LAB MANUAL

EC206PPC10

Data Communication and Computer Networks Lab

Bachelor of Technology
in
Electronics & Communication Engineering

Department of Electronics & Communication Engineering
School of Studies of Engineering & Technology
Guru Ghasidas Vishwavidyalaya
Bilaspur-495009 (C. G.)
Website: www.ggu.ac.in

SCHOOL OF STUDIES OF ENGINEERING & TECHNOLOGY
GURU GHASIDAS VISHWAVIDYALAYA, BILASPUR (C.G.)

(A CENTRAL UNIVERSITY)

CBCS-NEW SYLLABUS

B. TECH. THIRD YEAR (Electronics and Communication Engineering)

Vision and Mission of the Institute

Vision		To be a leading technological institute that imparts transformative education to create globally competent technologists, entrepreneurs, researchers and leaders for a sustainable society
Mission	1	To create an ambience of teaching learning through transformative education for future leaders with professional skills, ethics, and conduct.
	2	To identify and develop sustainable research solutions for the local and global needs.
	3	To build a bridge between the academia, industry and society to promote entrepreneurial skills and spirit

Vision and Mission of the Department

Vision		The Department endeavours for academic excellence in Electronics & Communication Engineering by imparting in depth knowledge to the students, facilitating research activities and cater to the ever-changing industrial demands, global and societal needs with leadership qualities.
Mission	1	To be the epitome of academic rigour, flexible to accommodate every student and faculty for basic, current and future technologies in Electronics and Communication Engineering with professional ethics.
	2	To develop an advanced research centre for local & global needs.
	3	To mitigate the gap between academia, industry & societal needs through entrepreneurial and leadership promotion.

Program Educational Objectives (PEOs)

The graduate of the Electronics and Communication Engineering Program will

PEO1: Have fundamental and progressive knowledge along with research initiatives in the field of Electronics & Communication Engineering.

PEO2: Be capable to contrive solutions for electronic & communication systems for real world applications which are technically achievable and economically feasible leading to academia, industry, government and social benefits.

PEO3: Have performed effectively in a multi-disciplinary environment and have self-learning & self-perceptive skills for higher studies, professional career or entrepreneurial endeavors to be confronted with a number of difficulties.

PEO4: Attain team spirit, communication skills, ethical and professional attitude for lifelong learning.

Programme Outcomes: Graduates will be able to:

PO1: Fundamentals: Apply knowledge of mathematics, science and engineering.

PO2: Problem analysis: Identify, formulate and solve real time engineering problems using first principles.

PO3: Design: Design engineering systems complying with public health, safety, cultural, societal and environmental considerations

PO4: Investigation: Investigate complex problems by analysis and interpreting the data to synthesize valid solution.

PO5: Tools: Predict and model by using creative techniques, skills and IT tools necessary for modern engineering practice.

PO6: Society: Apply the knowledge to assess societal, health, safety, legal and cultural issues for practicing engineering profession.

PO7: Environment: Understand the importance of the environment for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics, and responsibilities and norms of the engineering practice.

PO9: Teamwork: Function effectively as an individual and as a member or leader in diverse teams and multidisciplinary settings.

PO10: Communication: Communicate effectively by presentations and writing reports.

PO11: Management: Manage projects in multidisciplinary environments as member or a team leader.

PO12: Life-long learning: Engage in independent lifelong learning in the broadest context of technological change.

Programme Specific Outcomes:

PSO1: Identify, formulate and apply concepts acquired through Electronics & Communication Engineering courses to the real-world applications.

PSO2: Design and implement products using the cutting-edge software and hardware tools to attain skills for analyzing and developing subsystem/processes.

PSO3: Ability to adapt and comprehend the technology advancement in research and contemporary industry demands with demonstration of leadership qualities and betterment of organization, environment and society.

Sub Code	L	T	P	Duration	IA	ESE	Total	Credits
EC206PPC10	**-**	**-**	**2**	**2 Hours**	**30**	**20**	**50**	**1**

DATA COMMUNICATION AND COMPUTER NETWORK LAB

Course Objectives:

- Channel capacity theorem and its analysis.
- Details of ethernet and network topologies.
- Details of different network protocols.

Course Outcomes:

At the end of the course, the students will able to:

CO1 Analyze channel capacity and its analysis.
CO2 Comprehend different types of ethernet and its design.
CO3 Comprehend different types of network topologies and its working.
CO4 Illustrates different types of flow control methods and its working
CO5 Illustrates different other network protocols.

Course Outcomes and their mapping with Program Outcomes & Program Specific Outcomes:

CO	PO												PSO		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	2	2	3							3	3	2	1
CO2	3	1	2	2	3							3	3	2	1
CO3	3	1	2	2	3							3	3	2	2
CO4	3	1	2	2	3							3	3	2	2
CO5	3	1	2	2	3							3	3	2	2

Weightage: **1-Sightly; 2-Moderately; 3-Strongly**

LIST OF EXPERIMENTS:

Experiment No. -01

Objective: Introduction and installation of cisco packet tracer and wireshark

Resources Reqruired: PC/Laptop , cisco packet tracer setup, wireshark software

Theory:

Packet Tracer is a protocol simulator developed by Dennis Frezzo and his team at Cisco Systems. Packet Tracer (PT) is a powerful and dynamic tool that displays the various protocols used in networking, in either Real Time or Simulation mode. This includes layer 2 protocols such as Ethernet and PPP, layer 3 protocols such as IP, ICMP, and ARP, and layer 4 protocols such as TCP and UDP. Routing protocols can also be traced.

Cisco Packet Tracer is Cisco's simulation software. It can be used to create complicated network typologies, as well as to test and simulate abstract networking concepts. It acts as a playground for you to explore networking and the experience is very close to what you see in computer networks. Packet Tracer allows users to drag and drop routers, switches, and other network devices to create simulated network topologies.

Key Features:

- Unlimited devices
- E-learning
- Customize single/multi user activities
- Interactive Environment
- Visualizing Networks
- Real-time mode and Simulation mode
- Self-paced
- Supports majority of networking protocols
- International language support
- Cross platform compatibility

Wireshark is a network protocol analyzer, or an application that captures packets from a network connection, such as from your computer to your home office or the internet. Packet is the name given to a discrete unit of data in a typical Ethernet network. Wireshark does three things:

1.Packet Capture: Wireshark listens to a network connection in real time and then grabs entire streams of traffic – quite possibly tens of thousands of packets at a time.

2.Filtering: Wireshark is capable of slicing and dicing all of this random live data using filters. By applying a filter, you can obtain just the information you need to see. **3.Visualization:** Wireshark, like any good packet sniffer, allows you to dive right into the very middle of a network packet. It also allows you to visualize entire conversations and network streams.

Procedure:-

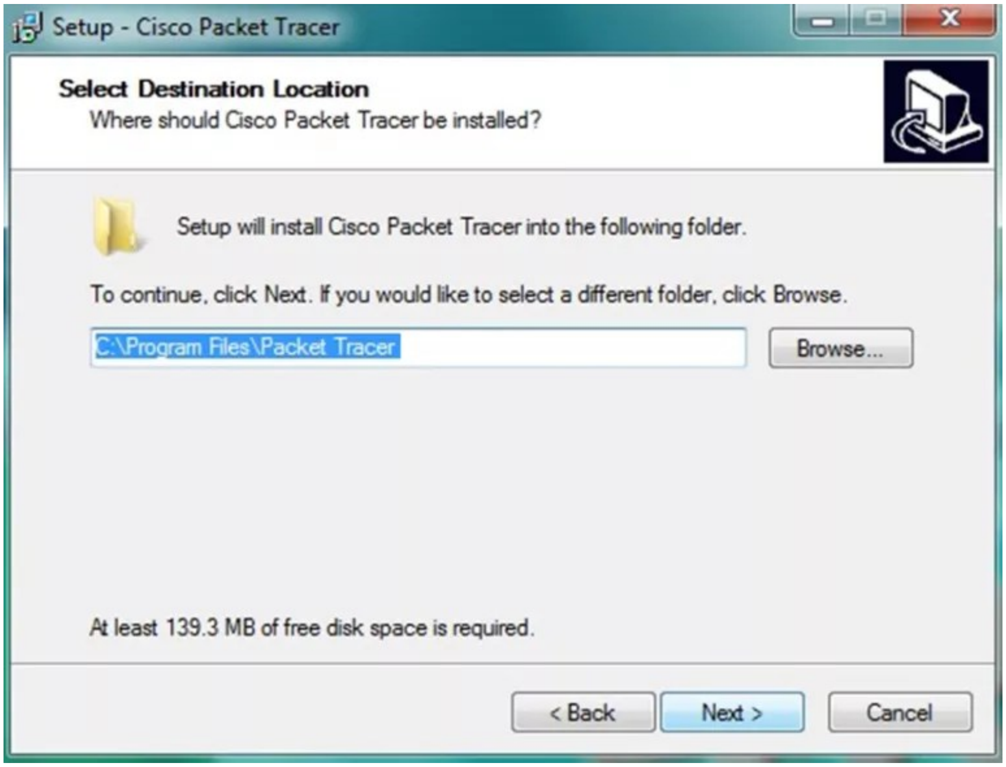
STEP 1 –

After **Cisco Packet Tracer download**, click on the downloaded exe file. Once below Window will appear, click the "Next" option

STEP 2 –

On the next screen, select "I accept the agreement" and click on "Next".

STEP 3-Setup will show the folder in which the program's shortcuts will be created. If you want to change the folder, you can change it. Click on "Next".



Setup - Cisco Packet Tracer

Select Destination Location

Where should Cisco Packet Tracer be installed?

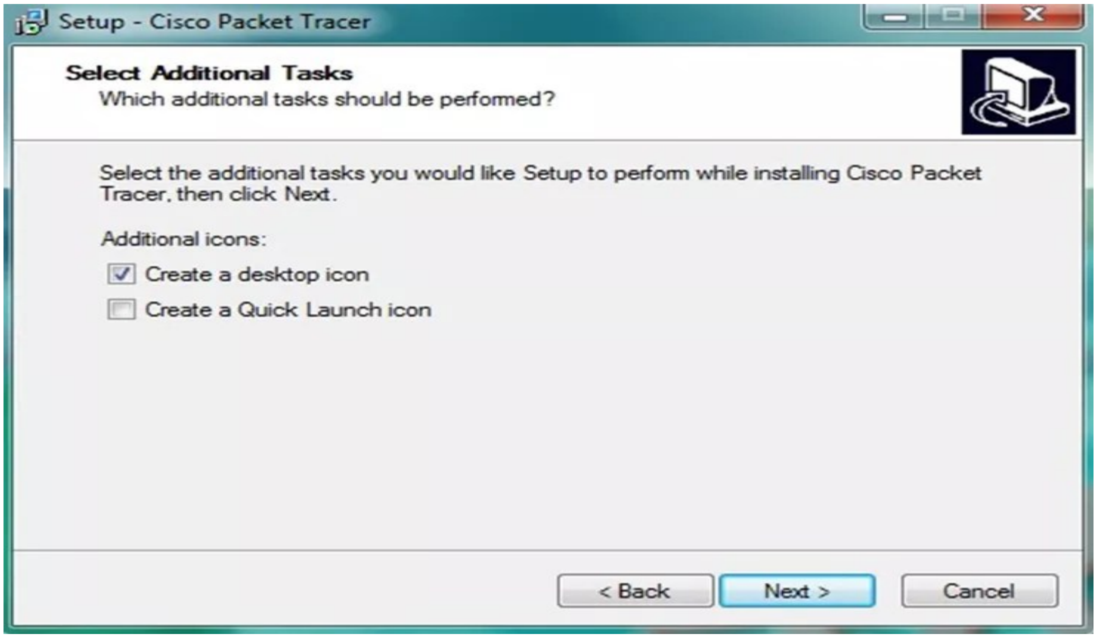
Setup will install Cisco Packet Tracer into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files\Packet Tracer

At least 139.3 MB of free disk space is required.

STEP 4- Then the program will ask whether to create a Desktop icon and create a Quick Launch icon. Make your choice and click on "Next".



Setup - Cisco Packet Tracer

Select Additional Tasks

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Cisco Packet Tracer, then click Next.

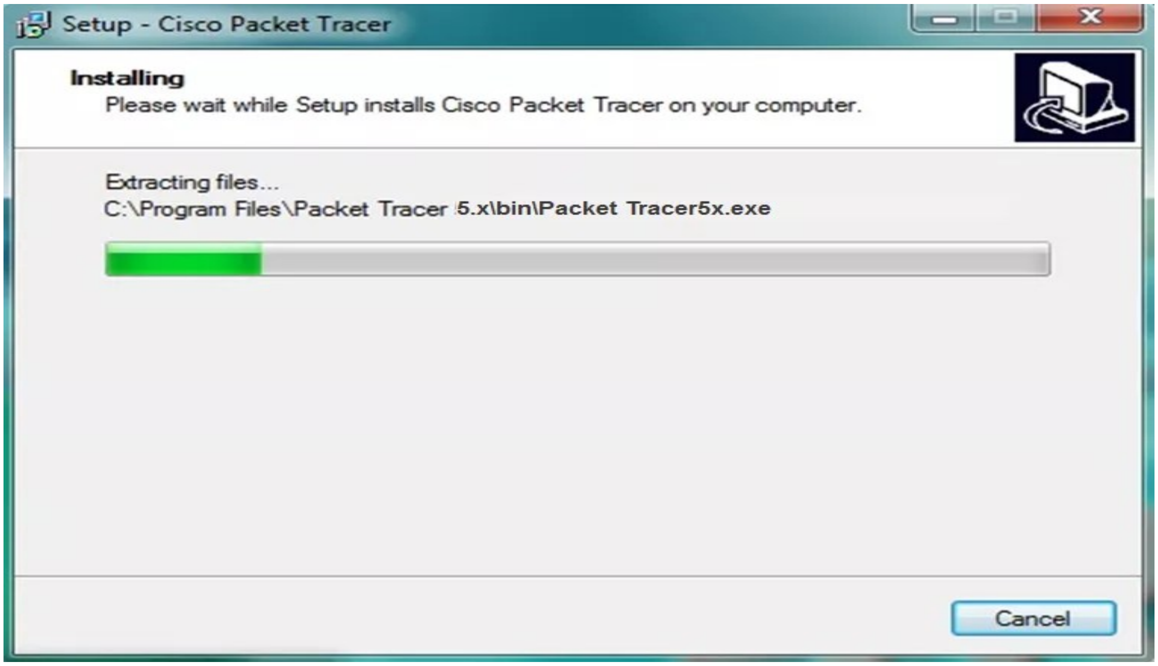
Additional icons:

☑ Create a desktop icon

□ Create a Quick Launch icon

STEP 5-The summary of the settings we selected is displayed. Click on *"Install"*.

STEP 6-The cisco packet tracer installation starts as shown below.



Setup - Cisco Packet Tracer

Installing

Please wait while Setup installs Cisco Packet Tracer on your computer.

Extracting files...

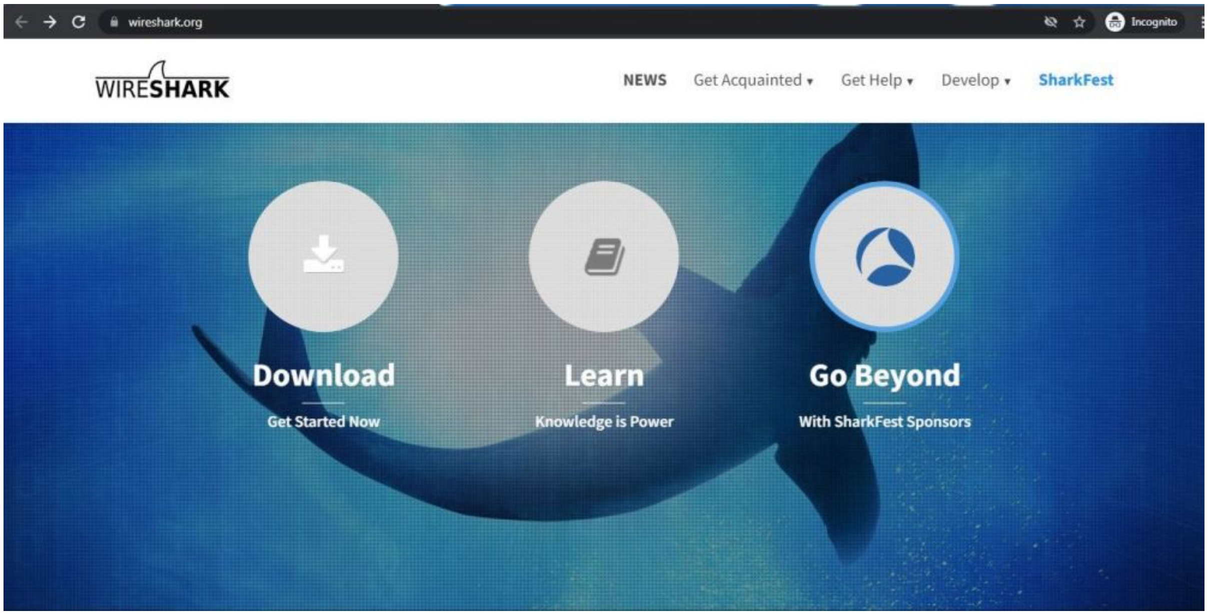
C:\Program Files\Packet Tracer 5.x\bin\Packet Tracer5x.exe

STEP 7- Cisco packet tracer Installation gets completed and the below screen is shown. Click on "Finish".Click "OK" on next popup asking you to close or restart your computer.

