Dhirubhai Ambani University

CT303: Digital Communications

Prof. Manish Kumar

Lab - 6

Name : Jeet Daiya

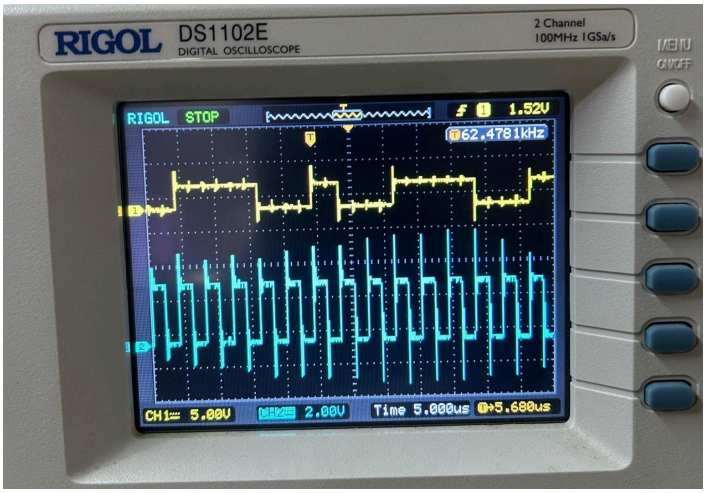
ID: 202301017 Lab Group: 1

Group:

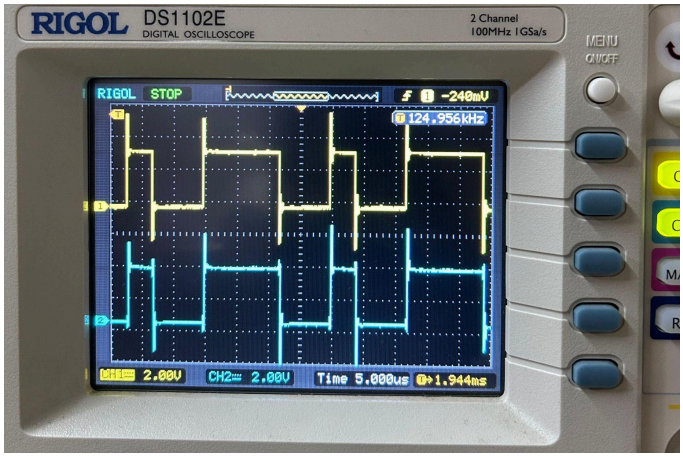
Dhruvil Mehta - 202301061
Jeet Daiya - 202301017
Yug Patel - 202301051
Jay Rathod - 202301006
Vansh Padaliya - 202301065

- **Experiment numbers 1 :**

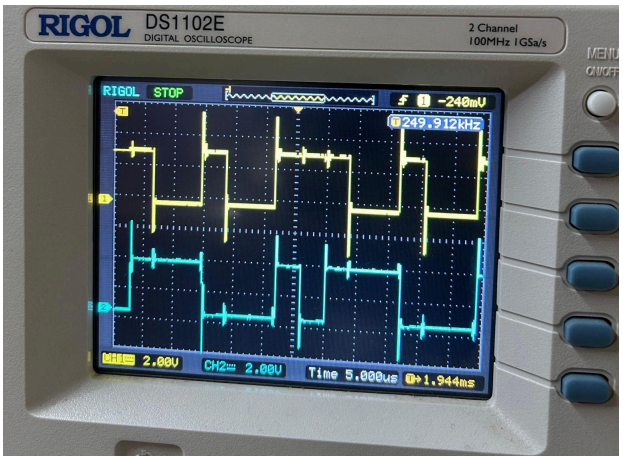
CH 1: DATA CLK (266 KHz) & CH 2: SERIAL DATA (00011011)



