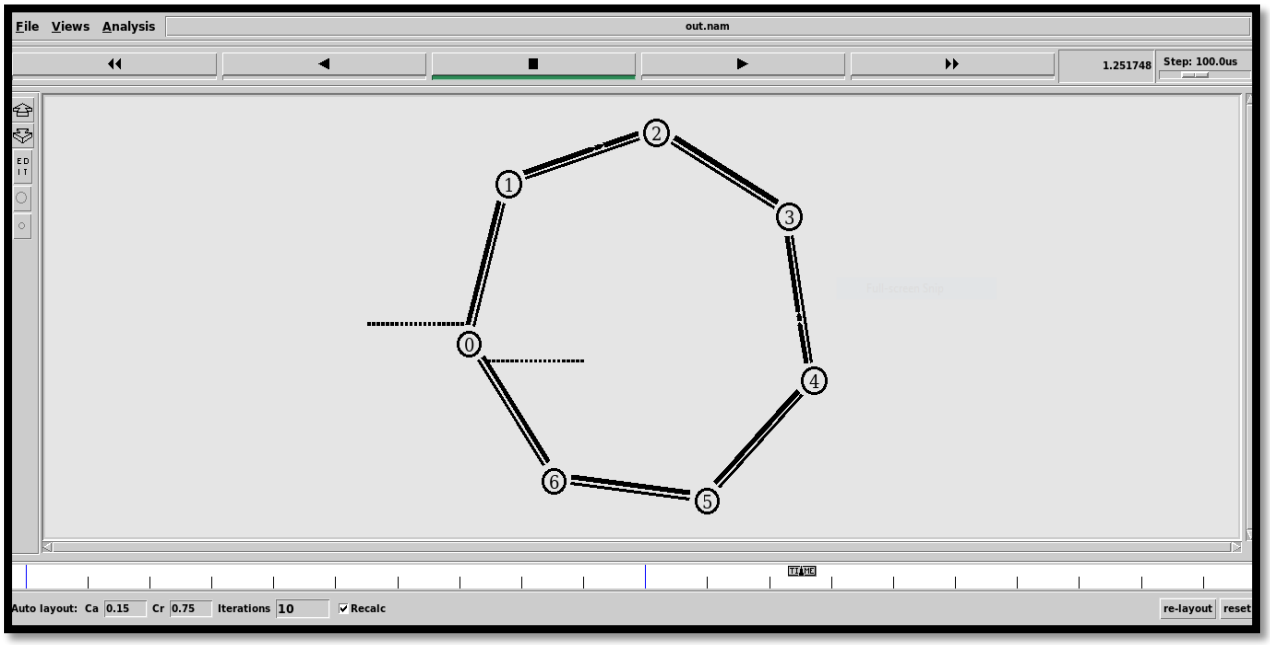
```
set ns [new Simulator]
$ns rtproto DV
set nf [open out.nam w]
$ns namtrace-all $nf
proc finish {} {
global ns nf
$ns flush-trace
close $nf
exec nam out.nam
exit 0 }
#Creating Nodes
for {set i 0} {$i<7} {incr i} {
set n($i) [$ns node] }
#Creating Links
for {set i 0} {$i<7} {incr i} {
$ns duplex-link $n($i) $n([expr ($i+1)%7]) 512Kb 5ms DropTail }
$ns duplex-link-op $n(0) $n(1) queuePos 1
$ns duplex-link-op $n(0) $n(6) queuePos 1
#Creating UDP agent and attching to node 0
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
#Creating Null agent and attaching to node 3
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0
$ns connect $udp0 $null0
#Creating a CBR agent and attaching it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 1024
$cbr0 set interval_ 0.01
$cbr0 attach-agent $udp0
$ns rtmodel-at 0.4 down $n(2) $n(3)
```

**Mohd. Aftab Alam**
**02013302717**

```
$ns rtmodel-at 1.0 up $n(2) $n(3)
$ns at 0.01 "$cbr0 start"
$ns at 1.5 "$cbr0 stop"
$ns at 2.0 "finish"
puts "running nam"
$ns run
```