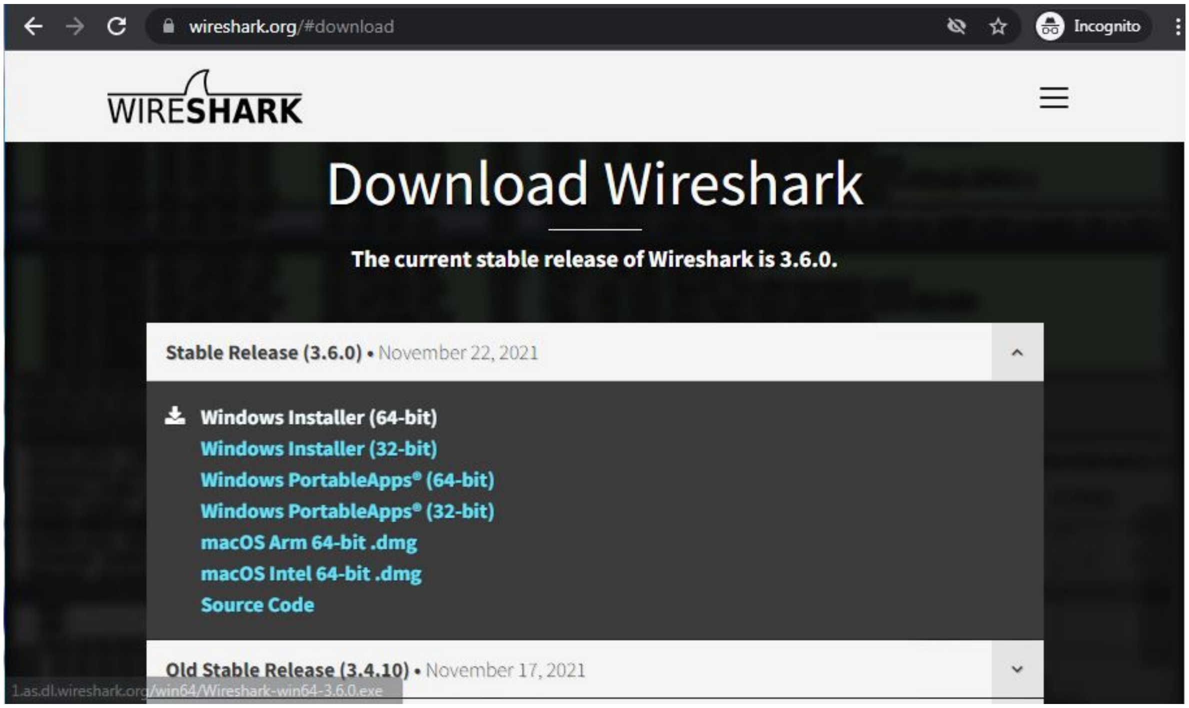
STEP 8- Packet Tracer is installed and ready to be used.

STEPS FOR DOWNLOADING WIRESHARK SOFTWARE:

Step 1: visit the official Wireshark website in any web browser



Step 2: Click on download, a new webpage will open with different installers of wireshark.



Download Wireshark

The current stable release of Wireshark is 3.6.0.

Stable Release (3.6.0) • November 22, 2021

- Windows Installer (64-bit)
- Windows Installer (32-bit)
- Windows PortableApps® (64-bit)
- Windows PortableApps® (32-bit)
- macOS Arm 64-bit .dmg
- macOS Intel 64-bit .dmg
- Source Code

Old Stable Release (3.4.10) • November 17, 2021

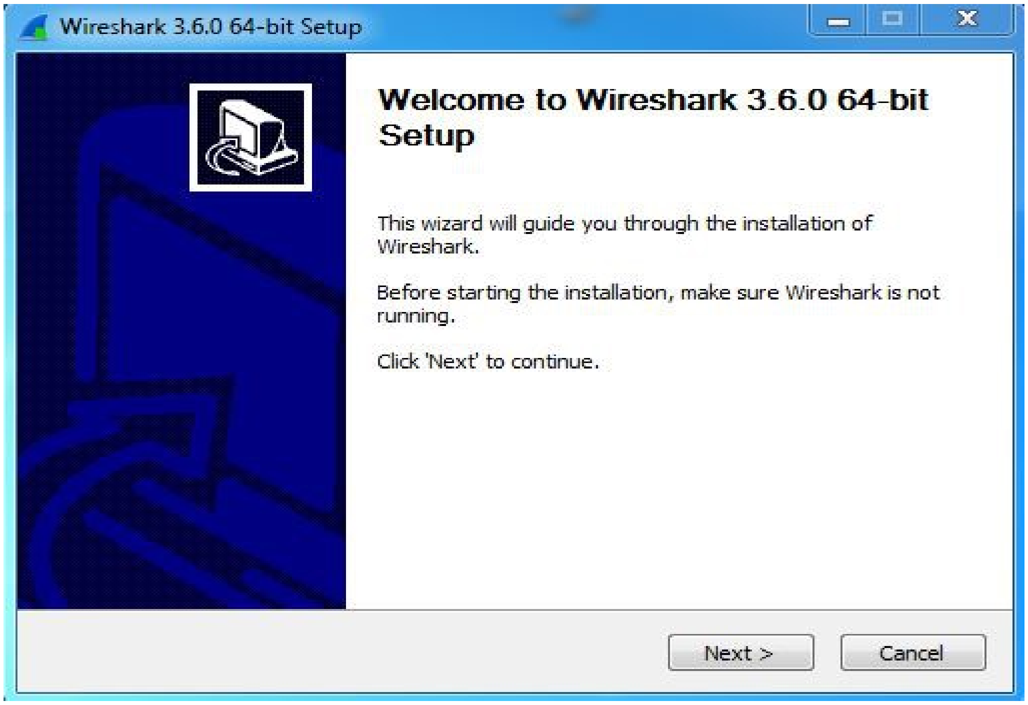
1.as.dl.wireshark.org/win64/Wireshark-win64-3.6.0.exe

Step 3: Downloading of the executable file will start shortly. It is a small 73.69 Mb file that will take some time

Step 4: Now check for the executable file in downloads in your system and run it.

Step 5: It will prompt confirmation to make changes to your system. Clicl on yes.

Step 6: Setup screen will appear, click on Next.


Wireshark 3.6.0 64-bit Setup

Welcome to Wireshark 3.6.0 64-bit Setup

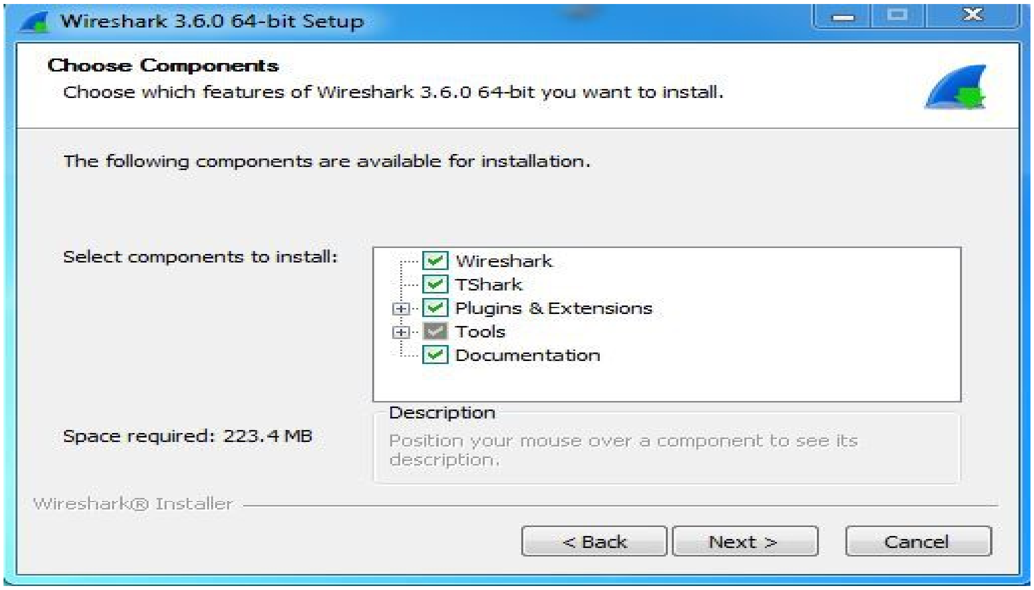
This wizard will guide you through the installation of Wireshark.

Before starting the installation, make sure Wireshark is not running.

Click 'Next' to continue.

Step 7: the next screen will be of License Agreement, click on Noted.

Step 8: This is for choosing components, all components are already marked so don't change anything just click on the Next button.


Wireshark 3.6.0 64-bit Setup

Choose Components

Choose which features of Wireshark 3.6.0 64-bit you want to install.

The following components are available for installation.

Select components to install:

- ☑ Wireshark
- ☑ TShark
- ☑ Plugins & Extensions
- ☑ Tools
- ☑ Documentation

Space required: 223.4 MB

Description

Position your mouse over a component to see its description.

Wireshark® Installer

Step 9: This screen is of choosing shortcuts like start menu or desktop icon along with file extensions which can be intercepted by wireshark, tick all boxes and click on Next button.

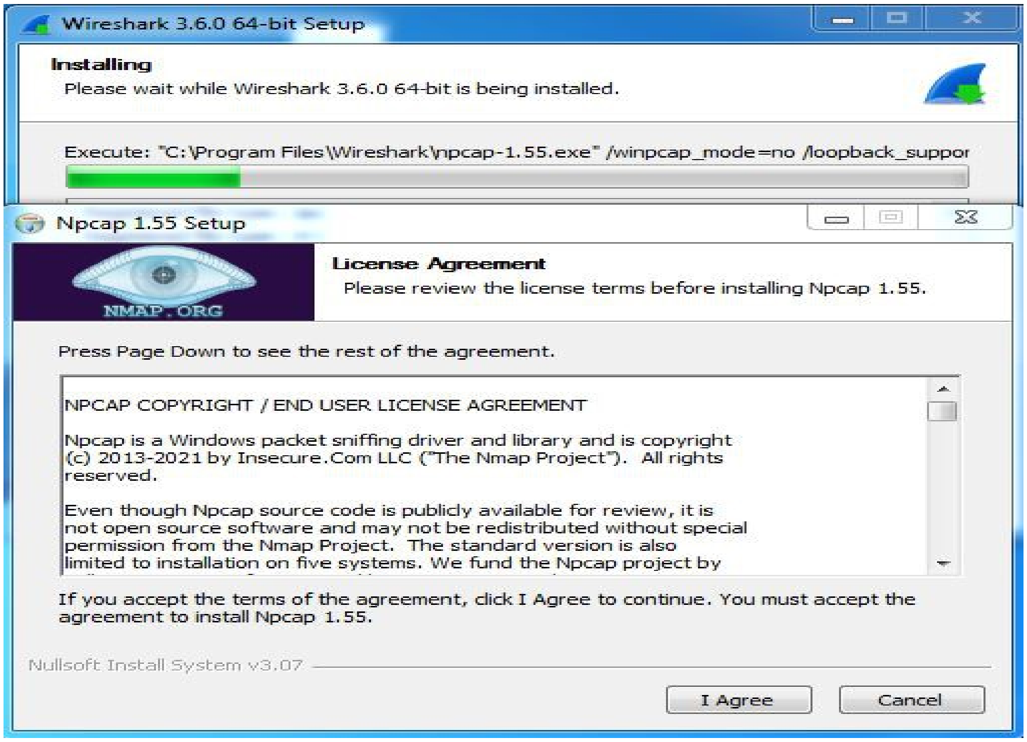
Step 10:The next screen will be of installing location so choose the drive which will have sufficient memory space for installation. It needed only a memory space of 223.4 MB.

Step 11: Next screen has an option to install Npcap which is used with Wireshark to capture packets *pcap*means packet capture so the install option is already checked don't change anything and click the next button.

Step 12: Next screen is about USB network capturing so it is one's choice to use it or not, click on Install.

Step 13: After this installation process will start.

Step 14: This installation will prompt for Npcap installation as already checked so the license agreement of Npcap will appear to click on the *I Agree* button.



Installing
Please wait while Wireshark 3.6.0 64-bit is being installed.

Execute: "C:\Program Files\Wireshark\npcap-1.55.exe" /winpcap_mode=no /loopback_suppor

License Agreement
Please review the license terms before installing Npcap 1.55.

Press Page Down to see the rest of the agreement.

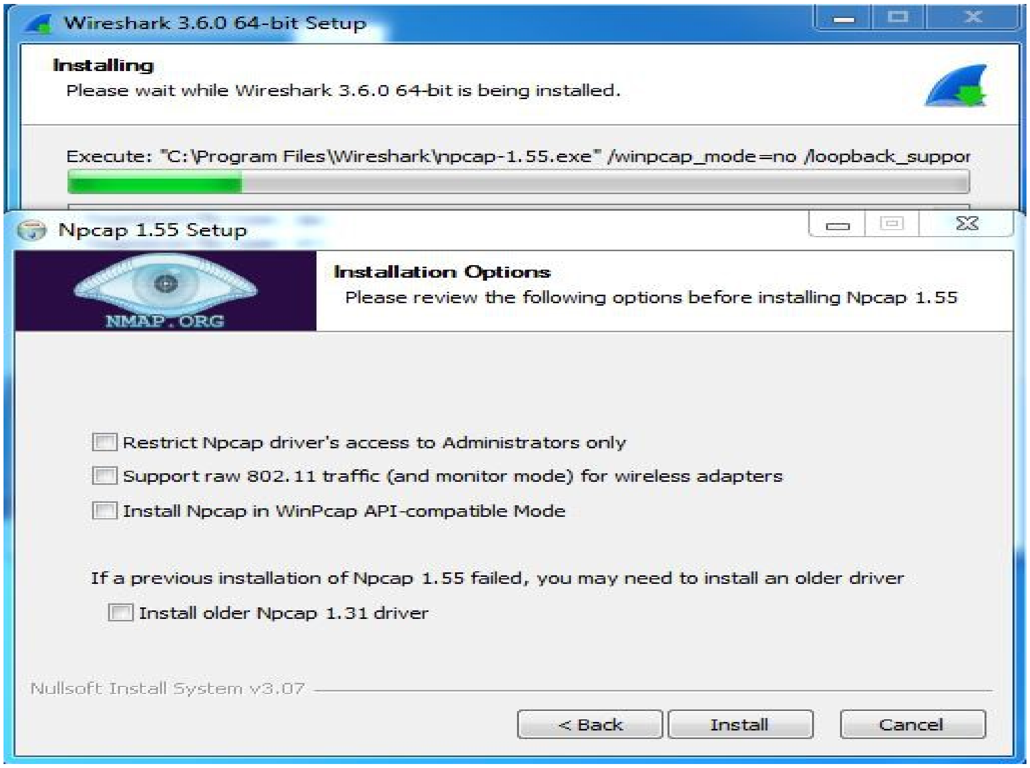
NPCAP COPYRIGHT / END USER LICENSE AGREEMENT

Npcap is a Windows packet sniffing driver and library and is copyright (c) 2013-2021 by Insecure.Com LLC ("The Nmap Project"). All rights reserved.

Even though Npcap source code is publicly available for review, it is not open source software and may not be redistributed without special permission from the Nmap Project. The standard version is also limited to installation on five systems. We fund the Npcap project by

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Npcap 1.55.

Step 15: Next screen is about different installing options of *npcap*, don't do anything click on Install.



Installing
Please wait while Wireshark 3.6.0 64-bit is being installed.