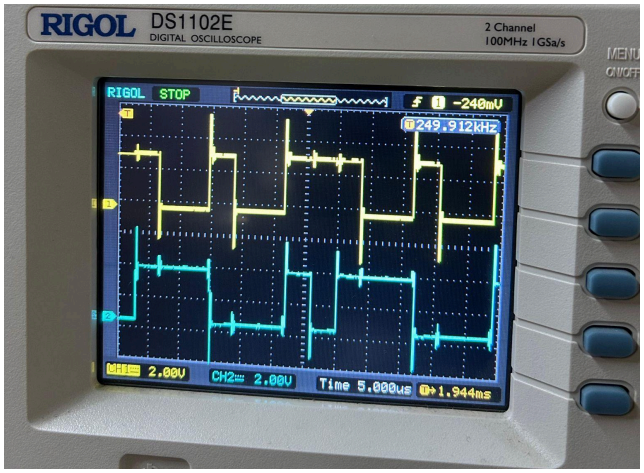
CH 1:DATA IN& CH 2:NRZ-L



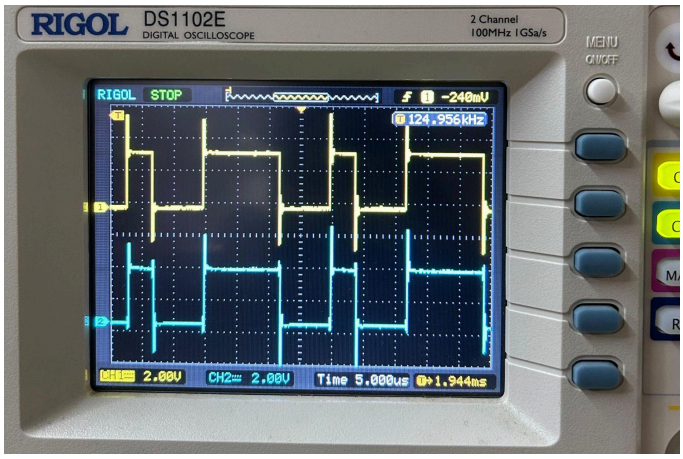
CH 1: DATA IN& CH 2: NRZ-M



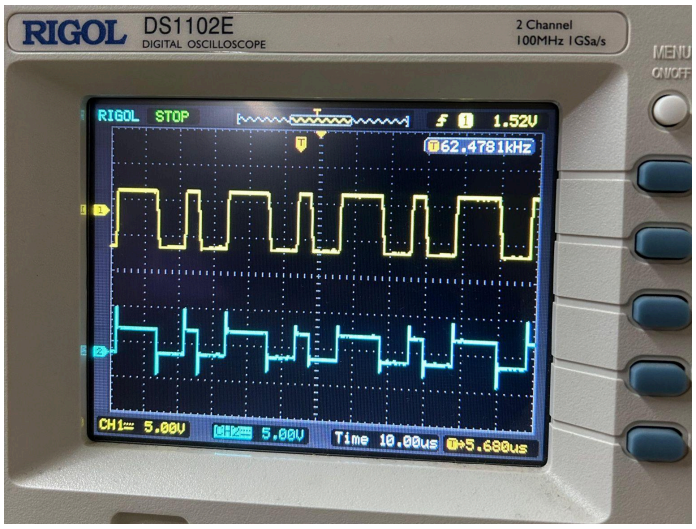
CH 1: DATA IN& CH 2: NRZ-M



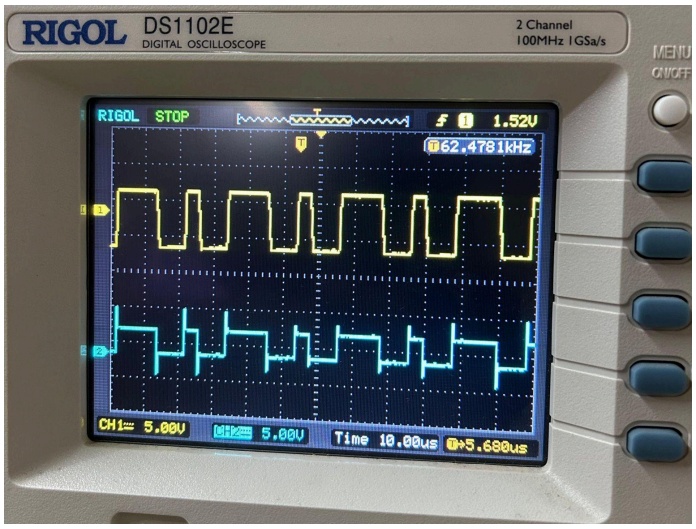
CH 1: DATA IN& CH 2: NRZ-S



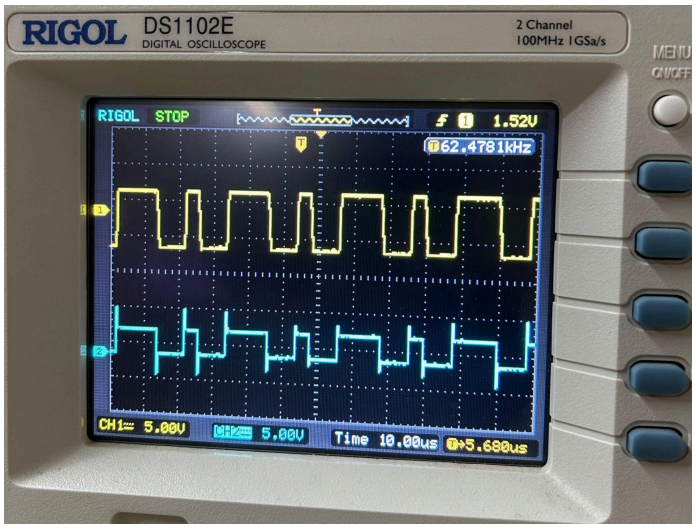
