

10/11/22  
Week 2

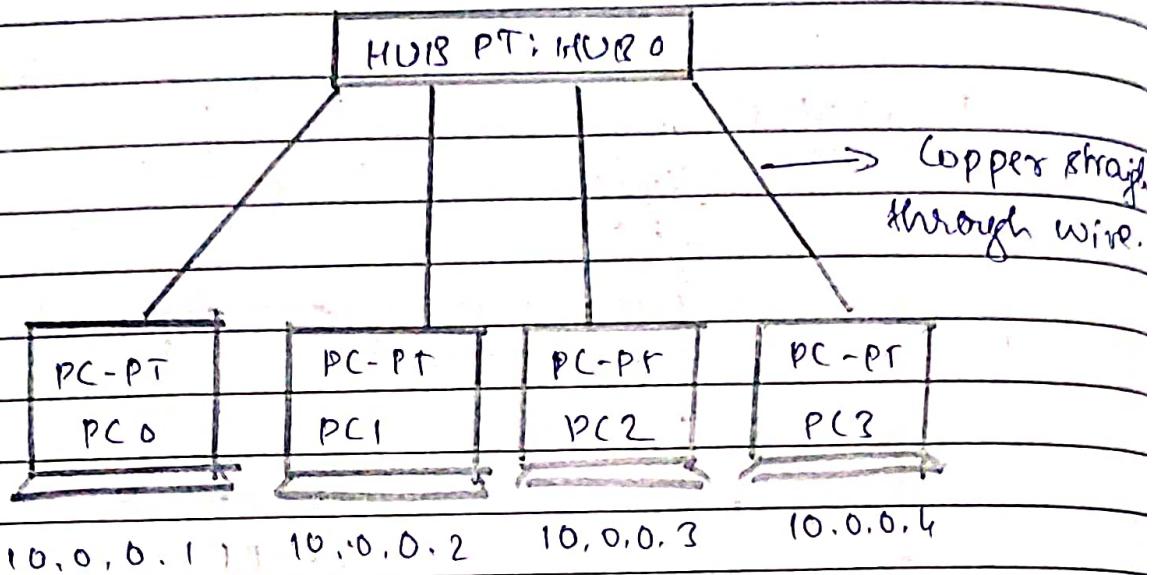
## Lab - 1

Date \_\_\_\_\_  
Page \_\_\_\_\_

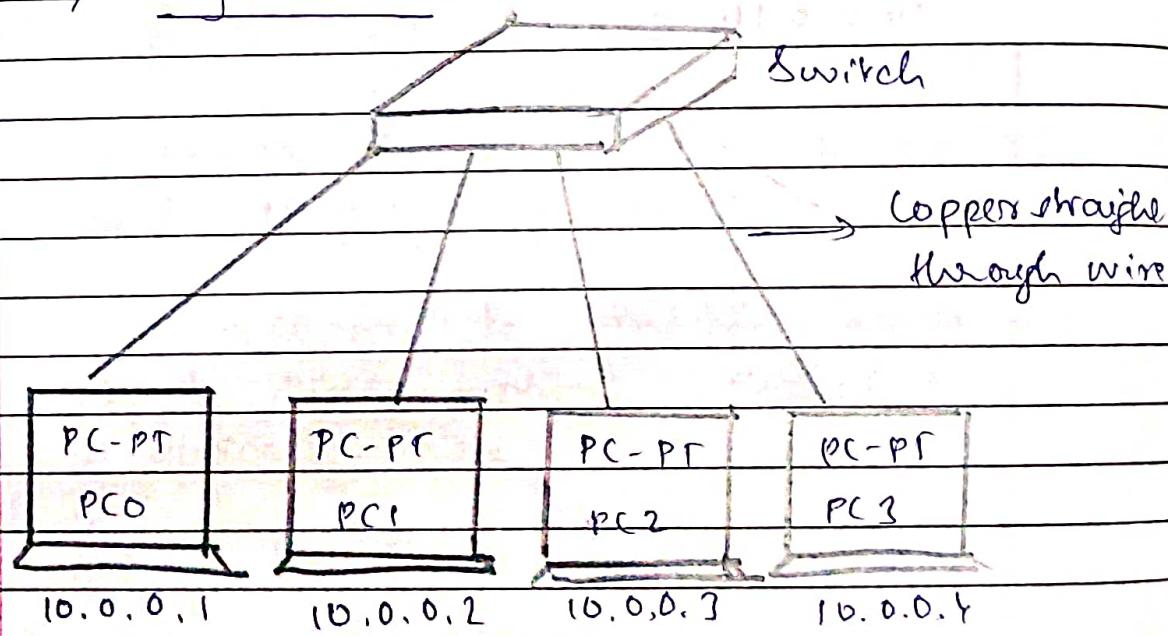
Aim: Creating a topology and stimulate sending a simple PDU from source to destination using HUB and Switch as connecting devices.