Execute: "C:\Program Files\Wireshark\npcap-1.55.exe" /winpcap_mode=no /loopback_suppor

Installation Options
Please review the following options before installing Npcap 1.55

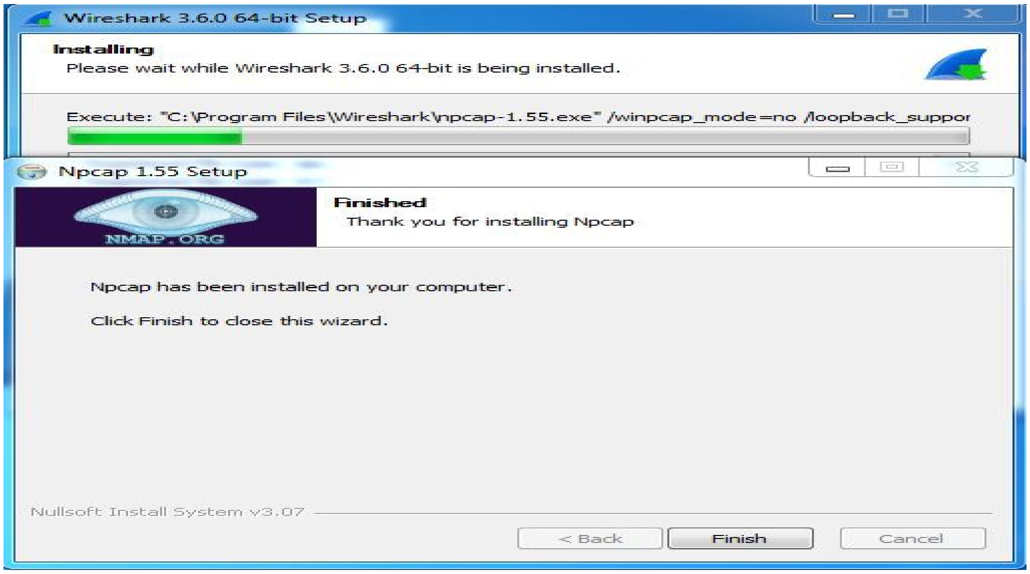
□ Restrict Npcap driver's access to Administrators only
□ Support raw 802.11 traffic (and monitor mode) for wireless adapters
□ Install Npcap in WinPcap API-compatible Mode

If a previous installation of Npcap 1.55 failed, you may need to install an older driver
□ Install older Npcap 1.31 driver

Step 16: After this installation process will start which will take only a minute.

Step 17: After this installation process will complete click on the Next button.

Step 18: Click on Finish after the installation process is complete.



Wireshark 3.6.0 64-bit Setup

Installing

Please wait while Wireshark 3.6.0 64-bit is being installed.

Execute: "C:\Program Files\Wireshark\npcap-1.55.exe" /winpcap_mode=no /loopback_suppor

Npcap 1.55 Setup

Finished

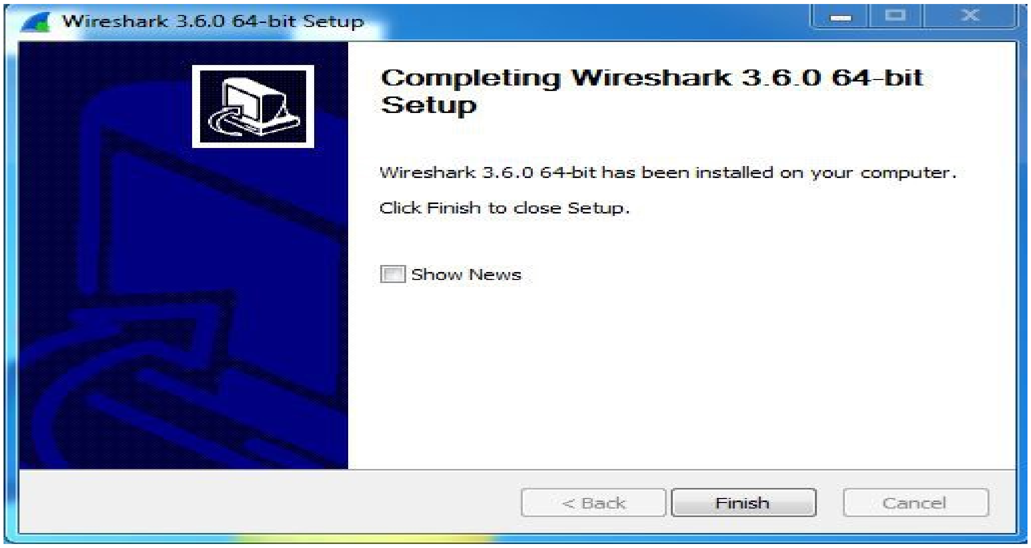
Thank you for installing Npcap

Npcap has been installed on your computer.

Click Finish to close this wizard.

Step 19: After this installation process of Wireshark will complete click on the Next button.

Step 20: Click on Finish after the installation process of Wireshark is complete



Wireshark 3.6.0 64-bit Setup

Completing Wireshark 3.6.0 64-bit Setup

Wireshark 3.6.0 64-bit has been installed on your computer.

Click Finish to close Setup.

□ Show News

Conclusion:

We have learnt installation of cisco packet tracer and wire shark.

Experiment No. -02

Objective: a) Study of types of connection links

b) Introduction of Hub, switch,routers.

Resources Required:

Laptop, Cisco packet tracer

Theory:

a) Study of types of connection links

Types of Connections:

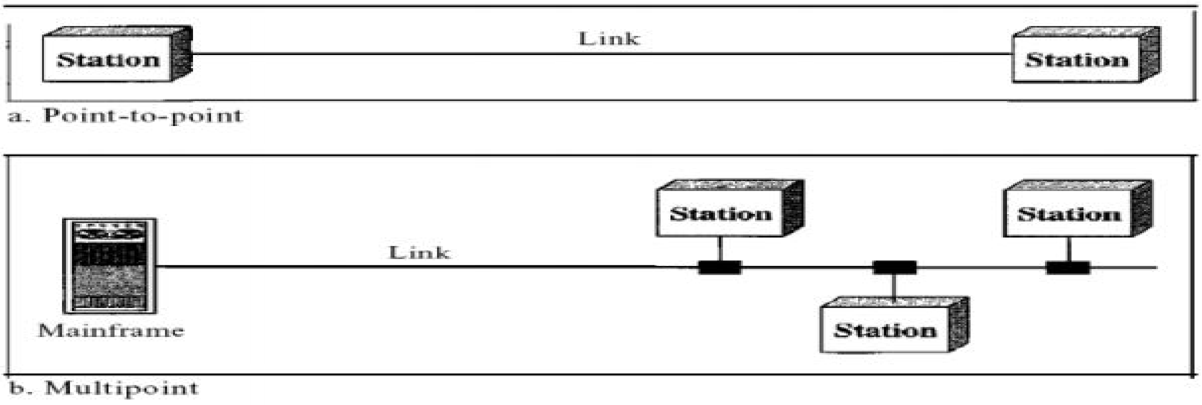
A network is two or more devices connected through links. A link is a communications pathway that transfers data from one device to another.

Point-to-Point connection:

A point-to-point connection provides a dedicated link between two devices. The entire capacity of the link is reserved for transmission between those two devices. Most point-to-point connections use an actual length of wire or cable to connect the two ends, but other options, such as microwave or satellite links, are also possible. When you change television channels by infrared remote control, you are establishing a point-to-point connection between the remote control and the television's control system.

Multipoint connection:

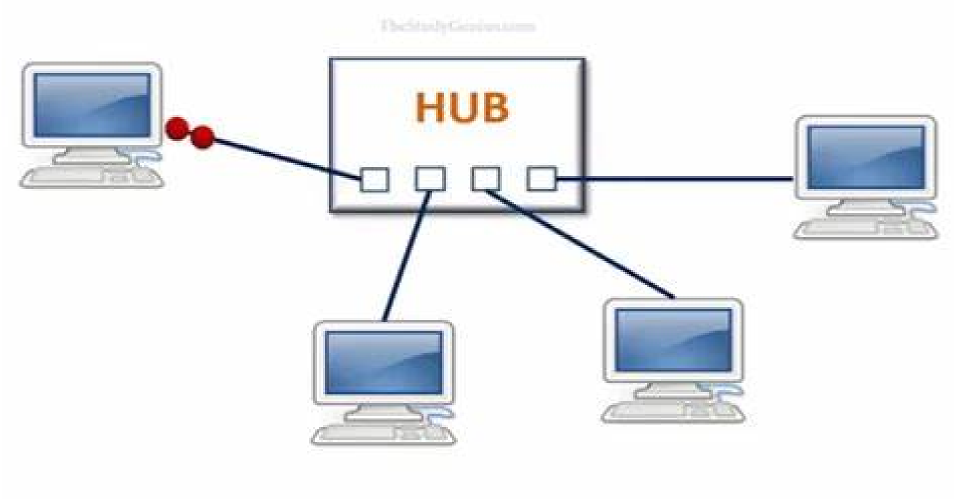
A multipoint (also called multidrop) connection is one in which more than two specific devices share a single link. In a multipoint environment, the capacity of the channel is shared, either spatially or temporally. If several devices can use the link simultaneously, it is a spatially shared connection. If users must take turns, it is a timeshared connection.



b) Introduction of Hub, switch,routers.

1. Hub:

Hub is a very simple network connecting device. In Star/hierarchical topology, a Repeater is called Hub. It is also known as a **Multiport Repeater Device .A Hub is a layer-1 device and operates only in the physical network of the OSI Model.** Since it works in the physical layer, it mainly deals with the data in the form of bits or electrical signals. A Hub is mainly used to create a network and connect devices on the same network only.A Hub is not an intelligent device, it forwards the incoming messages to other devices without checking for any errors or processing it. It does not maintain any address table for connected devices. It only knows that a device is connected to one of its ports. A Hub uses a half-duplex mode of communication.



***Following are the advantages of using a Hub*:**

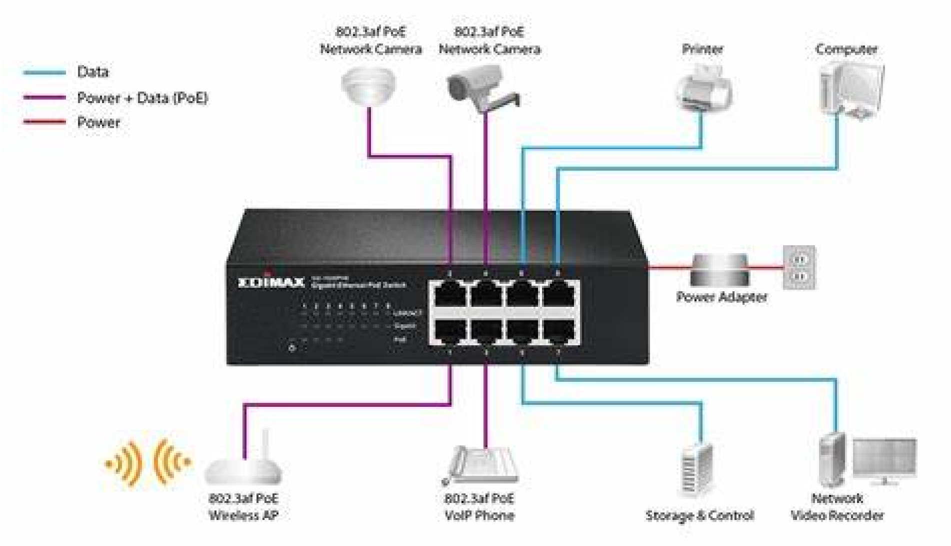
1. It is simple to implement.
2. The implementation cost is low.
3. It does not require any special system administration configuration. We can just plug and play it.

Following are the disadvantages of using a Hub:

1. It can connect devices of the same network only.
2. It uses a half-duplex mode of communication.
3. It is less secure, as it broadcasts the data packets.
4. It can be used in a limited network size only.
5. Broadcasting induces unnecessary traffic on the channel.

2.Switch:

A switch is a layer-2 network connecting device, i.e., it works on the physical and data-link layer of the OSI model. It interprets data in the form of data frames. A switch acts as a multiport bridge in the network. It provides the bridging functionality with greater efficiency.A switch maintains a Switch table which has the MAC addresses of all the devices connected to it. It uses the full-duplex mode of communication and saves bandwidth.



Following are the advantages of using a Switch:

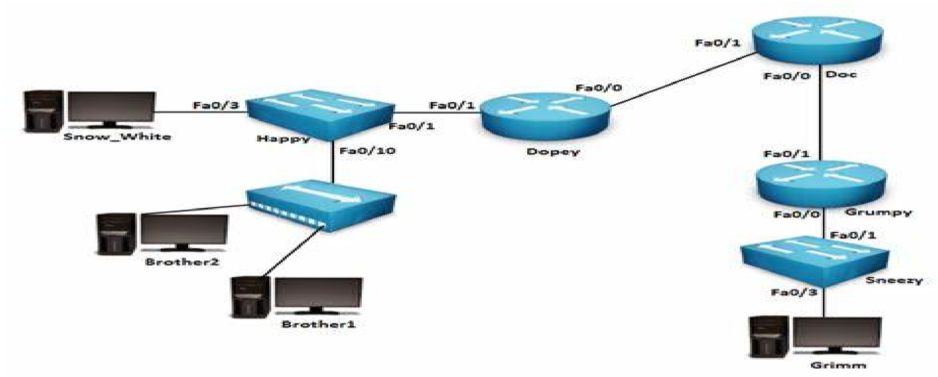
1. The implementation cost is medium.
2. It does not require any special system administration configuration. We can just plug and play it.
3. Improves security by limiting the scope of data frames.
4. It has the filtering capability.
5. It can be used in a large network.
6. It uses full-duplex mode of communication
7. It has multiple collision domains, so there are least or no collisions in the channel.

Following are the disadvantages of using a Switch:

1. It can connect devices of the same network only.
2. There is a delay in forwarding the frames due to error checking.
3. There is a need to maintain a Switch table.

3. Router

A Router is a layer-3 network connecting device, i.e., it works on the physical, data-link and network layer of the OSI model. It interprets data in the form of data packets. It is mainly an internetworking device, which can connect devices of different networks (implementing the same architecture and protocols). In other words, it can connect two physically and logically different network devices with each other. A Router is used to connect the networks or it routes traffic between the networks. **In other words, a Router is the Gateway of a network.**



Following are the advantages of using a Router:

1. It can connect devices and provides routing facilities over different networks implementing the same protocol and structure.
2. Improves security by limiting the scope of data packets.
3. It has the filtering capability.
4. It can be used in a large network.
5. It uses full-duplex mode of communication
6. It has control over both the collision and broadcast domain.

Following are the disadvantages of using a Router:

1. It is very complex to implement.
2. The implementation cost is quite high.
3. There is a need to maintain a Routing table.
4. There is a delay in forwarding the packets due to error checking.
5. It requires a special system administration configuration

Conclusion: We have studied types of connection links and about Hub, switch, routers

Experiment No. -03

Objective: To set up a basic network consisting a Hub and study of different configuration with GUI.

Resources Required:

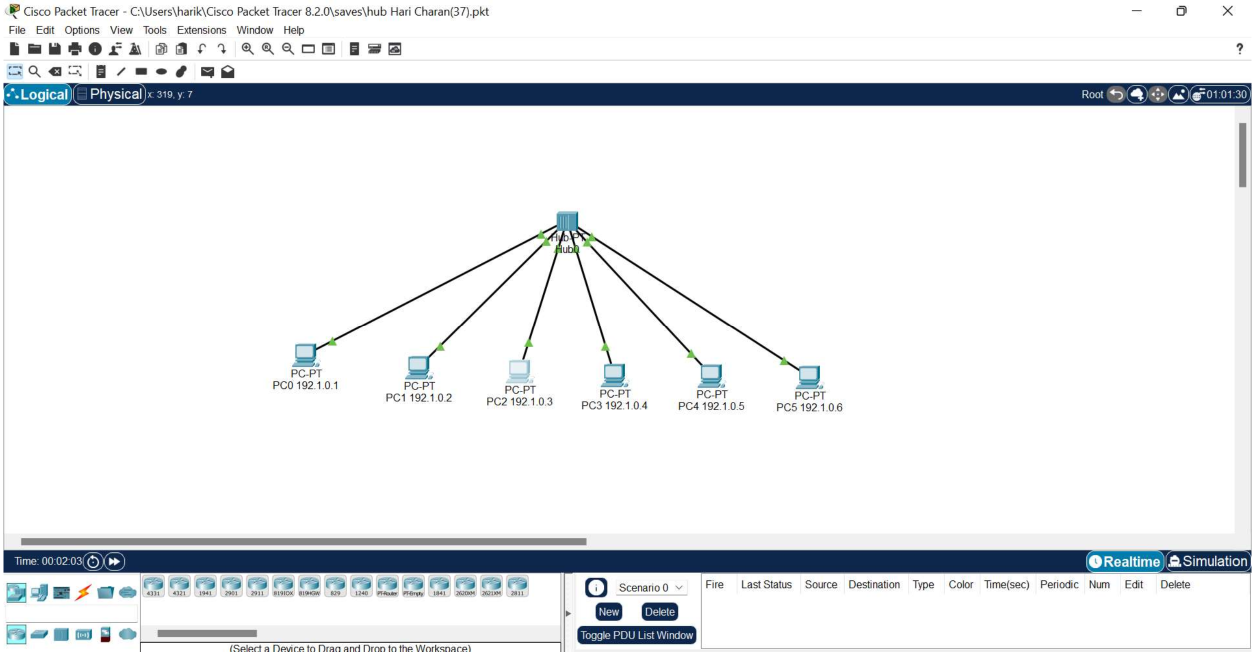
Cisco packet tracer

Theory:

HUB: A hub is a small, rectangular, inexpensive device that joins multiple network-enabled devices. They're often made of plastic and receive power from an ordinary wall outlet. The purpose of a hub is to form a single network segment on which all devices can communicate directly with each other.