CH 1:DATA IN& CH 2: OUT1



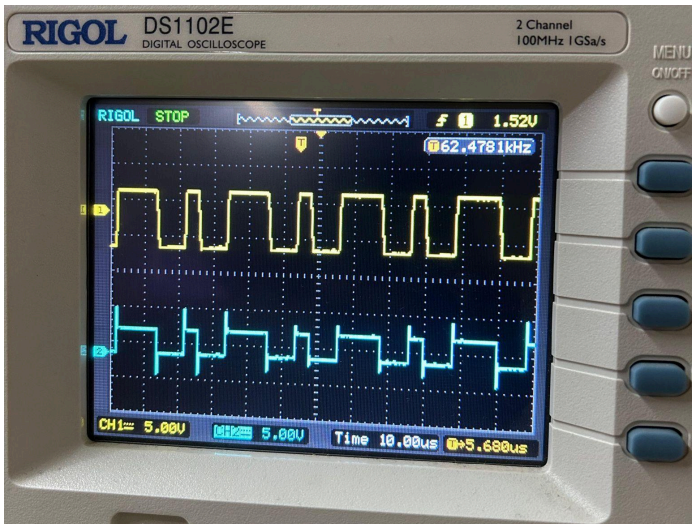
CH 1: DATA IN& CH 2: OUT2



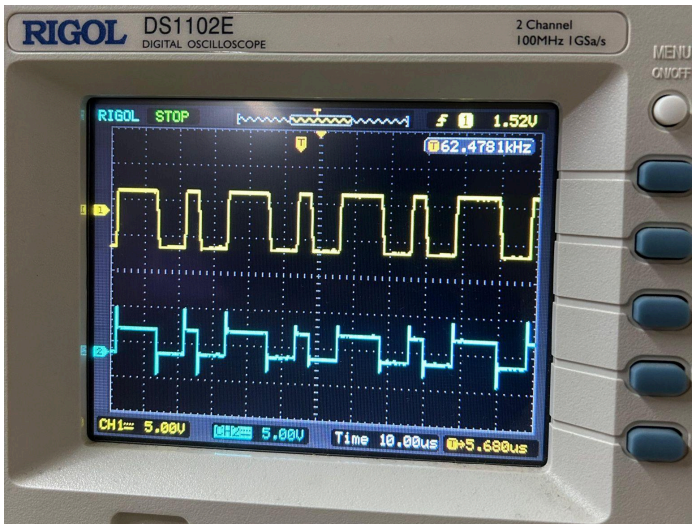
CH 1: DATA IN& CH 2: OUT3



CH 1: IN10& CH 2:OUT10

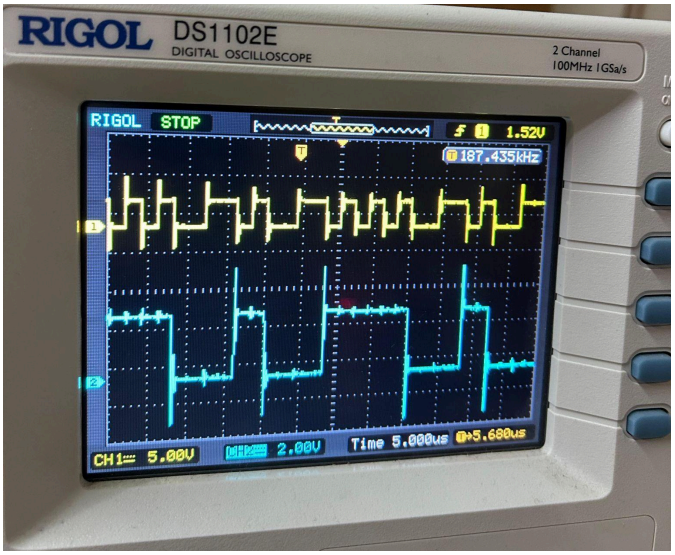


CH 1: IN10& CH 2: OUT11

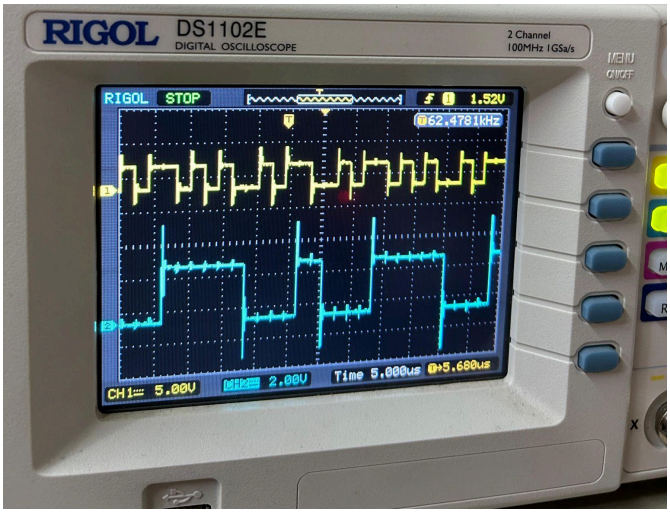


- ## Experiment numbers 2 :