**Screen Shots :**



**Mohd. Aftab Alam**
**02013302717**

**Aim :**

Write a Program for Star Topology in a Network

**Code :**

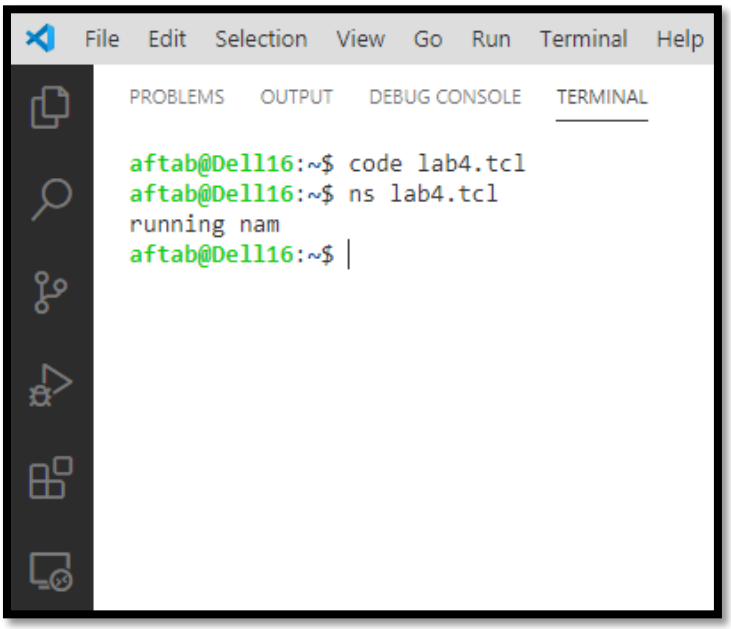
```
set ns [new Simulator]
$ns color 1 blue
$ns color 2 red
$ns rtproto DV
set nf [open out.nam w]
$ns namtrace-all $nf
proc finish {} {
global ns nf
$ns flush-trace
close $nf
exec nam out.nam
exit 0 }
#Creating Nodes
for {set i 0} {$i<7} {incr i} {
set n($i) [$ns node] }
#Creating Links
for {set i 1} {$i<7} {incr i} {
$ns duplex-link $n(0) $n($i) 512Kb 10ms SFQ }
#Orienting Nodes
$ns duplex-link-op $n(0) $n(1) orient left-up
$ns duplex-link-op $n(0) $n(2) orient right-up
$ns duplex-link-op $n(0) $n(3) orient right
$ns duplex-link-op $n(0) $n(4) orient right-down
$ns duplex-link-op $n(0) $n(5) orient left-down
$ns duplex-link-op $n(0) $n(6) orient left
#TCP_Config
set tcp0 [new Agent/TCP]
$tcp0 set class_ 1
$ns attach-agent $n(1) $tcp0
```