PROCEDURE:

TOPOLOGY:



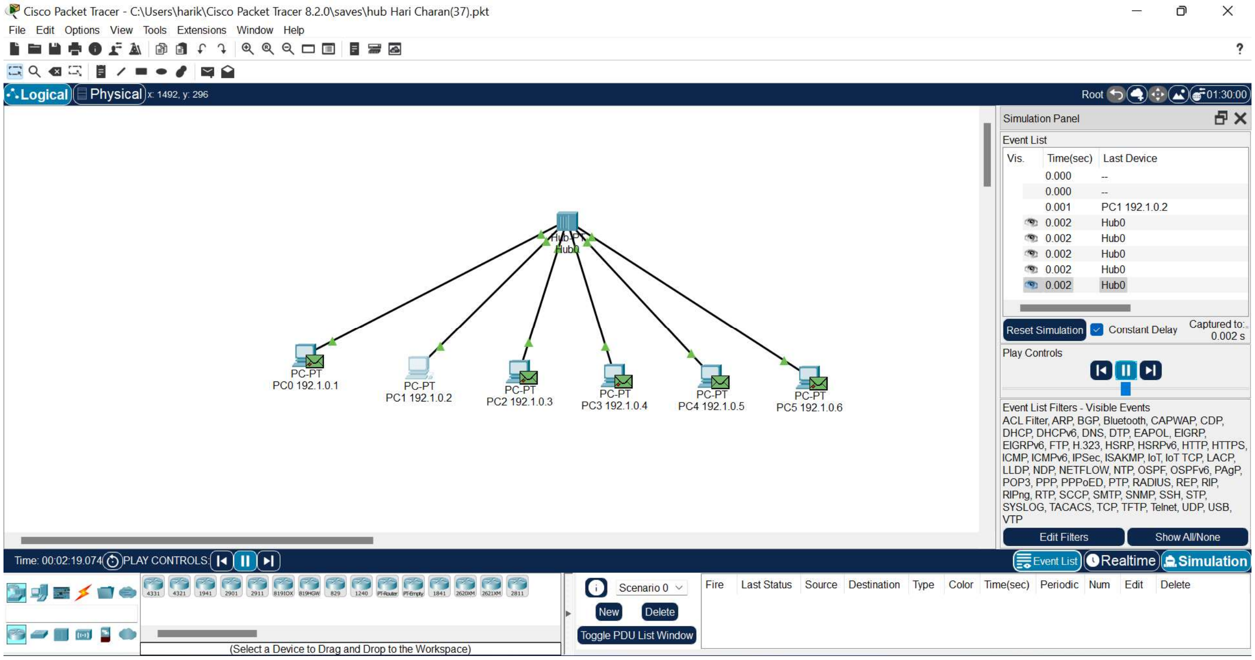
Cisco Packet Tracer - C:\Users\harik\Cisco Packet Tracer 8.2.0\saves\hub Hari Charan(37).pkt
File Edit Options View Tools Extensions Window Help
Logical
Physical
x: 319, y: 7
Root
01:01:30
Hub-PT
Hub0
PC-PT
PC0 192.1.0.1
PC-PT
PC1 192.1.0.2
PC-PT
PC2 192.1.0.3
PC-PT
PC3 192.1.0.4
PC-PT
PC4 192.1.0.5
PC-PT
PC5 192.1.0.6
Time: 00:02:03
Realtime
Simulation
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(sec) Periodic Num Edit Delete
(Select a Device to Drag and Drop to the Workspace)

Addressing table:

Device	IP address	Subnet mask
PC0	192.1.0.1	255.0.0.0
PC1	192.1.0.2	255.0.0.0
PC2	192.1.0.3	255.0.0.0
PC3	192.1.0.4	255.0.0.0
PC4	192.1.0.5	255.0.0.0
PC5	192.1.0.6	255.0.0.0

1. Setup the above shown topology using hub and end device pc
2. Connect the hub and the pc’s using copper straight connection links.
3. Power on the devices.
4. Set the IP addresses of the PC’s using above addressing table.
5. Start simulating the topology. The result is shown below.

Simulation:



Cisco Packet Tracer - C:\Users\harik\Cisco Packet Tracer 8.2.0\saves\hub Hari Charan(37).pkt

Simulation Panel

Event List

Vis.	Time(sec)	Last Device
	0.000	--
	0.000	--
	0.001	PC1 192.1.0.2
👁	0.002	Hub0
👁	0.002	Hub0
👁	0.002	Hub0
👁	0.002	Hub0
👁	0.002	Hub0

☑ Constant Delay

Captured to: 0.002 s

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP

Time: 00:02:19.074

Scenario 0

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete

Conclusion:

We have studied the basic network consisting a Hub.

Experiment No. -04

Objective: To set up a basic network consisting a switch and study of different configuration with command line interface

Resources Required:

laptop, Cisco packet tracer

Theory:

A switch operates in the layer 2, i.e. data link layer of the OSI model. It is an intelligent network device that can be conceived as a multiport network bridge. It uses MAC addresses (addresses of medium access control sublayer) to send data packets to selected destination ports. It uses packet switching technique to receive and forward data packets from the source to the destination device. It is supports unicast (one-to-one), multicast (one-to-many) and broadcast (one-to-all) communications. Transmission mode is full duplex, i.e. communication in the channel occurs in both the directions at the same time. Due to this, collisions do not occur.

Following are the advantages of using a Switch:

1.The implementation cost is medium.

2.It does not require any special system administration configuration. We can just plug and play it.

3.Improves security by limiting the scope of data frames.

4.It has the filtering capability.

5.It can be used in a large network.

6.It uses full-duplex mode of communication

7.It has multiple collision domains, so there are least or no collisions in the channel.

Following are the disadvantages of using a Switch:

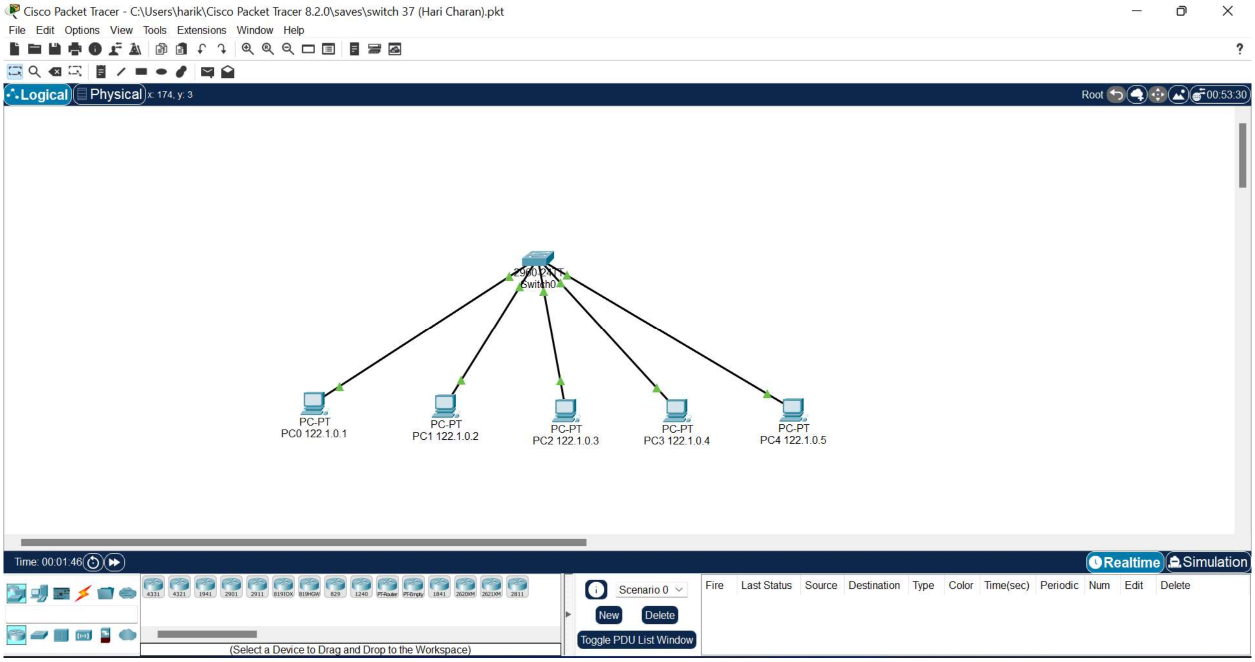
1.It can connect devices of the same network only.

2.There is a delay in forwarding the frames due to error checking.

3.There is a need to maintain a Switch table.

PROCEDURE:

Topology:

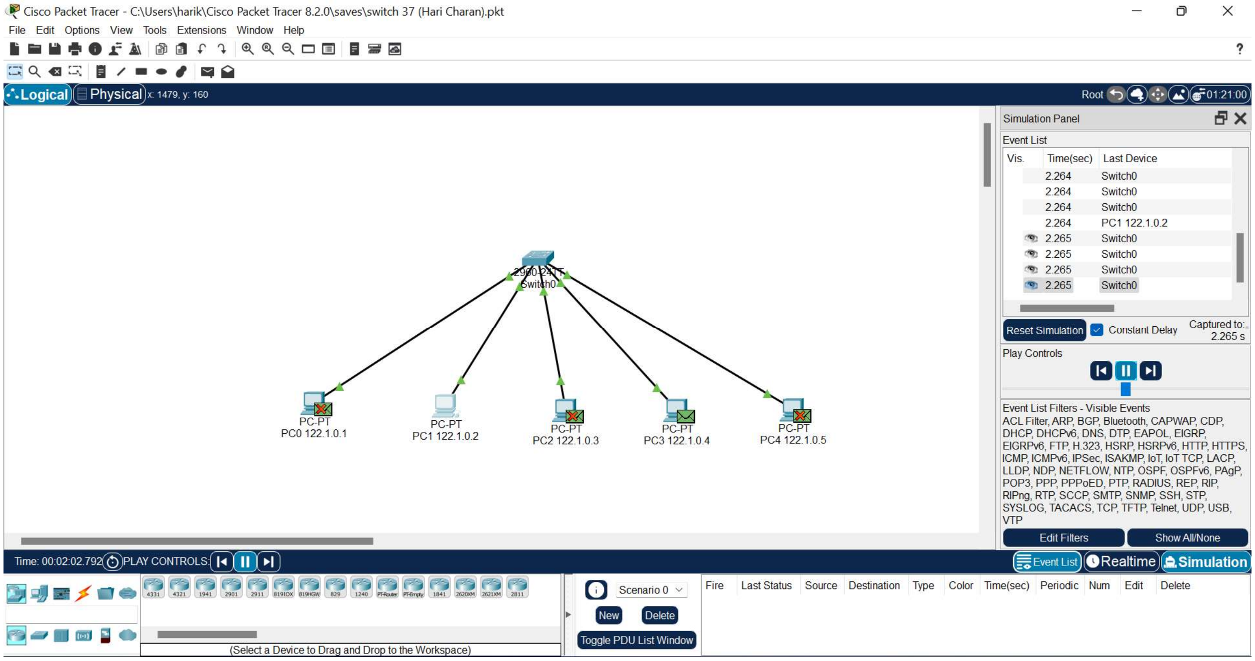


Addressing table:

Device	IP address	Subnet mask
PC0	122.1.0.1	255.0.0.0
PC1	122.1.0.2	255.0.0.0
PC2	122.1.0.3	255.0.0.0
PC3	122.1.0.4	255.0.0.0
PC4	122.1.0.5	255.0.0.0

1. Setup the above shown topology using switch and 5 end device pc's
2. Connect the switch and the pc's using copper straight connection links.
3. Power on the devices.
4. Set the IP addresses of the PC's using above addressing table.
5. Start simulating the topology. The result is shown below.

Simulation:

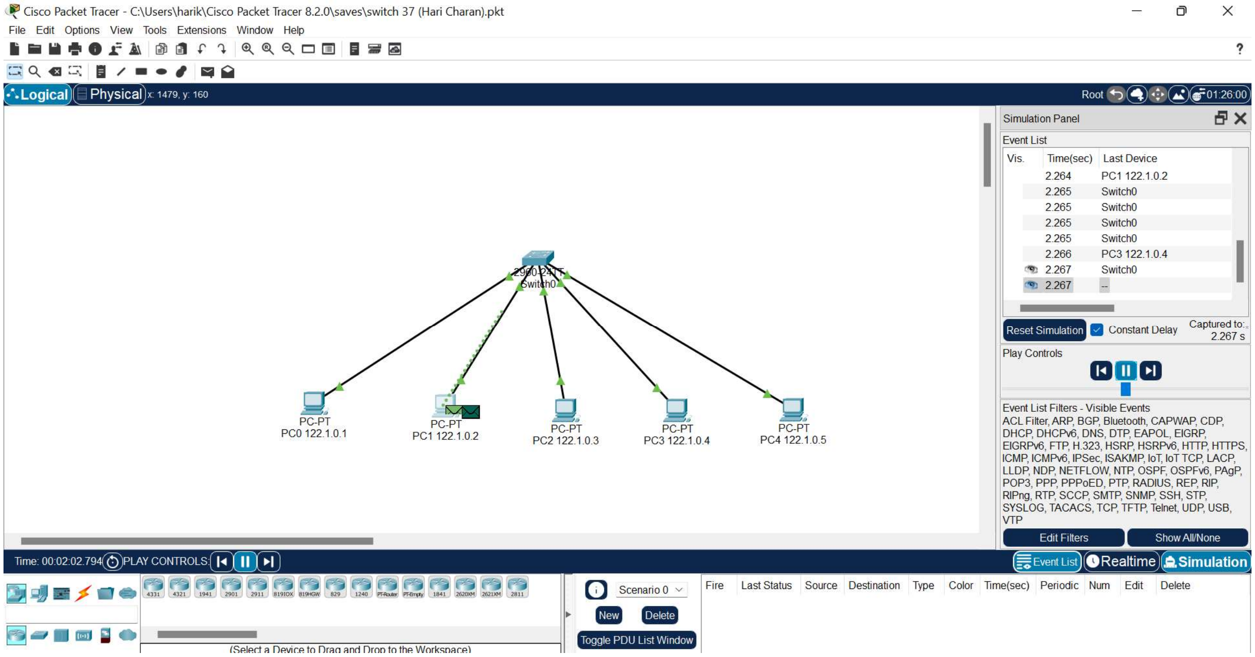


Vis.	Time(sec)	Last Device
	2.264	Switch0
	2.264	Switch0
	2.264	Switch0
	2.264	PC1 122.1.0.2
👁	2.265	Switch0
👁	2.265	Switch0
👁	2.265	Switch0
👁	2.265	Switch0

☑ Constant Delay

Captured to: 2.265 s

Event List Filters - Visible Events
ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP



Vis.	Time(sec)	Last Device
	2.264	PC1 122.1.0.2
	2.265	Switch0
	2.265	Switch0
	2.265	Switch0
	2.265	Switch0
	2.266	PC3 122.1.0.4
👁	2.267	Switch0
👁	2.267	--

☑ Constant Delay

Captured to: 2.267 s

Event List Filters - Visible Events
ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP

Conclusion:

We have studied the basic network consisting a switch

Experiment No. -05

Objective: Set up a basic network consisting two subnetwork connected by a router and study of router configuration

Resources Required:
laptop, cisco packet tracer

Theory:

Router:

A router is a layer 3 or network layer device. It connects different networks together and sends data packets from one network to another. A router can be used both in LANs (Local Area Networks) and WANs (Wide Area Networks). It transfers data in the form of IP packets. In order to transmit data, it uses IP address mentioned in the destination field of the IP packet. Routers have a routing table in it that is refreshed periodically according to the changes in the network. In order to transmit data packets, it consults the table and uses a routing protocol. In order to prepare or refresh the routing table, routers share information among each other. Routers provide protection against broadcast storms.

Following are the advantages of using a Router:

1.It can connect devices and provides routing facilities over different networks implementing the same protocol and structure.

2.Improves security by limiting the scope of data packets.

3.It has the filtering capability.

4.It can be used in a large network.

5.It uses full-duplex mode of communication

6.It has control over both the collision and broadcast domain.

Following are the disadvantages of using a Router:

1.It is very complex to implement.

2.The implementation cost is quite high.