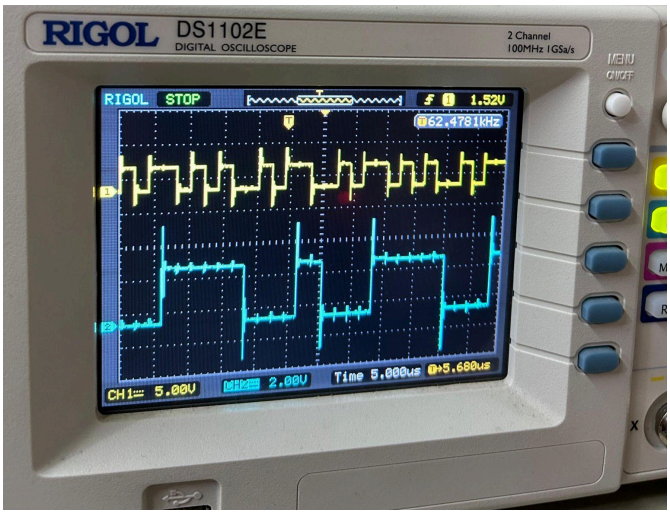
CH 1: DATA CLK (266 KHz) & CH 2: SERIAL DATA (00011011)



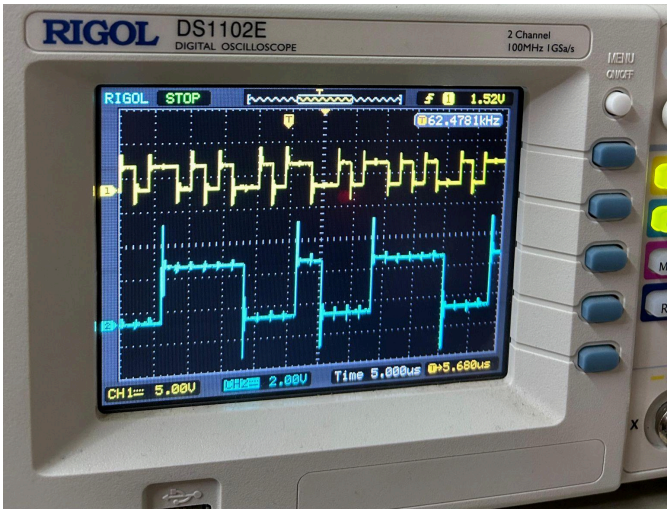
CH 1: DATA IN& CH 2: BIO-L



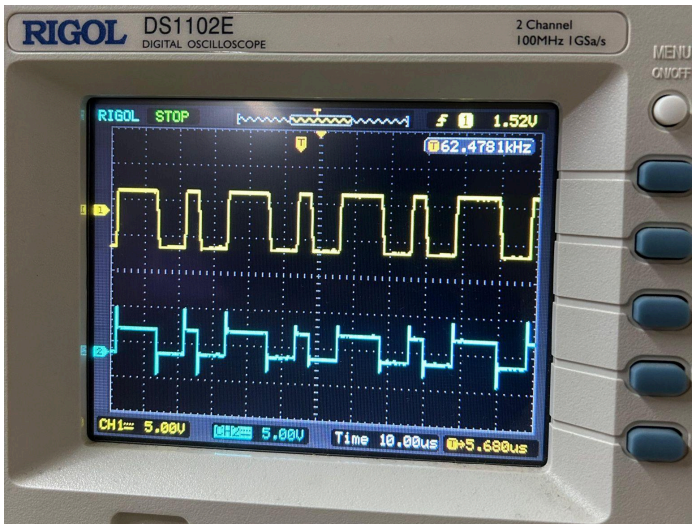
CH 1: DATA IN & CH 2: BIO-M



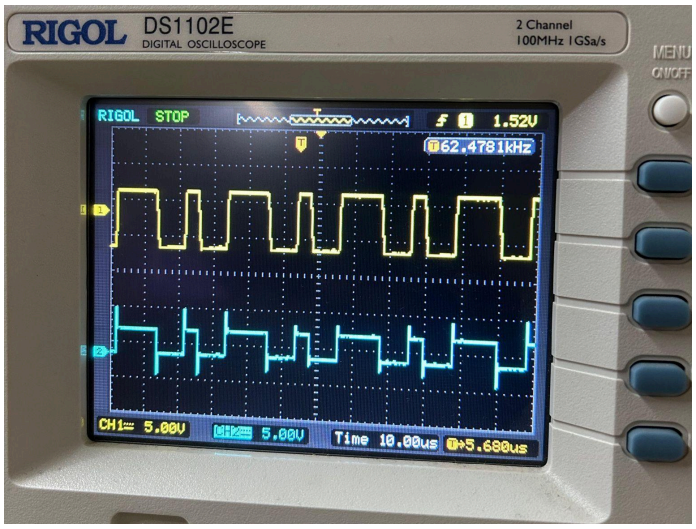
CH 1: DATA IN& CH 2: BIO-S