### TOPOLOGY:

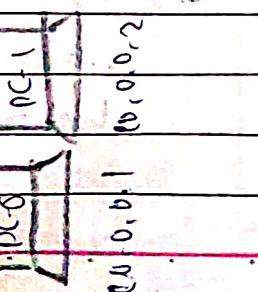
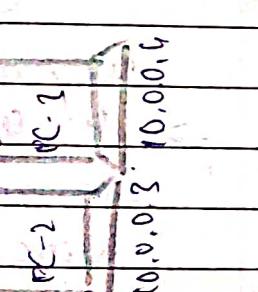
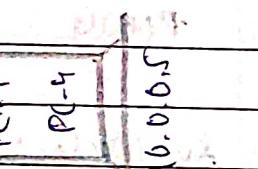
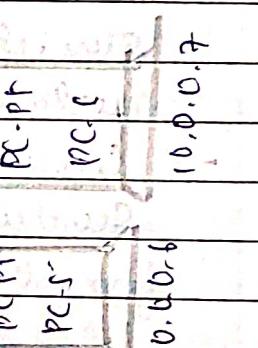
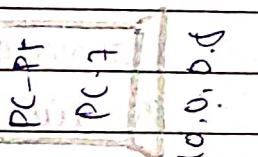
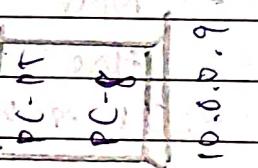
→ Using hub :-



→ Using switch:-



Date \_\_\_\_\_  
Page \_\_\_\_\_  
100.0.12



HUB-2

HUB-1

HUB-0

SUMITRA

## # Procedure:-

### Hubs :-

- A generic hub is placed in the logical workspace.
- 2 generic PC's are also placed and connected to hub by copper straight wire.
- Note is placed below PC's specifying the IP address used for those PC's.
- It is to be noted that hub and PC's are connected through fast ethernet connection.
- A simple PDU is placed on & devices and packet passing can be seen in simulation mode, after we click the auto capture.
- Command prompt is opened for the sending PC. Ping command is given for the destination IP address.

PING destination\_IP\_address.

### Switch :-

- A generic switch is placed in logical workspace.
- 4 generic PC's are also placed and connected to switch by copper straight wire.
- IP addresses for PC's are set and note is placed specifying same below PC's.

- In simulation mode, after selecting 2 devices implement simple PDU, click auto-capture.
- In real time mode, click on sending PC open command prompt and type ping destination - IP address.

### Hybrid:

- A switch, 3 hubs and 12 generic PC's are placed in logical workspace.
- 3 hubs are connected by switch using copper cross over wire and 12 generic PC's connected to 3 hubs (4 PCs each hub) using copper straight wire.
- Assign IP addresses to each PC and place notes below.
- Select 2 PC's from different hubs with applying simple PDU and click on auto-capture.
- In real time mode open command prompt by clicking on sending PC and give command ping destination - IP address.

### # Observations

\* Hub:

Working Outcome :-

After source sends message to hub, it broadcast to

all end devices, build only destination device ready and send response back to hub (for source to get response)  
→ Hub establishes connection to end devices quickly and signals by green-light (acknowledges)

Result:

PC → ping 10.0.0.3

pinging 10.0.0.3 with 32 bytes of data;

Reply from 10.0.0.3 byte=32 time=2ms

Reply from 10.0.0.3 byte=32 time=0ms

Reply from 10.0.0.3 byte=32 time=0ms

Ping statistics for 10.0.0.3

packets: sent=4, received=4, loss=0

→ Switch

learning outcome:

→ Unlike hub, here the switch doesn't give green signal immediately.

→ It takes some time and packets are sent once green signal is generated (known as learning time)

→ After switch gets packet, initially broadcasts packet to all end devices to locate destination.

→ Once destination located, message is sent to it only further.

Between source and destination

Results:

PING 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data.

Reply from 10.0.0.2 with bytes = 32 time = 1ms

Reply from 10.0.0.2 bytes = 32 time = 9ms

Reply from 10.0.0.2 bytes = 32 time = 0ms

Reply from 10.0.0.2 bytes = 32 time = 0ms

Ping statistics for 10.0.0.2

Packets sent = 4, received = 4, loss = 0

### \* Hybrid Network

Working Outcomes :

→ Switch and hubs are connected through copper crossover wire to indicate they belong to same layer.

→ Message from source PC is sent to switch. From switch message sent to hub having the destination PC.

→ But the hub sends (broadcasts) to all end device and only destination PC sends back acknowledgement  
 (Connection happens between source and destination PCs)

Result:

Ping 10.0.0.7

Pinging 10.0.0.7 with 32 bytes of data

Reply from 10.0.0.7 bytes = 32 time = 1ms TTL = 128

Reply from 10.0.0.7 bytes = 32 time = 20ms TTL = 128

Reply from 10.0.0.7 bytes = 32 time = 0ms TTL = 128

Reply from 10.0.0.7 bytes = 32 time = 0.019999999999999994 TTL = 120

ping statistics for 10.0.0.7:

Packets: Sent = 4, Received = 4, Lost = 0 (0.0% loss)

17/1/22