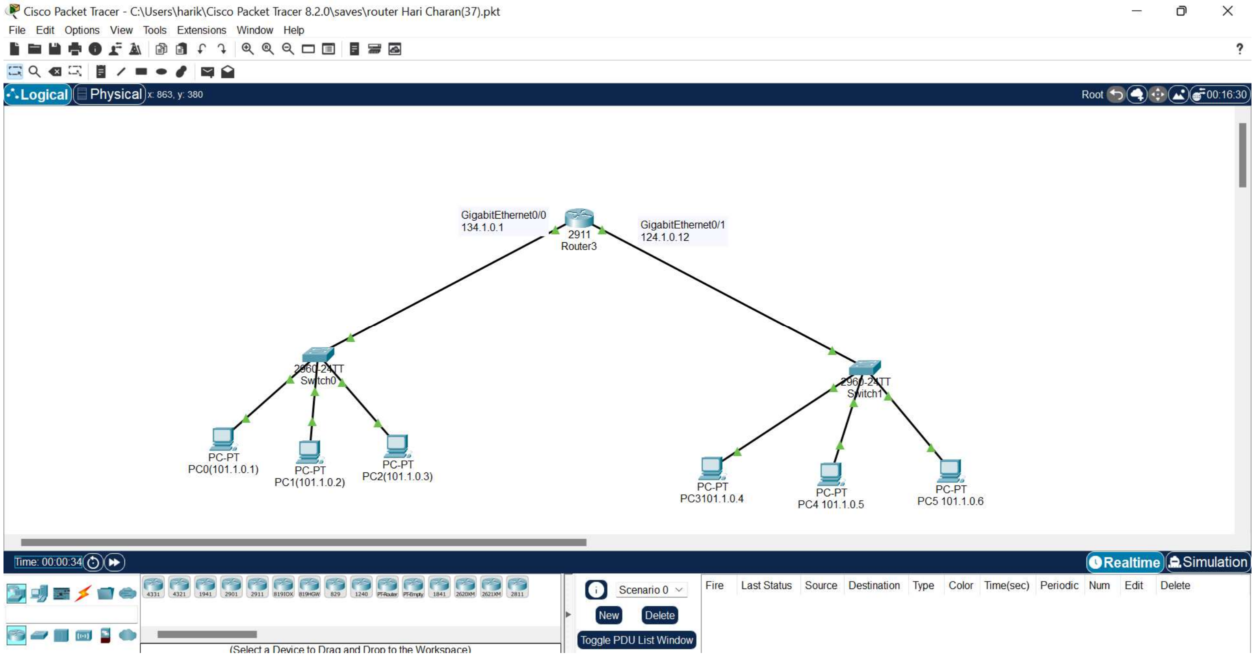
3.There is a need to maintain a Routing table.

4. There is a delay in forwarding the packets due to error checking.

5.It requires a special system administration configuration

Procedure:

Topology:

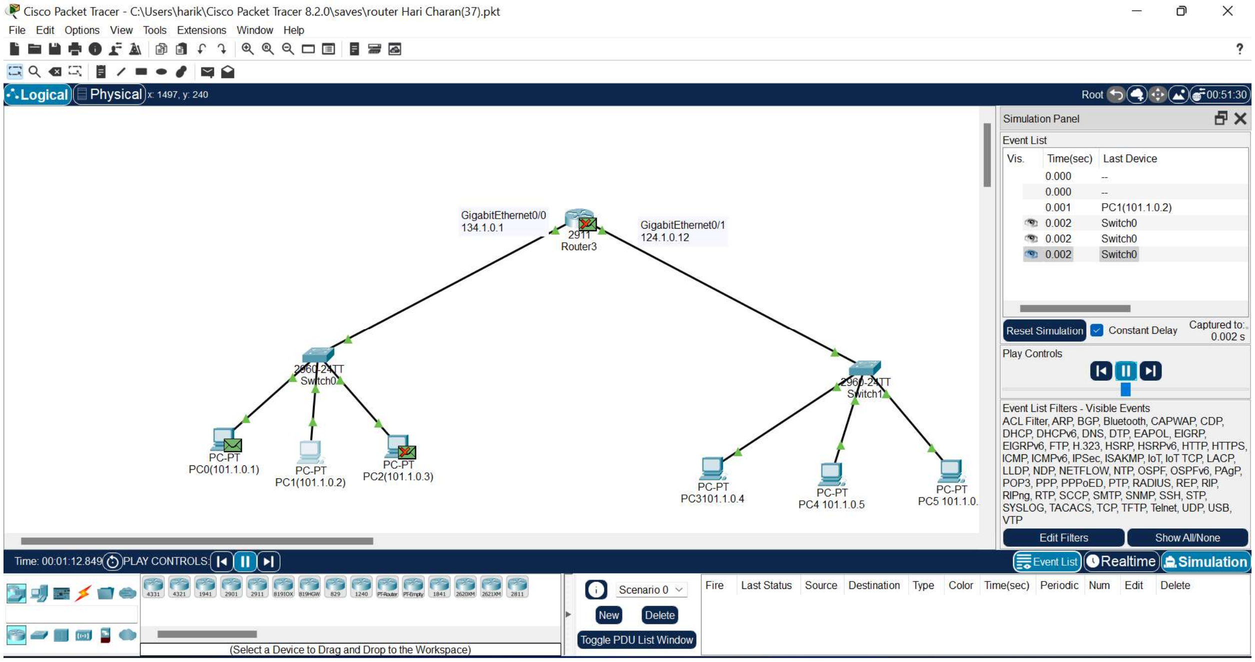


Addressing table:

Device	IP address	Subnet mask
PC0	101.1.0.1	255.0.0.0
PC1	101.1.0.2	255.0.0.0
PC2	101.1.0.3	255.0.0.0
PC3	101.1.0.4	255.0.0.0
PC4	101.1.0.5	255.0.0.0
PC5	101.1.0.6	255.0.0.0

1. Setup the above shown topology using router, switch and 6 end device pc's
2. Connect the router with two switches and the pc's using copper straight connection links.
3. Power on the devices.
4. Set the IP addresses of the PC's using above addressing table.
5. Start simulating the topology. The result is shown below.

Simulation:



Cisco Packet Tracer - C:\Users\harik\Cisco Packet Tracer 8.2.0\saves\router Hari Charan(37).pkt

Simulation Panel

Event List

Vis.	Time(sec)	Last Device
	0.000	--
	0.000	--
	0.001	PC1(101.1.0.2)
👁	0.002	Switch0
👁	0.002	Switch0
👁	0.002	Switch0

☑ Constant Delay

Captured to: 0.002 s

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP

Time: 00:01:12.849

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete

Conclusion:

We have studied the basic network consisting two sub-network connected by a router.

Experiment No. -06

Objective: Design & implement a network setup for our University

Resources Required:

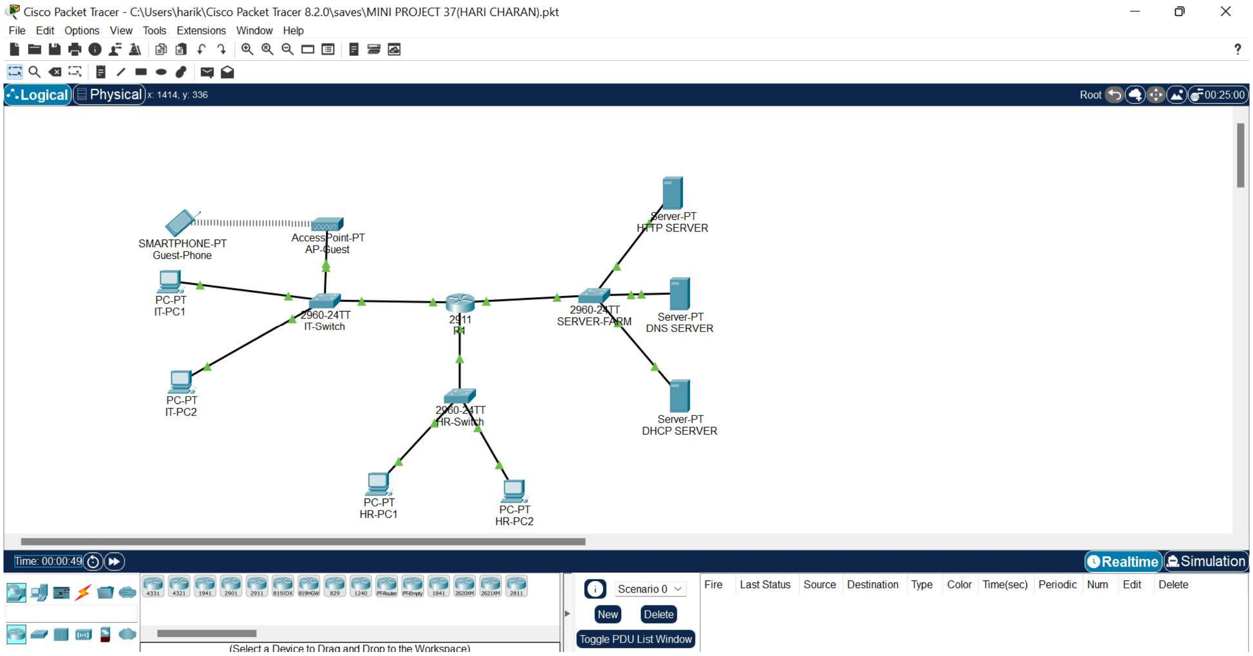
laptop, cisco packet tracer

Theory:

Switch: A network switch or switching hub is a computer networking device that connects network segments.The term commonly refers to a network bridge that processes and routes data at the data link layer (layer 2) of the OSI model. Switches that additionally process data at the network layer (layer 3 and above) are often referred to as Layer 3 switches or multilayer switches.

Router: A router is an electronic device that interconnects two or more computer networks, and selectively interchanges packets of data between them. Each data packet contains address information that a router can use to determine if the source and destination are on the same network, or if the data packet must be transferred from one network to another. Where multiple routers are used in a large collection of interconnected networks, the routers exchange information

Procedure: **Topology:**



Addressing table:

Device	Interface	IPv6 address/prefix		Default gateway	comments
		IP address	Subnet mask		
R1	G0/0	2001:DB8:ACAD:1::1/64		Not applicable	Connected to IT-Switch G0/1
		192.168.1.1	255.255.255.128	Not applicable	
	G0/1	2001:DB8:ACAD:128::1/64		Not applicable	Connected to HR-Switch G0/1
		192.168.1.129	255.255.255.192	Not applicable	
	G0/2	2001:DB8:ACAD:1::1/64		Not applicable	Connected to SERVER-FARM G0/1
		192.168.1.193	255.255.255.224	Not applicable	
IT-Switch	VLAN1	192.168.1.2	255.255.255.128	192.168.1.1	SVI for IT-Switch management
HR-Switch	VLAN 1	192.168.1.130	255.255.255.192	192.168.1.129	SVI for HR-Switch management
SERVER-FARM	VLAN 1	192.168.1.194	255.255.255.224	192.168.1.193	SVI for SERVER-FARM management
HTTP SERVER	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to SERVER-FARM Fa0/1
		192.168.1.221	255.255.255.224	192.168.1.193	
DNS SERVER	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to SERVER-FARM Fa0/2
		192.168.1.222	255.255.255.224	192.168.1.193	
DHCP SERVER	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to SERVER-FARM Fa0/3
		192.168.1.220	255.255.255.224	192.168.1.193	
IT-PC1	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to IT-Switch Fa0/1
		192.168.1.3	255.255.255.128	192.168.1.1	
IT-PC2	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to IT-Switch Fa0/2
		192.168.1.4	255.255.255.128	192.168.1.1	
HR-PC1	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to HR-Switch Fa0/1
		192.168.1.131	255.255.255.192	192.168.1.129	
HR-PC2	NIC	2001:DB8:ACAD:1::1/64		FE80::1	Connected to HR-Switch Fa0/2
		192.168.1.132	255.255.255.192	192.168.1.129	
Guest_phone	Wireless NIC	SLAAC			Wirelessly connected to AP_Guest
		DHCP			

Set the topology as shown in above figure.

Step 1: Configure the router host name.

Set the host name on the router to **R1** by using these commands.

Router>**enable**

Router#**configure terminal**

Router(config)#**hostname R1**

Step 2: Configure the privileged mode and secret passwords.

a. In global configuration mode, set the password to **cisco**.

R1(config)#**enable password cisco**

Set an encrypted privileged password to **cisco123** using the **secret** command.

```
R1(config)#enable secret cisco123
```

Step 3: Configure the console password.

a. In global configuration mode, switch to line configuration mode to specify the console line.

```
R1(config)#line console 0
```

Set the password to **cisco123**, require that the password be entered at login, and then exit line configuration mode.

```
R1(config-line)#password cisco123
R1(config-line)#login
R1(config-line)#exit
R1(config)#
```

Step 4: Configure the vty password to allow Telnet access to the router.

a. In global configuration mode, switch to line configuration mode to specify the vty
lines.

```
R1(config)#line vty 0 15
```

Set the password to **cisco123**, require that the password be entered at login, exit line configuration mode, and then
exit the configuration session.

```
R1(config-line)#password cisco123
R1(config-line)#login
R1(config-line)#exit
R1(config)#
```

Step 5: Configure password encryption, a MOTD banner, and turn off domain server lookup.

a. Currently, the line passwords and the enable password are shown in clear text when you show the
running configuration. Verify this now by entering the **show running-config** command.

To avoid the security risk of someone looking over your shoulder and reading the passwords, encrypt all clear text passwords.

```
R1(config)#service password-encryption
```

Use the **show running-config** command again to verify that the passwords are encrypted.

To provide a warning when someone attempts to log in to the router, configure a MOTD banner

```
R1(config)#banner motd Authorized Access Only
R1>emable
Translating "emable"...domain server (255.255.255.255)
```

To prevent this from happening, use the following command to stop all DNS lookups from the router
CLI.
Save the running configuration to the startup configuration.
R1(config)#**end**
R1#**copy run start**

FOR A IT-SWITCH:

Step 1: Establish a console connection to a switch.

For this activity, direct access to the IT-Switch Config and CLI tabs is disabled. You must establish a console session through IT-PC1.

Step 2: Configure the host name and VLAN 1.

a. Configure the switch host name as IT-Switch.

b. Configure port Fa0/1. Set the mode on Fast Ethernet 0/1 to access mode.

i. IT-Switch (config)#**interface fastethernet 0/1**

ii. IT-Switch (config-if)#**switchport mode access**

c. Configure IP connectivity on S1 using VLAN 1.

i. IT-Switch (config)#**interface vlan 1**

ii. IT-Switch (config-if)#**ip address 172.17.99.11 255.255.255.0**

iii. IT-Switch (config-if)#**no shutdown**

FOR SERVER-FARM:

Step 1: Establish a console connection to a switch.

For this activity, direct access to the SERVER-FARM Config and CLI tabs is disabled. You must establish a console session through IT-PC1.

Step 2: Configure the host name and VLAN 1.

a. Configure the switch host name as IT-Switch.

b. Configure port Fa0/1. Set the mode on Fast Ethernet 0/1 to access mode.

i. SERVER-FARM (config)#**interface fastethernet 0/1**

ii. SERVER-FARM (config-if)#**switchport mode access**

c. Configure IP connectivity on S1 using VLAN 1.

i. SERVER-FARM (config)#**interface vlan 1**

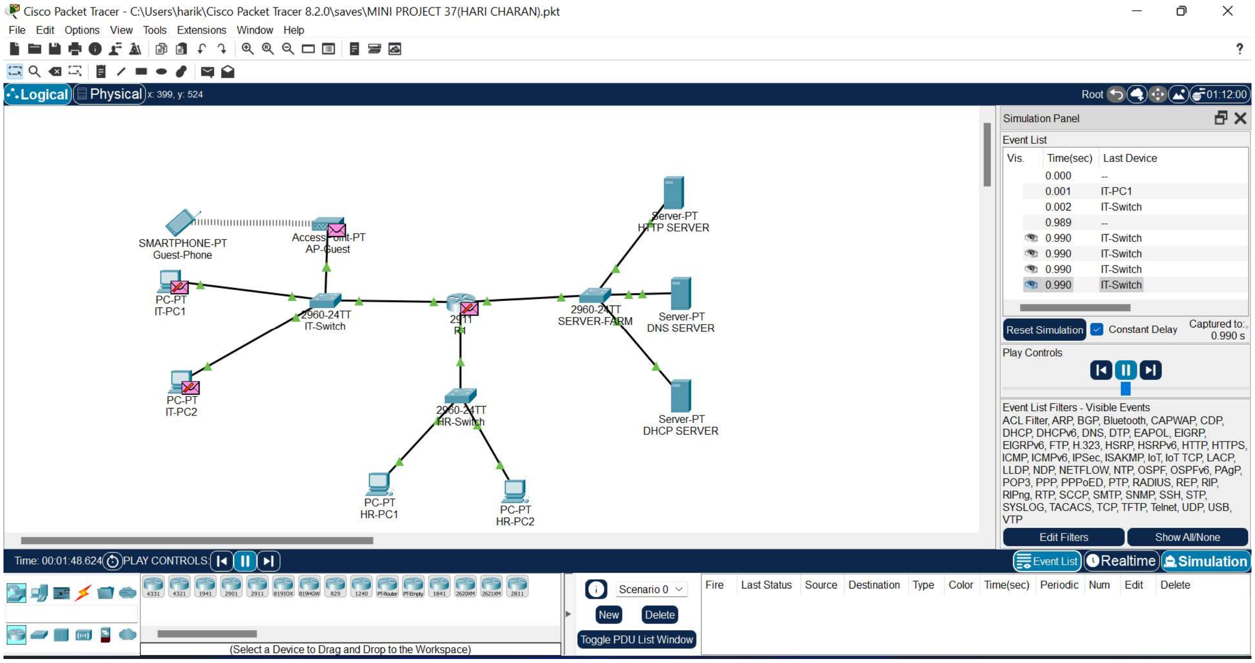
ii. SERVER-FARM (config-if)#**ip address 172.17.99.11 255.255.255.0**

iii. SERVER-FARM (config-if)#**no shutdown**

Setup all the ip addresses of end devices as shown in addressing table.