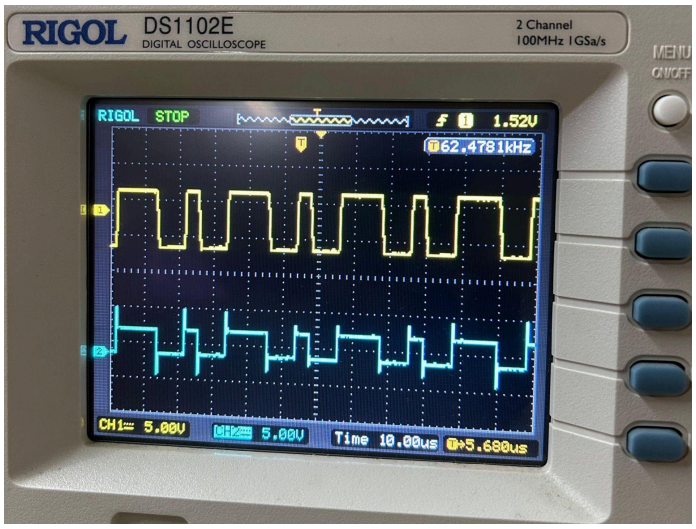
CH 1: DATA IN& CH 2: OUT5



CH 1: DATA IN& CH 2: OUT6

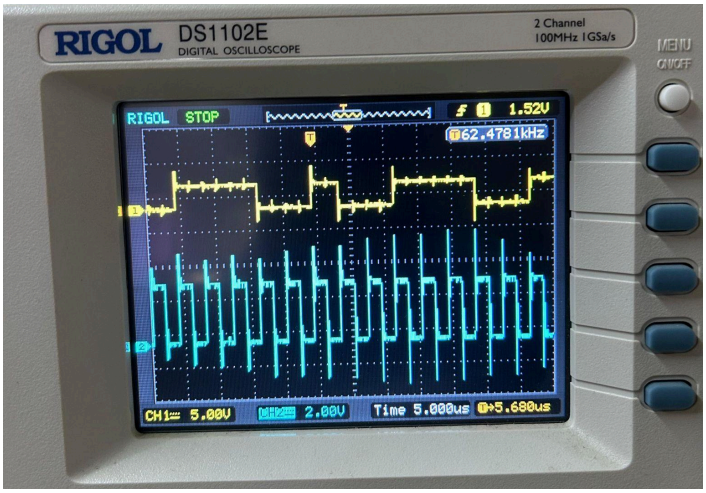


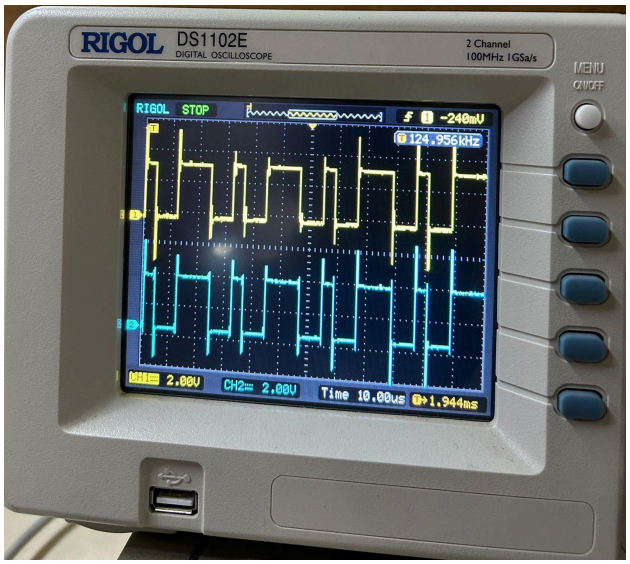
CH 1: DATA IN& CH 2: OUT7



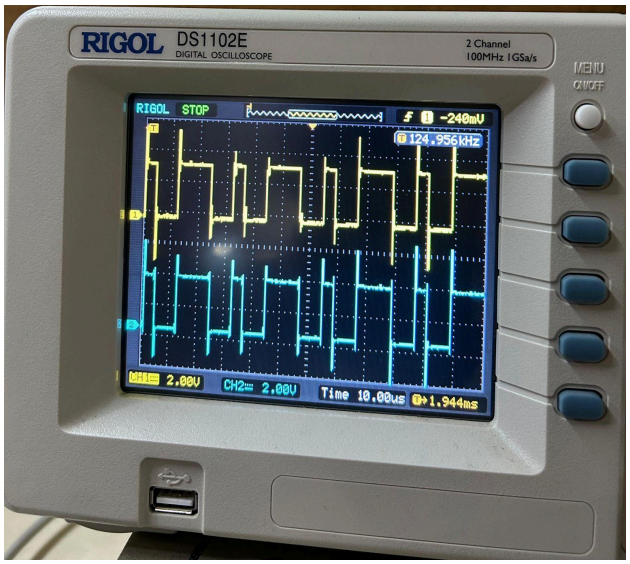
- **Experiment numbers 3 :**

CH 1: DATA CLK (266 KHz) & CH 2: SERIAL DATA (00011011)