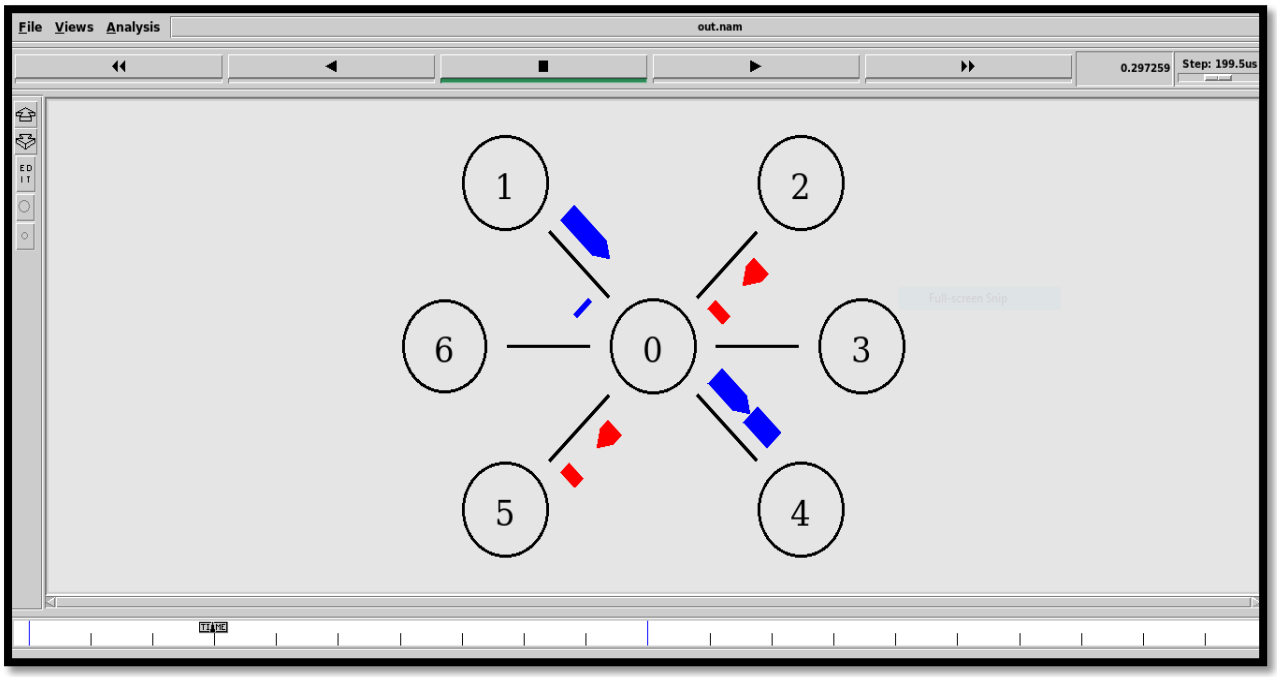
**Mohd. Aftab Alam**
**02013302717**

```
set sink0 [new Agent/TCPSink]
$ns attach-agent $n(4) $sink0
$ns connect $tcp0 $sink0

#UDP_Config
set udp0 [new Agent/UDP]
$udp0 set class_ 2
$ns attach-agent $n(2) $udp0
set null0 [new Agent/Null]
$ns attach-agent $n(5) $null0
$ns connect $udp0 $null0

#CBR_Config
set cbr0 [new Application/Traffic/CBR]
$cbr0 set rate_ 256Kb
$cbr0 attach-agent $udp0

#FTP_Config
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0
#Scheduling Events
$ns rtmodel-at 0.5 down $n(0) $n(5)
$ns rtmodel-at 0.9 up $n(0) $n(5)
$ns rtmodel-at 0.7 down $n(0) $n(4)
$ns rtmodel-at 1.2 up $n(0) $n(4)
$ns at 0.1 "$ftp0 start"
$ns at 1.5 "$ftp0 stop"
$ns at 0.2 "$cbr0 start"
$ns at 1.3 "$cbr0 stop"
$ns at 2.0 "finish"
puts "running nam"
$ns run
```

**Mohd. Aftab Alam**
**02013302717**

**Screen Shots :**



**Mohd. Aftab Alam**
**02013302717**

**Aim :**

Write a Program for Bus Topology in a Network

**Code :**

```
set ns [new Simulator]

set nf [open out.nam w]

$ns namtrace-all $nf

proc finish {} {

global ns nf

$ns flush-trace

close $nf

exec nam out.nam &

exit 0 }

#Create five nodes

set n0 [$ns node]

set n1 [$ns node]

set n2 [$ns node]

set n3 [$ns node]

set n4 [$ns node]

#LAN between the nodes

set lan0 [$ns newLan "$n0 $n1 $n2 $n3 $n4" 0.5Mb 40ms LL Queue/DropTail MAC/Csma/Cd
Channel]

#Create a TCP agent and attach it to node n0

set tcp0 [new Agent/TCP]

$tcp0 set class_ 1

$ns attach-agent $n1 $tcp0

#Create a TCP Sink agent (a traffic sink) for TCP and attach it to node n3

set sink0 [new Agent/TCPSink]

$ns attach-agent $n3 $sink0

#Connect the traffic sources with the traffic sink

$ns connect $tcp0 $sink0

#Create a CBR traffic source and attach it to tcp0

set cbr0 [new Application/Traffic/CBR]

$cbr0 set packetSize_ 500

$cbr0 set interval_ 0.01

$cbr0 attach-agent $tcp0
```

**Mohd. Aftab Alam**
**02013302717**

```
#Schedule events for the CBR agents
$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"
#Call the finish procedure after 5 seconds of simulation time
$ns at 5.0 "finish"
puts "running nam"
$ns run
```

**Screen Shots :**