Simulate the topology the results are shown below.

SIMULATION:


Cisco Packet Tracer - C:\Users\harik\Cisco Packet Tracer 8.2.0\saves\MINI PROJECT 37(HARI CHARAN).pkt

Simulation Panel

Event List

Vis.	Time(sec)	Last Device
	0.000	--
	0.001	IT-PC1
	0.002	IT-Switch
	0.989	--
👁	0.990	IT-Switch
👁	0.990	IT-Switch
👁	0.990	IT-Switch
👁	0.990	IT-Switch

☑ Constant Delay

Captured to: 0.990 s

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP

Time: 00:01:48.624

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete

Conclusion:

We have designed & simulated a network setup for our University.

Experiment No. -07

Objective: To study different types of network topologies.

Apparatus:

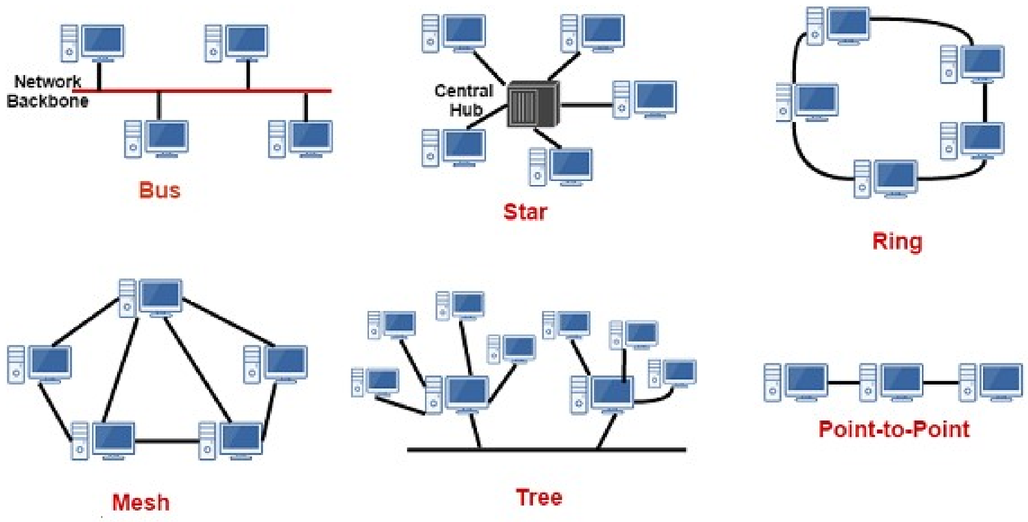
2 computers
Ethernet cables
Switch

Theory:

Star Topology:
In the star topology, all the computers connect with the help of a hub. This cable is called a central node, and all other nodes are connected using this central node. It is most popular on LAN networks as they are inexpensive and easy to install. Here is a diagram of a star topology:

Mesh Topology:
In a mesh topology, every device is connected to every other device. This topology is used in situations where high reliability is required. Here is a diagram of a mesh topology:

Tree Topology:
A tree topology combines the Star and Bus topology features. It has a host computer like the star topology, and a single cable connects all the devices like a bus topology. This topology divides the network into multiple levels. It is also known as a hierarchy topology. Here is a diagram of a tree topology:



Procedure:

- Connect two computers to the switch using Ethernet cables.
- Turn on the computers and the switch.
- Open the command prompt on both computers.
- Type ipconfig in the command prompt and press enter.

- Note down the IP address of both computers.
- Open the network and sharing center on both computers.
- Click on the "Change adapter settings" option.
- Right-click on the Ethernet connection and select "Properties".
- Select "Internet Protocol Version 4 (TCP/IPv4)" and click on the "Properties" button.
- Select the "Use the following IP address" option.
- Enter the IP address of the first computer in the "IP address" field and enter the IP address of the second computer in the "Default gateway" field.
- Click on the "OK" button.
- Repeat steps 8-12 for the second computer, but swap the IP addresses.
- Open the command prompt on both computers.
- Type ping <IP address of the other computer> and press enter.
- Verify that both computers can communicate with each other.

Conclusion: This experiment helps students understand the concept of network topologies and how they can be implemented in a LAN environment. It also helps students understand the importance of IP addresses and how they are used to identify devices on a network.
Here are some diagrams of different network topologies:

Experiment No. -08

Experiment Title: Comparative Analysis of Flow Control Protocols in Data Communication

Objective:

The primary objective of this experiment is to study and compare different flow control protocols in the context of data communication and computer networks. Flow control plays a crucial role in ensuring efficient and reliable data transfer between devices in a network. By conducting this experiment, we aim to gain insights into the performance characteristics, advantages, and limitations of various flow control protocols.

Equipment and Materials:

Computers with network interfaces
Network simulation software (e.g., Cisco Packet Tracer, GNS3)
Ethernet cables
Switches and routers (if not using simulation software)
Data communication devices (such as PCs or laptops)
Protocol analyzer tool (Wireshark)
Experimental Setup:

Establish a simple network topology using either physical devices or network simulation software. Ensure that the network consists of at least two nodes capable of data communication.

Select and implement different flow control protocols for the experiment. Common flow control protocols include:

What is flow control?

Flow control is a technique used to regulate data transfer between computers or other nodes in a network. Flow control ensures that the transmitting device does not send more data to the receiving device than it can handle. If a device receives more data than it can process or store in memory at any given time, the data is lost and needs to be retransmitted.

The purpose of flow control is to throttle the amount of data transmitted to avoid overwhelming the receiver's resources. This is accomplished through a series of messages that the receiver transmits to the sender to acknowledge if frames have been received. The sender uses these messages to determine when to transmit more data. If the sender does not receive an acknowledgement (ACK), it concludes that there has been a problem with the transmission and retransmits the data.

Flow control is implemented in different ways, depending on how the sender and receiver handle messages and track data frames. There are two basic approaches to flow control: stop and wait and sliding window. The stop-and-wait approach is the simplest to implement, but it is not as efficient as sliding window, which delivers better network performance and utilizes network resources more effectively.

Stop-and-Wait flow control

In the stop-and-wait approach, the sender segments the data into frames and then transmits one frame at a time to the receiver, which responds to each frame with an ACK message. This process occurs through the following steps:

1. The sender transmits a data frame to the receiver.

2. The sender waits for the receiver to respond.
3. Upon receiving the frame, the receiver transmits an ACK to the sender.
4. Upon receiving the ACK, the sender sends the next frame to the receiver and waits for the next ACK. If the sender does not receive an ACK within a defined time limit, known as a *timeout*, the sender retransmits the same frame.
5. The process continues until the sender has finished transmitting all the data to the receiver.

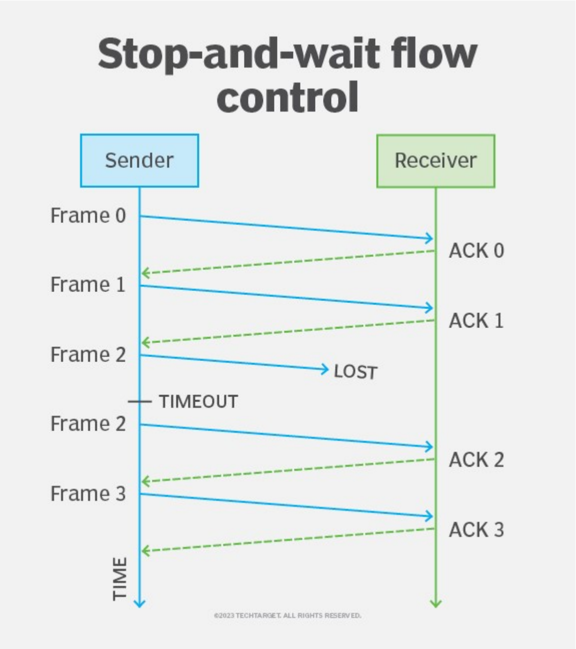


Figure 1. How stop-and-wait flow control works

Stop-and-Wait Protocol:
Figure 1 illustrates how this exchange works. In this case, the sender starts by transmitting Frame 0 and then waiting for the ACK. When Frame 0 reaches its destination, the receiver sends ACK 0 to the sender.

After receiving ACK 0, the sender transmits Frame 1 and waits for ACK 1. When that arrives, the sender transmits Frame 2 and waits again. This time, however, the sender does not receive ACK 2 before the timeout occurs, so it retransmits Frame 2. The frame now arrives at its destination, so the receiver sends ACK 2. When the sender receives ACK 2, it transmits Frame 3, which is also acknowledged by the receiver.

Stop and wait belongs to a category of error control mechanisms called *automatic repeat requests* (ARQs), which rely on the use of ACKs to determine if a data transmission was successful or if retransmission is needed. Other ARQs include Go-Back-N ARQ and Selective Repeat ARQ, both of which use the sliding window protocol.

Stop and wait is simpler to implement than sliding window. It is also fairly reliable because the sender receives an ACK for each frame successfully transmitted to the receiver. These qualities, however, also make data communications much slower, which can be exacerbated by long distances and heavy traffic. The stop-and-wait approach also tends to underutilize network resources.

Sliding Window (Selective Repeat and Go-Back-N)

The sliding window approach addresses many of the issues that come with stop and wait because the sender can transmit multiple frames at once without having to wait for an ACK for each frame. However, this approach also comes with additional complexity.

When first connecting, the sender and receiver establish a window that determines the maximum number of frames the sender can transmit at a time. During the transmission, the sender and receiver must carefully track which frames have been sent and received to ensure that all the data reaches its destination and is reassembled in the correct order.

Sliding window flow control can be implemented using one of two approaches: Go-Back-N and Selective Repeat. With the Go-Back-N approach, the sender can send one or more frames but never more frames than the window allows. As the receiver acknowledges the frames, the sender moves to the next batch, or window, of frames that can now be sent. If there is a problem with a transmitted frame, the sender retransmits all the frames in the current window.

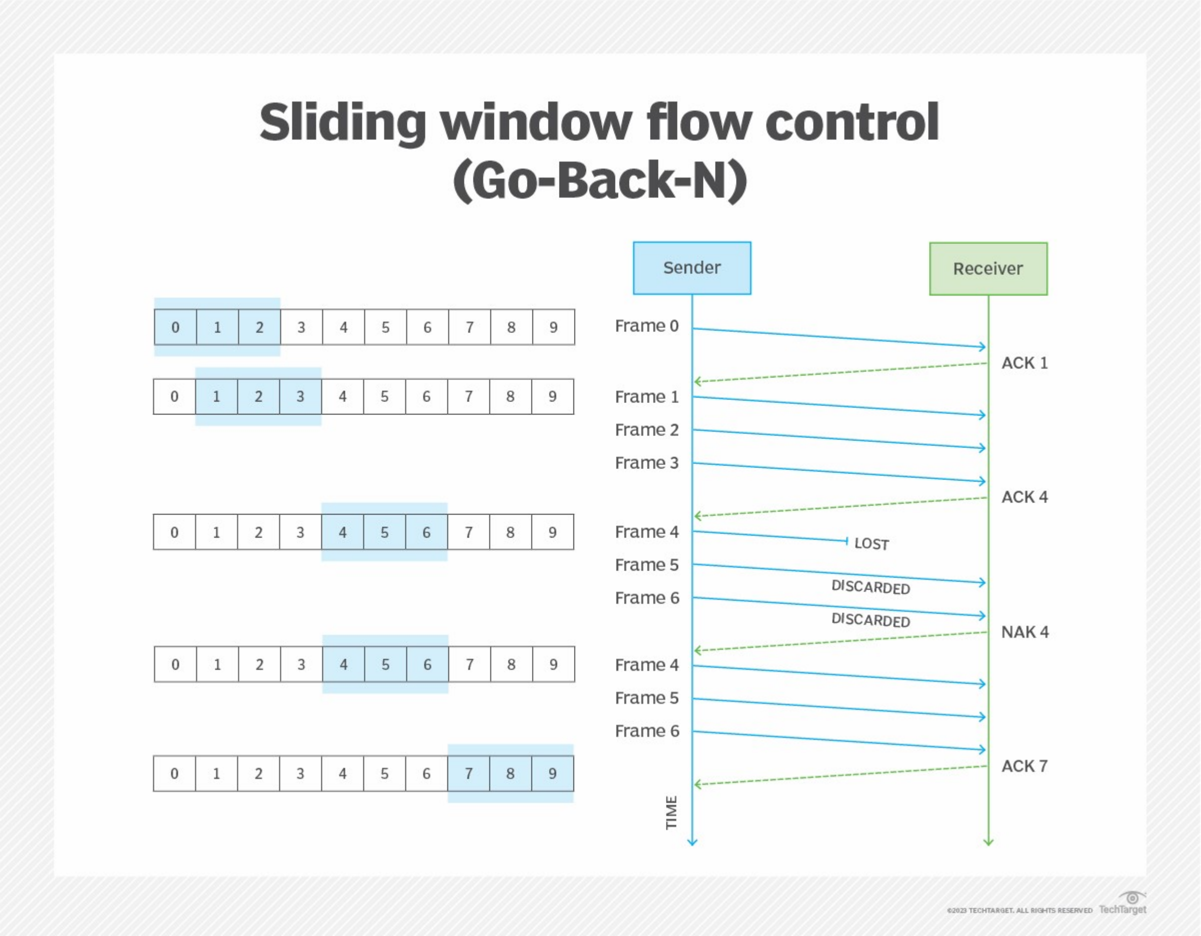


Figure 2. How Go-Back-N type of sliding window flow control works

Figure 2 shows an example of how Go-Back-N works. In this case, the window consists of only three frames -- initially, Frames 0 through 2. The sender begins by transmitting Frame 0 to the receiver. Upon receiving Frame 0, the receiver sends an ACK that specifies the next frame to send (Frame 1), rather than specifying the frame that has just been received.

When the sender receives ACK 1, it moves the window over by one position, dropping Frame 0 and adding Frame 3. The sender then transmits Frames 1, 2 and 3, which represent the window's entire contents. The sender does not necessarily need to send Frame 0 first, followed by Frames 1 through 3. This is illustrated in Figure 2 to demonstrate how the process works.

Upon receiving the three frames, the receiver sends a cumulative ACK that specifies the next frame to send, which is Frame 4. The ACK indicates that the receiver now has all the preceding frames (0 through 3).

When the sender receives ACK 4, it adjusts the window so that it now includes Frames 4 through 6 and then transmits those frames. This time, however, Frame 4 gets lost in the transmission, while Frames 5 and 6 reach their destination. Upon receiving Frame 5, the receiver detects that Frame 4 is missing and sends a negative acknowledgement (NAK) that specifies Frame 4. At the same time, the receiver discards Frames 5 and 6.

When the sender receives the NAK, it retransmits Frames 4 through 6 and waits for the ACK. The frames arrive with no errors the second time around, so the receiver returns an ACK indicating that the sender can now transmit Frame 7. The sender adjusts the window accordingly and transmits the next set of frames, starting with Frame 7.

The Selective Repeat approach is similar to Go-Back-N. The primary difference is that Selective Repeat does not retransmit the entire window if there is an error, only the individual frame in dispute. Selective Repeat does not support cumulative ACK messages like Go-Back-N, so each ACK is specific to the frame that was just received, which is what enables the sender to identify the precise frame that needs to be retransmitted.