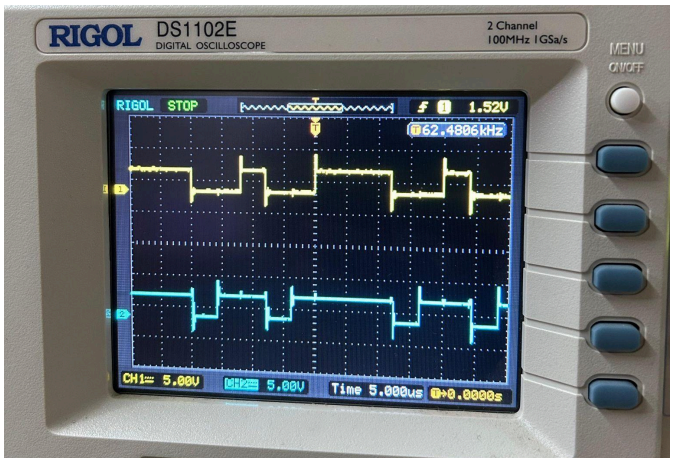




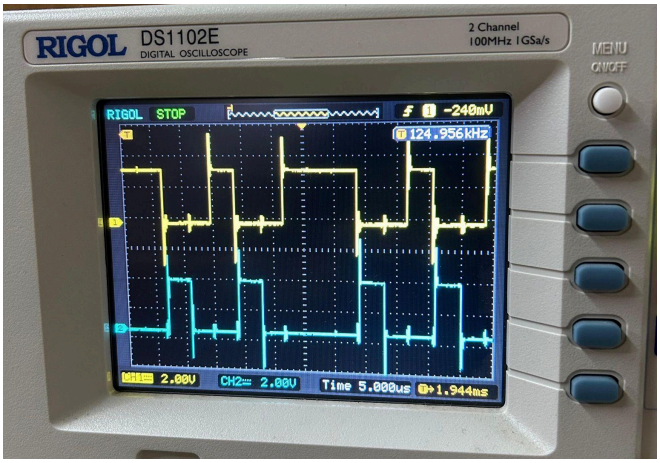
CH 1: DATA IN& CH 2: URZ



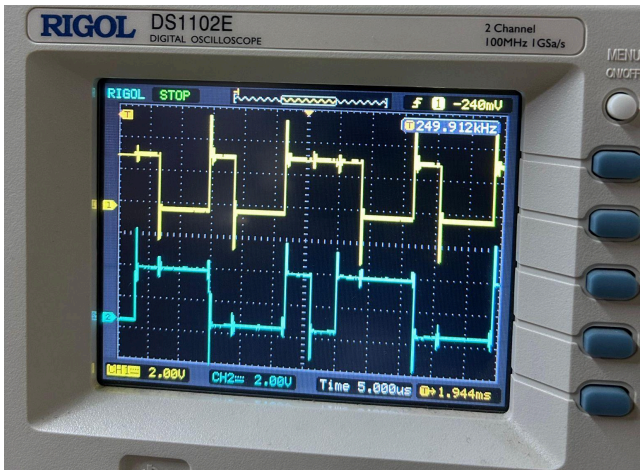
CH 1: DATA IN& CH 2: OUT4



CH 1: IN8 & CH 2: OUT8



CH 1: IN8 & CH 2: OUT9



- Problem 6.8 :