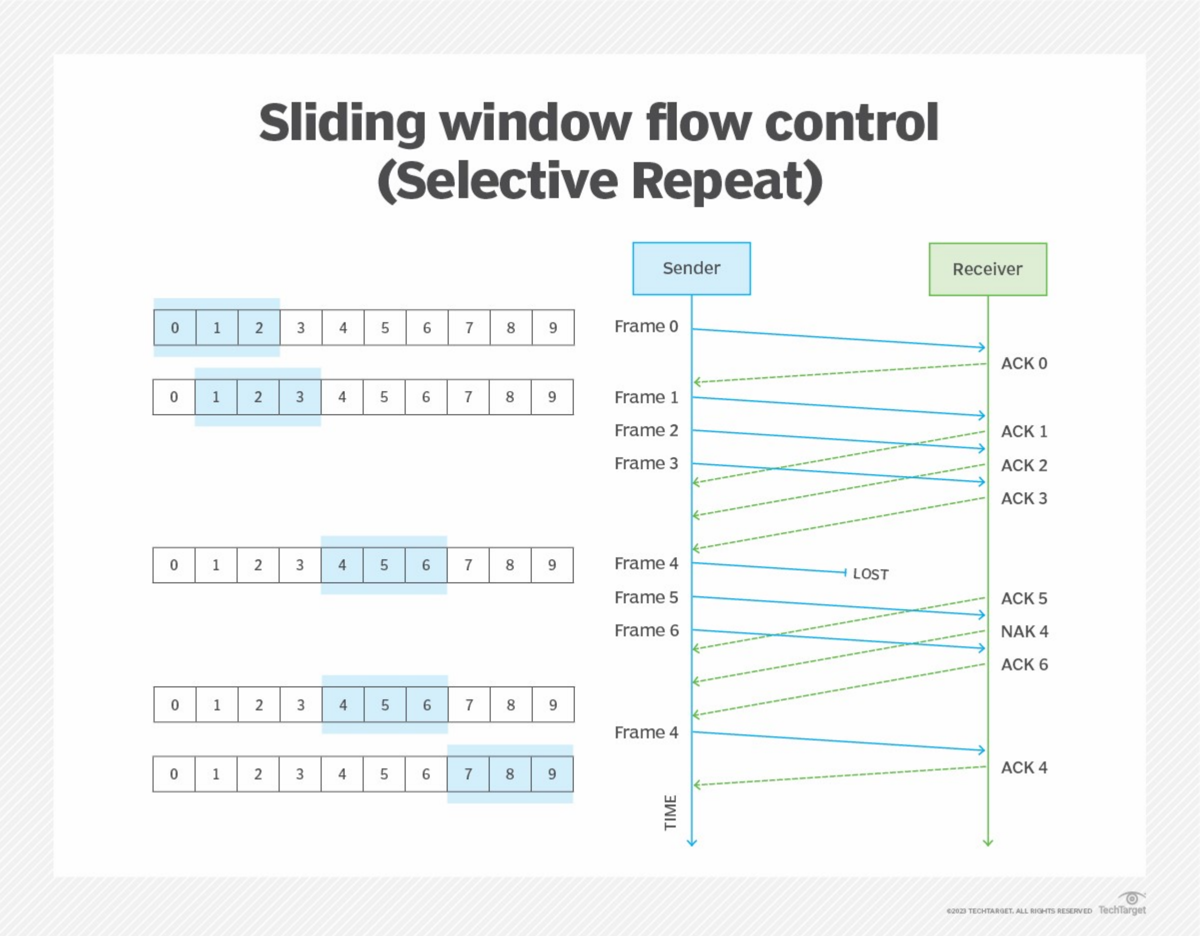


Figure 3. How Selective Repeat sliding window flow control works

Figure 3 illustrates an example of the Selective Repeat process. After transmitting Frame 0, the sender receives an ACK, so it transmits Frames 1 through 3 and receives an ACK for each one. The sender then transmits Frames 4 through 6. When Frames 5 and 6 arrive at the receiver, but not Frame 4, the receiver sends ACK 5 and ACK 6, along with NAK 4. The sender responds to the NAK by retransmitting Frame 4. Upon receiving Frame 4, the receiver sends an ACK. The sender then adjusts the window and transmits the next three frames, starting with Frame 7.

Both Selective Repeat and Go-Back-N are more efficient than the stop-and-wait approach, but there are important differences between the two sliding window approaches. The Go-Back-N approach can consume more bandwidth because all the frames in a window are retransmitted if an error occurs. However, it is not as complex to implement as Selective Repeat and does not require the same amount of system resources. Selective Repeat comes with greater management overhead because the frames must be tracked and sorted throughout the data transmission.

Summary of Sliding Window Protocol (Selective Repeat and Go-Back-N):

The Sliding Window protocol is an advanced flow control mechanism that allows for the concurrent transmission of multiple frames, enhancing data transfer efficiency. Two variations of this protocol are Selective Repeat and Go-Back-N.

Selective Repeat:

1. **Sender (S) Side:**
    - S divides the data stream into a sequence of frames and sends a window of frames to R without waiting for individual acknowledgments.

- S maintains a record of which frames were sent in the current window.
- Upon receiving acknowledgments, S updates its record, and any unacknowledged frames are resent.
- The receiver (R) individually acknowledges correctly received frames.

2. **Receiver (R) Side:**
    - R receives a window of frames and individually acknowledges each correctly received frame.
    - If a frame is damaged or lost, R requests retransmission only for that specific frame.
    - Out-of-sequence frames are buffered until the missing frame is received.

Go-Back-N:

1. **Sender (S) Side:**
    - S sends a continuous stream of frames without waiting for individual acknowledgments.
    - S maintains a "window" of frames awaiting acknowledgment.
    - If an acknowledgment is not received within a specified timeout, S retransmits all frames in the current window.
2. **Receiver (R) Side:**
    - R receives frames and individually acknowledges each correctly received frame.
    - If a frame is damaged or lost, R discards all subsequent frames until the correct one is received.
    - R sends cumulative acknowledgments, indicating the highest correctly received frame.

Leaky Bucket Algorithm

The Leaky Bucket Algorithm is a traffic shaping mechanism used to control the rate at which data is sent into a network. It is often employed in scenarios where there is a need to smooth out bursty traffic and ensure a consistent flow of data. In this algorithm, the "bucket" has a finite capacity, and data is added to it at a constant rate. If the bucket overflows, excess data is discarded or marked for slower transmission. This helps in controlling the rate at which data is released into the network, preventing congestion. The diagram below illustrates the concept of the Leaky Bucket Algorithm, where incoming data is added to the bucket, and the network allows a controlled output rate.

Leaky Bucket Algorithm:

The Leaky Bucket Algorithm is a traffic shaping mechanism used to control the rate at which data is sent into a network. It is commonly employed in scenarios where there is a need to smooth out bursty traffic and ensure a consistent flow of data. The algorithm is named after its analogy with a bucket that has a leak.

Basic Operation:

Bucket Structure:

The "bucket" has a finite capacity, representing the maximum amount of data that can be transmitted in a given time period.
Incoming data is added to the bucket at a constant rate.
Leakage:

The bucket has a leakage mechanism, allowing it to release data at a controlled rate.
If the incoming data rate exceeds the leak rate, the bucket may overflow.
Data Transmission:

Data is transmitted into the network at the rate determined by the leaky bucket, preventing bursts of data that could lead to network congestion.
If the bucket is full, excess data is either discarded or marked for slower transmission.
Configure the network parameters, such as data transfer rate, latency, and error rates, to create realistic network conditions.

Design a data transfer scenario where a significant amount of data needs to be transferred between the nodes in the network.

Conduct the following experiments for each flow control protocol:
a. Measure and compare the throughput of data transfer under normal conditions.
b. Introduce network congestion or errors and observe how each flow control protocol handles these situations.
c. Analyze the impact of varying parameters like window size in sliding window protocols on the overall performance.

Use a protocol analyzer tool (e.g., Wireshark) to capture and analyze the network traffic during the experiments. This will provide insights into the efficiency of each flow control protocol.

Data Collection and Analysis:

Record the throughput, delay, and any retransmission events for each flow control protocol in different scenarios.

Analyze the captured network traces to identify the behavior of each protocol under various conditions.

Compare the performance of different flow control protocols based on the collected data and draw conclusions regarding their suitability for specific network conditions.

Conclusion:
Summarize the findings of the experiment, highlighting the strengths and weaknesses of each flow control protocol. Provide recommendations for selecting the most appropriate flow control protocol based on specific network requirements and conditions. Discuss potential areas for further research and improvement in flow control mechanisms.

Experiment No. -09

Experiment Title: Study of ALOHA Protocol in Data Communication
Objective: The main objective of this experiment is to study the Pure ALOHA protocol and analyze its performance characteristics in the context of data communication. Pure ALOHA is a simple, contention-based protocol used for media access control in shared communication channels. Through this experiment, we aim to understand the efficiency, collision characteristics, and throughput of the Pure ALOHA protocol under different network conditions.
Equipment and Materials:

1. Computers or devices with network interfaces
2. Network simulation software (e.g., Cisco Packet Tracer, GNS3)
3. Ethernet cables
4. Switches and routers (if not using simulation software)
5. Data communication devices (such as PCs or laptops)
6. Stopwatch or timer
7. Protocol analyzer tool (e.g., Wireshark)

Theory:

The ALOHA protocol was first developed at the University of Hawaii in the early 1970s for packet radio networks. However, it can be used in any situation where multiple devices share a common communication channel. This protocol allows devices to transmit data at any time, without a set schedule. This is known as a random access technique, and it is asynchronous because there is no coordination between devices. When multiple devices attempt to transmit data at the same time, it can result in a collision, where the data becomes garbled. In this case, each device will simply wait a random amount of time before attempting to transmit again. The basic concept of the ALOHA protocol can be applied to any system where uncoordinated users are competing for the use of a shared channel.

What is Pure ALOHA:

- Pure ALOHA refers to the original ALOHA protocol. The idea is that each station sends a frame whenever one is available. Because there is only one channel to share, there is a chance that frames from different stations will collide.
- The pure ALOHA protocol utilizes acknowledgments from the receiver to ensure successful transmission. When a user sends a frame, it expects confirmation from the receiver. If no acknowledgment is received within a designated time period, the sender assumes that the frame was not received and retransmits the frame.
- When two frames attempt to occupy the channel simultaneously, a collision occurs and both frames become garbled. If the first bit of a new frame overlaps with the last bit of a frame that is almost finished, both frames will be completely destroyed and will need to be retransmitted. If all users retransmit their frames at the same time after a time-out, the frames will collide again.
- To prevent this, the pure ALOHA protocol dictates that each user waits a random amount of time, known as the back-off time, before retransmitting the frame. This randomness helps to avoid further collisions.

- The time-out period is equal to the maximum possible round-trip propagation delay, which is twice the amount of time required to send a frame between the two most widely separated stations (2 x Tp).

- Let all the packets have the same length. And each requires a one-time unit for transmission (tp). Consider any user to send packet A at a time. If any other user B has generated a packet between time (to), and (to + tp), the end of packet B will collide with the beginning of packet A. Since in a pure ALOHA packet, a station does not listen to the channel before transmitting, it has no way of knowing that the above frame was already underway.

- Similarly, if another user wants to transmit between (to, +tp) and (to +2tp) i.e. packet C, the beginning of packet C will collide with the end of packet A. Thus if two packets overlap by even the smallest amount in the vulnerable period both packets will be corrupted and need to be retransmitted.

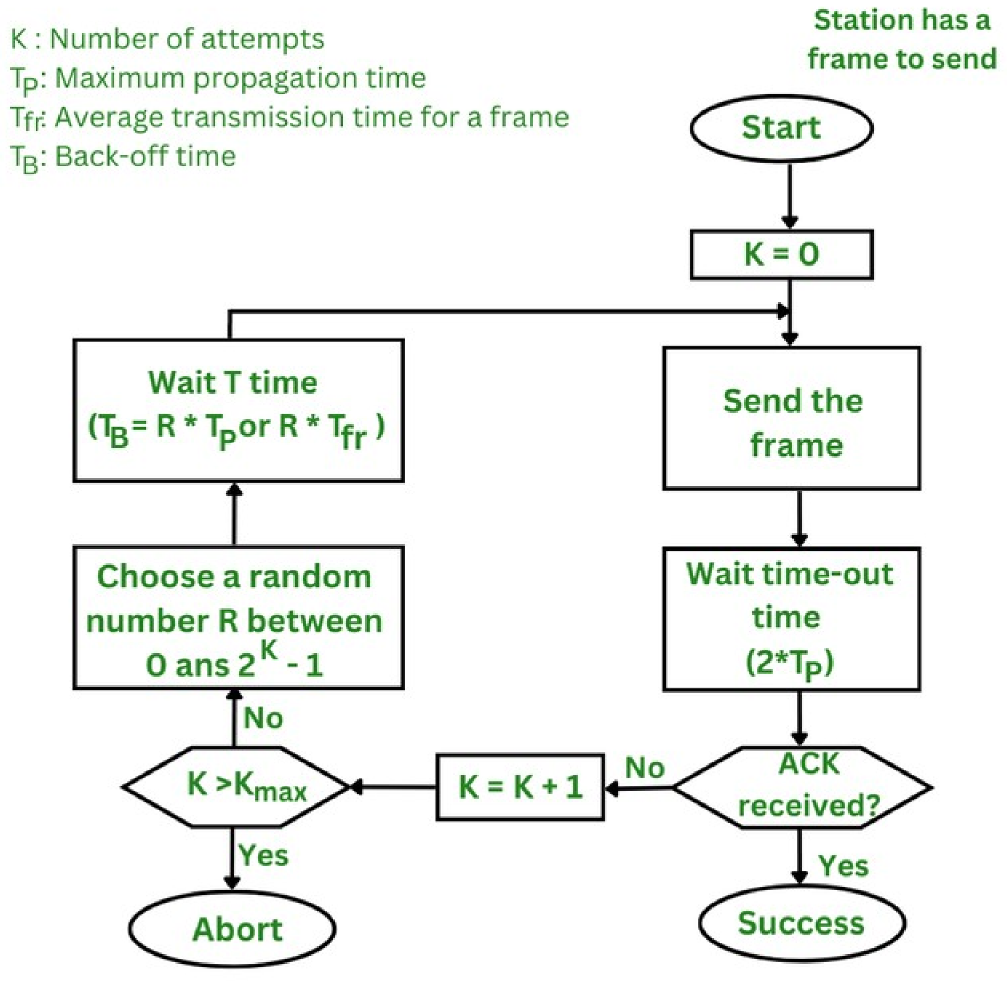

K : Number of attempts
T_P: Maximum propagation time
T_{fr}: Average transmission time for a frame
T_B: Back-off time
Station has a frame to send
Start
K = 0
Send the frame
Wait time-out time (2*T_P)
ACK received?
Yes
Success
No
K = K + 1
K >K_{max}
Yes
Abort
No
Choose a random number R between 0 ans 2^K - 1
Wait T time (T_B= R * $T_P$or R * T_{fr})

Figure: *Pure ALOHA procedure*

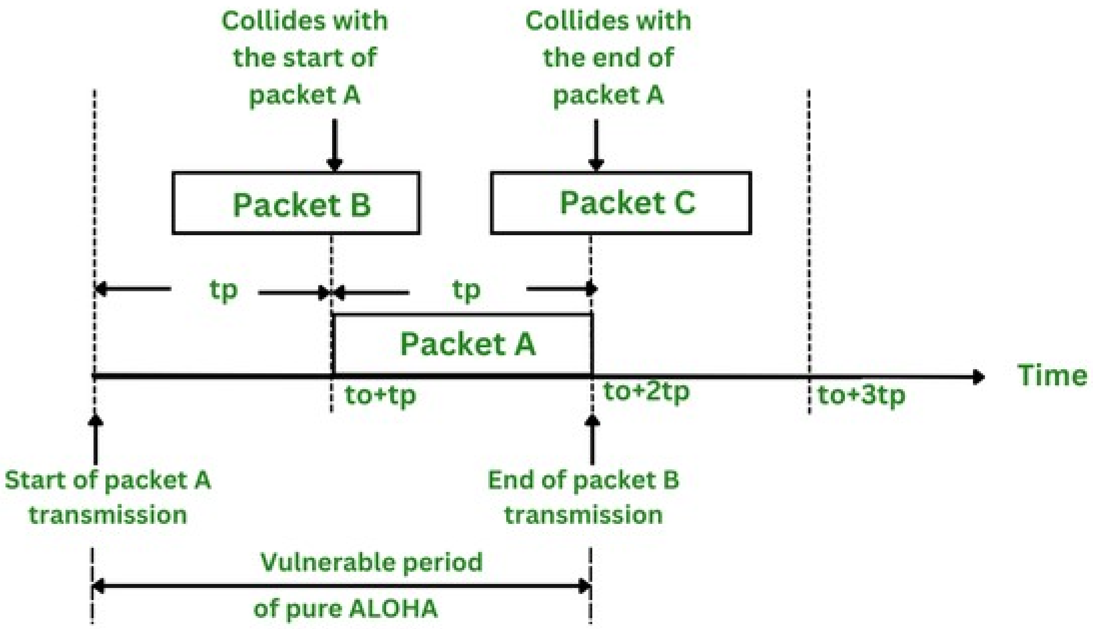

Collides with the start of packet A
Collides with the end of packet A
Packet B
Packet C
tp
tp
Packet A
Time
to+tp
to+2tp
to+3tp
Start of packet A transmission
End of packet B transmission
Vulnerable period of pure ALOHA

Figure: Vulnerable period for packet A

Key Features of Pure ALOHA:
1. Random Access: Devices can send data whenever they have something to transmit, without needing to wait for a predetermined time slot.
2. Uncoordinated Transmission: Devices do not coordinate with each other before transmitting. They simply attempt to send data whenever they have data to send.
3. Simple Implementation: Pure ALOHA is straightforward to implement, making it suitable for early network experiments and scenarios with low traffic.
4. Persistent Approach: Devices continue to attempt transmission even after a collision, using a form of exponential backoff. This means they introduce random delays before retrying, which helps reduce the chances of repeated collisions.
5. Contention-Based: Since devices transmit without coordination, collisions may occur if two or more devices transmit simultaneously. Collisions are detected through feedback from the receiver or by the transmitting device itself.

What is Slotted ALOHA:

Slotted ALOHA is an improved version of the pure ALOHA protocol that aims to make communication networks more efficient. In this version, the channel is divided into small, fixed-length time slots and users are only allowed to transmit data at the beginning of each time slot. This synchronization of transmissions reduces the chances of collisions between devices, increasing the overall efficiency of the network.

The channel time is separated into time slots in slotted ALOHA, and stations are only authorized to transmit at particular times. These time slots correspond to the packet transmission time exactly. All users are then synchronized to these time slots so that whenever a user sends a packet, it must precisely match the next available channel slot. As a result, wasted time due to collisions can be reduced to one packet time or the susceptible period can be half.

When a user wants to transmit a frame, it waits until the next time slot and then sends the frame. If the frame is received successfully, the receiver sends an acknowledgment. If the acknowledgment is not received within a time-out period, the sender assumes that the frame was not received and retransmits the frame in the next time slot.

Slotted ALOHA increases channel utilization by reducing the number of collisions. However, it also increases the delay for users, as they have to wait for the next time slot to transmit their frames. It's also worth noting that there is a variant of slotted ALOHA called "non-persistent slotted ALOHA" which is a variation of slotted ALOHA, in this variant the station that wants to send data, first listens to the channel before sending the data. If the channel is busy it waits for a certain time before trying again.

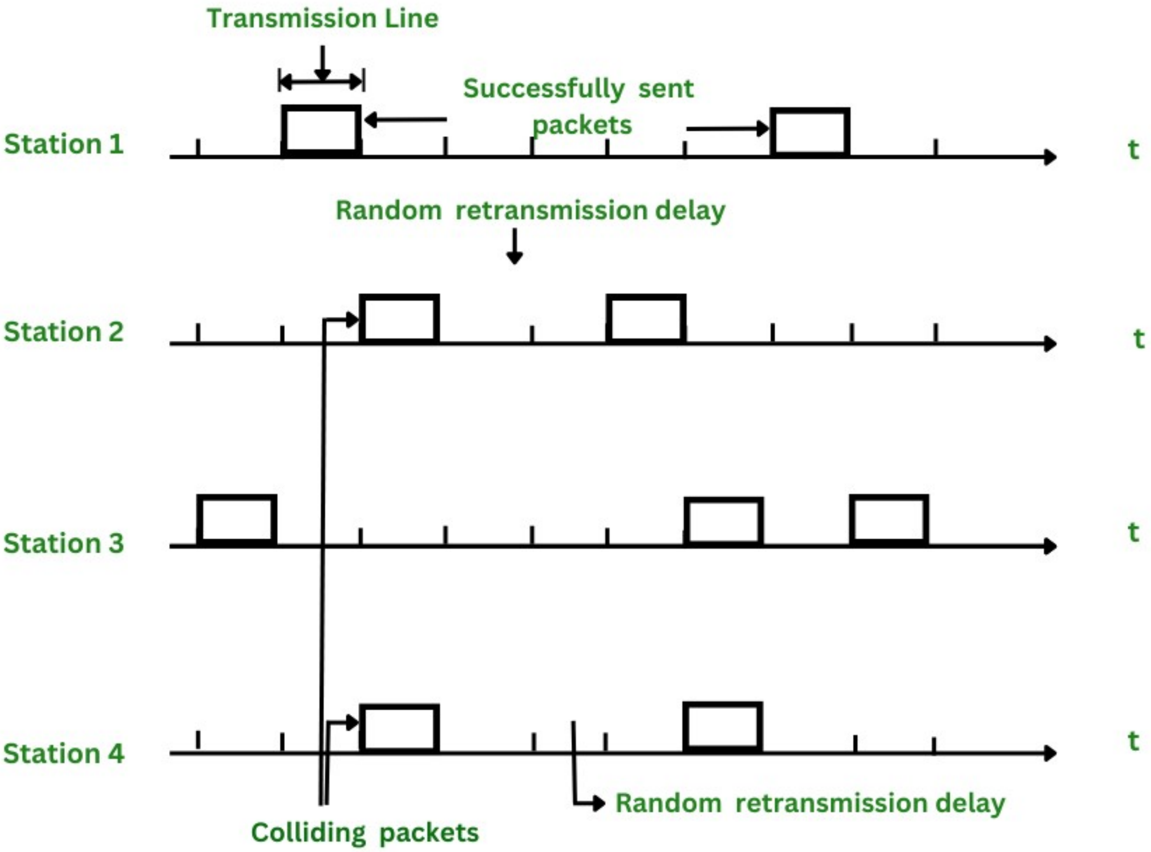

Transmission Line
Successfully sent packets
Station 1
t
Random retransmission delay
Station 2
t
Station 3
t
Station 4
t
Random retransmission delay
Colliding packets

Figure: *Transmission attempts and random retransmission delay for colliding packets in slotted ALOHA*

Throughput of Slotted ALOHA:

The quantity of successful transmissions at each time slot determines the throughput of the Slotted Aloha protocol. The Slotted Aloha protocol has a maximum throughput of about 18.4%. This is because there is a significant risk of collisions when numerous nodes try to transmit at the same time, which causes missed packets and a decreased overall throughput. When less than or equal to 37% of the network's total nodes are actively transmitting data, the maximum throughput is reached.

Due to the high frequency of collisions, the throughput of Slotted Aloha is typically substantially lower than 18.4% in practice It is not a widely used protocol in today's contemporary networks because of this.

The maximum throughput of a slotted ALOHA channel is given by the formula:

Throughput (S) = G x exp(-G)
The maximum Throughput occurs at G = 1,
i.e. S = 1/e = 0.368

Where: G = the offered load (or the number of packets being transmitted per time slot). The offered load is a measure of the number of nodes attempting to transmit in a given time slot.

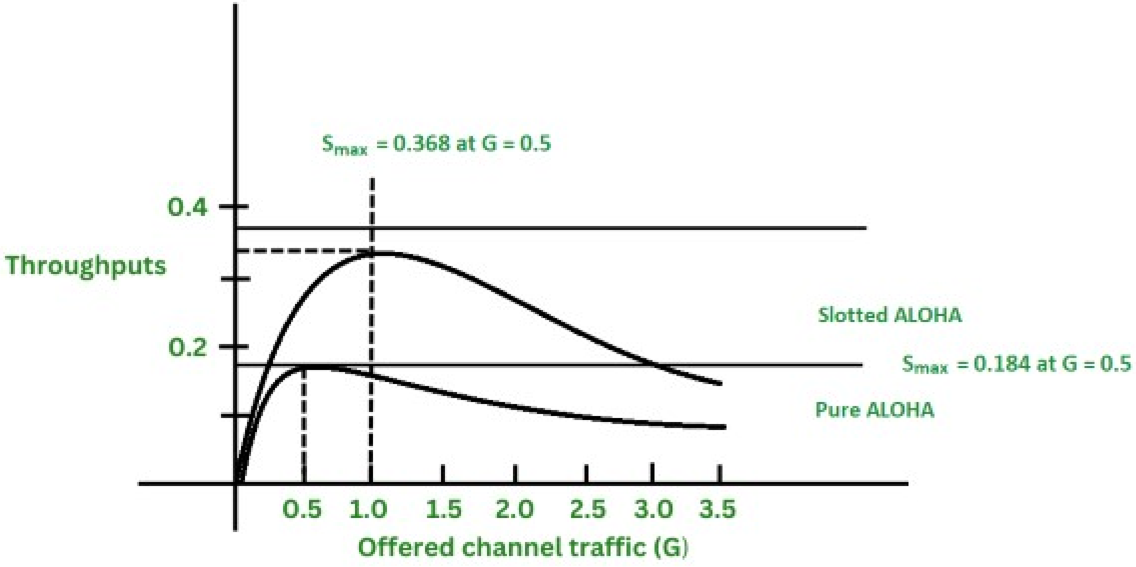


Figure: Comparison of the throughput as a function of offered load for Pure and Slotted ALOHA

The throughput is a function of the offered load and it ranges from 0 to 1. As the offered load increases, the throughput decreases as more collisions occur, resulting in less successful transmissions. The maximum throughput is achieved when the offered load is equal to 0.37 and it is approximately 0.184. It is important to note that the above equation assumes that all the packets are of the same length and that the channel is error-free. In practice, the throughput is usually much lower than this due to a number of factors such as packet errors, channel noise, and the overhead of retransmissions.

Assumption of Slotted ALOHA:

- All frames are of the same size.
- Time is divided into equal-sized slots, a slot equals the time to transmit one frame
- Nodes start to transmit frames only at beginning of slots.
- Nodes are synchronized.
- If two or more nodes transmit in a slot, all nodes detect collision before the slot ends.

Advantages of Slotted ALOHA:

- **Simplicity:** The Slotted Aloha protocol is relatively simple to implement and understand, making it an easy option for low-complexity networks.
- **Flexibility:** Slotted Aloha can be used in a wide range of network environments, including those with varying numbers of nodes and varying traffic loads
- **Low overhead:** Slotted Aloha does not require complex management or control mechanisms, which can help to reduce the overhead and complexity of the network.

Disadvantages of Slotted ALOHA:

- **Low throughput:** The maximum throughput of the Slotted Aloha protocol is relatively low at around 18.4%, which can be limiting for high-bandwidth applications.

- **High collision rate:** The high collision rate in slotted ALOHA can result in a high packet loss rate, which can negatively impact the overall performance of the network.
- **Inefficiency:** The protocol is inefficient at high loads, as the efficiency decreases as the number of nodes attempting to transmit increases.

Experimental Setup:

1. **Pure ALOHA Configuration:**
    - Set up a simple network topology with at least two nodes capable of data communication.
    - Implement the Pure ALOHA protocol on the nodes, specifying parameters such as frame size, transmission time, and acknowledgment time.
2. **Frame Transmission:**
    - Configure the nodes to generate data frames for transmission at random intervals.
    - Implement a collision detection mechanism to identify and handle frame collisions.
3. **Performance Metrics:**
    - Measure the round-trip time (RTT) for successful frame transmissions.
    - Record the number of collisions and identify their impact on the network efficiency.
    - Calculate the throughput of the Pure ALOHA protocol under different data loads.
4. **Varying Network Conditions:**
    - Introduce variations in network conditions such as increased traffic load and frame size.
    - Evaluate the effect of these variations on the protocol's performance.
5. **Data Collection and Analysis:**
    - Use a protocol analyzer tool (e.g., Wireshark) to capture and analyze the network traffic during the experiments.
    - Record the time stamps for frame transmissions, collisions, and acknowledgments.

Data Analysis:

1. Analyze the collected data to calculate the throughput of the Pure ALOHA protocol.
2. Evaluate the efficiency of the protocol by considering the impact of collisions on the overall performance.
3. Compare the observed performance metrics with the theoretical predictions for Pure ALOHA.
4. Identify any limitations or challenges encountered during the experiments.

Conclusion: Summarize the findings of the experiment, discussing the efficiency and limitations of the Pure ALOHA protocol. Provide insights into the impact of varying network conditions on the protocol's performance and draw conclusions regarding its suitability for different scenarios. Discuss potential improvements or modifications to enhance the efficiency of contention-based protocols in shared communication channels.

Experiment No. -10

Experiment Title: Study of Differential Manchester Code in Data Communication

Objective: The primary objective of this experiment is to study the principles and performance characteristics of the Differential Manchester encoding scheme in data communication. Differential Manchester encoding is a technique used for binary data transmission, providing advantages such as synchronization and error detection. The experiment aims to understand the encoding and decoding processes, evaluate the efficiency of the scheme, and observe its behavior under different conditions.

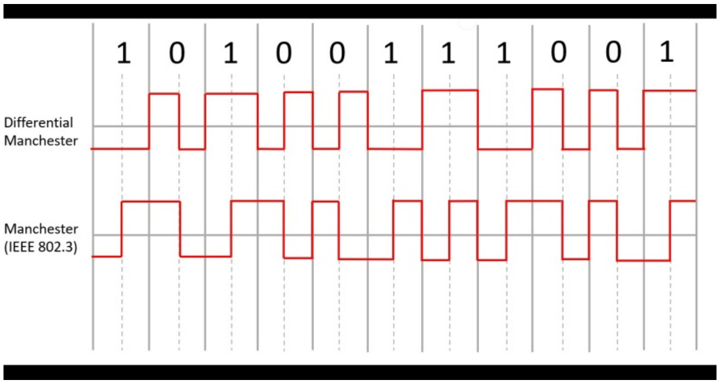
Equipment and Materials:

1. Computers or devices with network interfaces
2. Network simulation software (e.g., Cisco Packet Tracer, GNS3)
3. Ethernet cables
4. Data communication devices (such as PCs or laptops)
5. Oscilloscope or signal analyzer
6. Differential Manchester encoder/decoder (if available)
7. Signal generator (if available)

Differential Manchester encoding is a type of Manchester encoding, a digital signaling scheme used for data transmission. Unlike basic Manchester encoding, which uses both transitions (low to high or high to low) within each bit period, Differential Manchester encoding relies on transitions to represent the data.

In Differential Manchester encoding, the information is conveyed based on the presence or absence of transitions at the middle of each bit period. It ensures a clear distinction between 0s and 1s and provides self-clocking, making it easier for the receiver to synchronize with the incoming data.

Here's a detailed explanation of how Differential Manchester encoding works:



1. **Bit Representation:**
   - Each bit period is divided into two halves: the first half and the second half.
   - A transition at the middle of the bit period represents one type of bit (either 0 or 1), while the absence of a transition represents the other type.
2. **Transition Rules:**
   - For a "0" bit:
     - If the previous bit was a "0," a transition occurs in the middle of the bit period.
     - If the previous bit was a "1," there is no transition in the middle of the bit period.
   - For a "1" bit:
     - If the previous bit was a "0," there is no transition in the middle of the bit period.

        - If the previous bit was a "1," a transition occurs in the middle of the bit period.

3. **Waveform Examples:**
    - Consider the following scenarios:
        - If the previous bit is 0, and the current bit is 0, there is a transition in the middle.
        - If the previous bit is 0, and the current bit is 1, there is no transition in the middle.
        - If the previous bit is 1, and the current bit is 0, there is no transition in the middle.
        - If the previous bit is 1, and the current bit is 1, there is a transition in the middle.
    - This pattern ensures that each bit period has a transition, either from high to low or low to high, and the absence or presence of the transition indicates the value of the current bit.
4. **Advantages:**
    - **Self-Clocking:** The receiver can synchronize with the incoming data by detecting transitions, making clock recovery more straightforward.
    - **Reliable Data Transmission:** The clear distinction between 0s and 1s, combined with self-clocking, enhances the reliability of data transmission.
5. **Disadvantages:**
    - **Double the Bit Rate:** The bit rate of the signal is double the original data rate, as each bit is represented by two signal transitions.

In summary, Differential Manchester encoding is a differential encoding scheme that provides reliable data transmission by utilizing transitions in the middle of each bit period to represent information. Its self-clocking feature simplifies the synchronization process for the receiver.

Experimental Setup:

1. **Differential Manchester Encoding:**
    - Set up a simple network topology with at least two nodes capable of data communication.
    - Implement Differential Manchester encoding on the transmitting node.
    - Configure the encoding parameters such as bit rate, signal levels, and initial conditions.
2. **Frame Transmission:**
    - Generate binary data frames for transmission from the transmitting node.
    - Implement the encoding scheme to convert the binary data into Differential Manchester-encoded signals.
3. **Decoding Process:**
    - Implement Differential Manchester decoding on the receiving node.
    - Configure the decoding parameters to synchronize with the transmitted signal.
    - Record the decoded binary data at the receiving end.
4. **Signal Analysis:**
    - Use an oscilloscope or signal analyzer to visualize and analyze the Differential Manchester-encoded signals.
    - Observe the waveform and identify key features such as the transition points and signal structure.
5. **Performance Metrics:**
    - Measure the bit error rate (BER) by intentionally introducing noise or errors into the transmission.
    - Evaluate the synchronization capabilities of Differential Manchester encoding under varying conditions.
6. **Varying Conditions:**
    - Introduce variations in signal quality, such as changes in amplitude, phase, or noise levels.

- Evaluate the robustness of the Differential Manchester encoding scheme under these conditions.

Data Collection and Analysis:

1. Record the encoded and decoded signals for analysis.
2. Document the bit error rate and synchronization performance under different conditions.
3. Compare the observed performance with theoretical expectations for Differential Manchester encoding.

Conclusion: Summarize the findings of the experiment, highlighting the efficiency and characteristics of Differential Manchester encoding. Discuss its advantages, limitations, and applicability in practical data communication scenarios. Provide recommendations for using Differential Manchester encoding based on the experiment results and identify potential areas for further research or improvement.