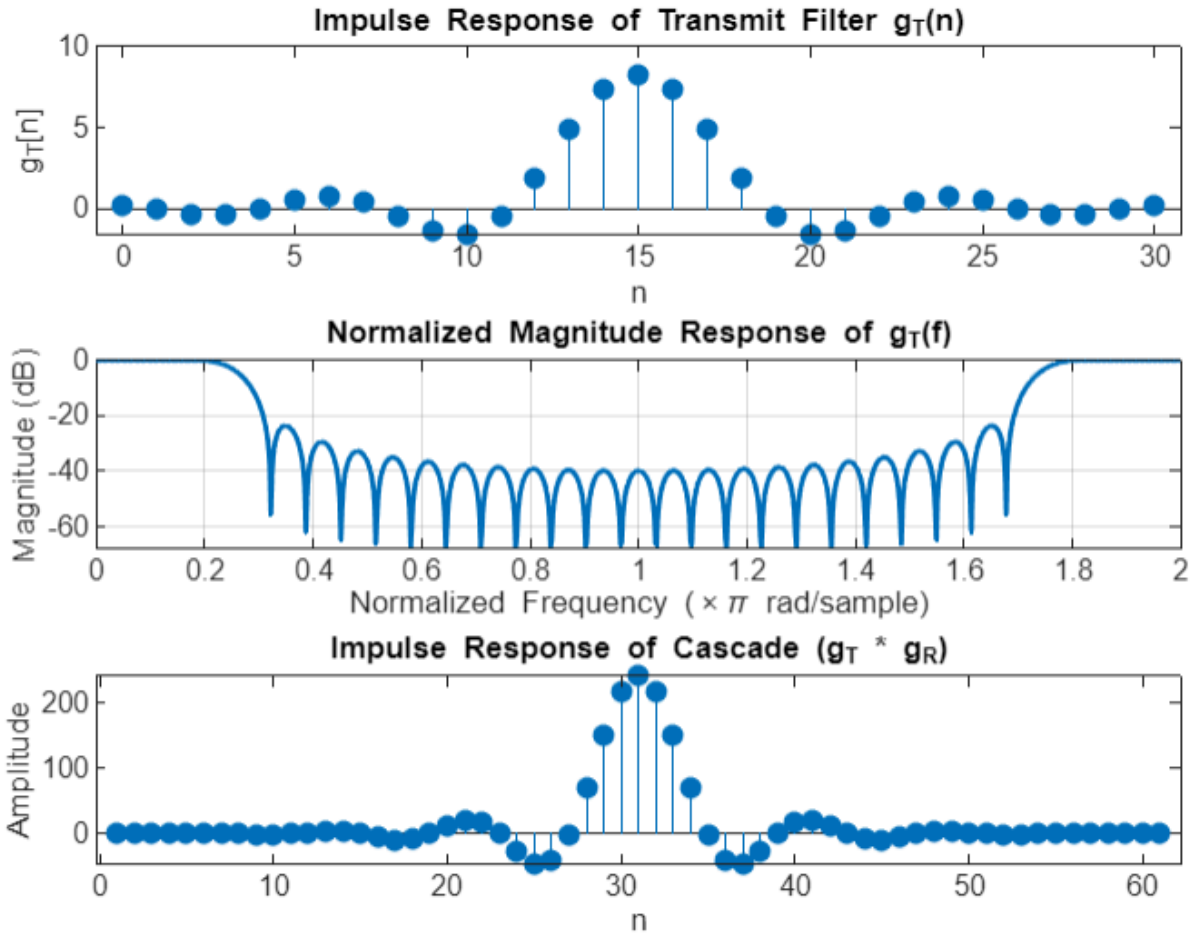
Code :

```
clc; clear; close all;

%% Parameters

N     = 31;         % Filter length (odd for symmetry)

T     = 1;          % Symbol period

alpha = 0.25;       % Roll-off factor

n     = -(N-1)/2:(N-1)/2;   % Time index

%% Compute transmitter filter impulse response g_T

g_T = zeros(1, length(n));   % initialize

for i = 1:length(n)

   for m = -(N-1)/2:(N-1)/2

       % Xrc() is the raised cosine spectrum function

       g_T(i) = g_T(i) + sqrt(xrc(4*m/(N*T), alpha, T)) * ...

                exp(1j*2*pi*m*n(i)/N);

   end

end
```

```
%% Take real part (imag part is numerical error)
g_T = real(g_T);
%% Delay to make linear phase
n2 = 0:N-1;
g_T_shifted = g_T;  % already symmetric, shift not mandatory for
FIR plots
%% Receiver filter (matched filter to g_T)
g_R = g_T;
%% Cascade impulse response
imp_resp_cascade = conv(g_T, g_R);
%% Frequency response of g_T
[GT, W] = freqz(g_T, 1, 1024, 'whole');
magGT_dB = 20*log10(abs(GT)/max(abs(GT)));
%% Plot results
figure;
subplot(3,1,1);
stem(n2, g_T_shifted, 'filled');
title('Impulse Response of Transmit Filter g_T(n)');
xlabel('n'); ylabel('g_T[n]');
subplot(3,1,2);
plot(W/pi, magGT_dB, 'LineWidth', 1.5);
title('Normalized Magnitude Response of g_T(f)');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude (dB)'); grid on;
subplot(3,1,3);
stem(imp_resp_cascade, 'filled');
```

```
title('Impulse Response of Cascade (g_T * g_R)');
xlabel('n'); ylabel('Amplitude');
%% Function: Raised Cosine Spectrum
function y = xrc(f, alpha, T)
   % Raised cosine frequency response (Xrc(f))
   if abs(f) > (1+alpha)/(2*T)
       y = 0;
   elseif abs(f) > (1-alpha)/(2*T)
       y = (T/2) * (1 + cos((pi*T/alpha) * ...
           (abs(f) - (1-alpha)/(2*T))));
   else
       y = T;
   end
end
```

- Output :
    - Photos :



🔹 Conclusion :

1. **Parameters:**
    - `N = 31` ➜ Filter length (odd so filter is symmetric).
    - `alpha = 0.25` ➜ Roll-off factor of the raised cosine.
    - `T = 1` ➜ Symbol period.

2. **Transmit Filter `g_T(n)`:**
    - Constructed using the **raised cosine frequency response (Xrc)**.
    - Take the real part to remove tiny imaginary numerical errors.
    - Since the filter is symmetric, the linear phase is already satisfied.
3. **Receiver Filter `g_R`:**
    - Same as the transmitter filter (matched filter).
4. **Cascade Response:**
    - Convolution of `g_T` and `g_R` gives the **overall system response** (transmitter + receiver).
5. **Frequency Response:**
    - Computed with `freqz`.
    - Normalized magnitude is shown in **dB**.

🔹 What the Output Shows

- **Figure (3 Subplots):**
    - **Impulse Response of g_T(n):**
      Looks symmetric ➔ ensures **linear phase** and no distortion in time domain.
    - **Magnitude Response of g_T(f):**
        - Flat in the passband (ideal transmission).

- Smooth transition band controlled by roll-off factor `α = 0.25`.
- Attenuates outside passband → avoids ISI (Inter-Symbol Interference).

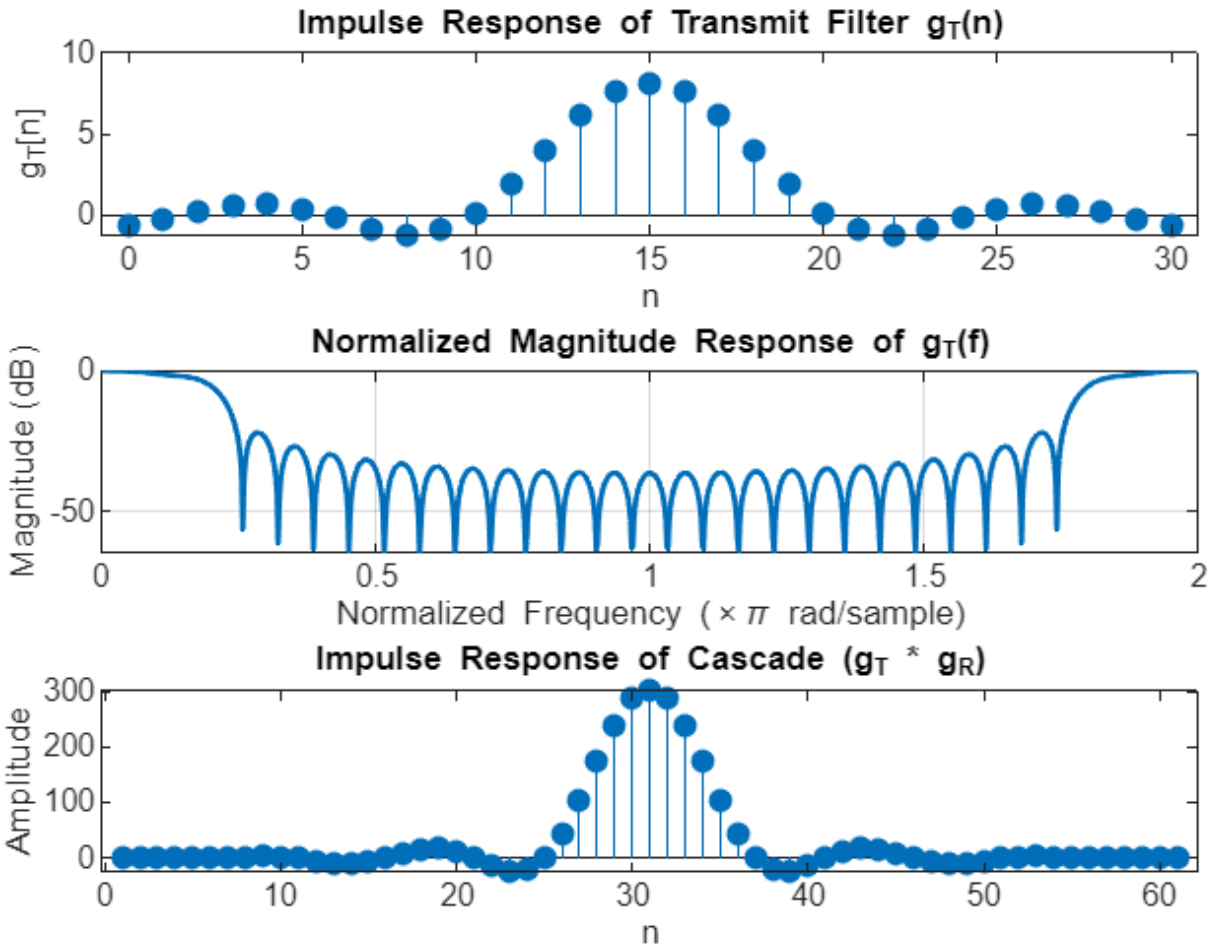
- **Cascade Impulse Response (g_T * g_R):**
  - Peaks sharply at center, small sidelobes → behave like a **Nyquist filter**.
  - Ensures **ISI-free transmission** when sampled at symbol intervals.

- Problem 6.9 :
  - Code :

```
clc; clear; close all;
%% Parameters
N = 31;           % Filter length (odd for symmetry)
T = 1;            % Symbol period
W = 1/(2*T);      % Bandwidth
n = -(N-1)/2:(N-1)/2;    % Time index
%% Compute transmitter filter impulse response g_T
g_T = zeros(1, length(n));  % initialize
for i = 1:length(n)
   for m = -(N-1)/2:(N-1)/2
       f_val = (4*m)/(N*T);  % frequency grid
       if abs(f_val) <= W
           g_T(i) = g_T(i) + sqrt((1/W) * cos((2*pi*m)/(N*T*W))) *
...
                   exp(1j*2*pi*m*n(i)/N);
       end
```

```
    end
end
%% Take real part (remove numerical error)
g_T = real(g_T);
%% Receiver filter (matched filter to g_T)
g_R = g_T;
%% Cascade impulse response
imp_resp_cascade = conv(g_T, g_R);
%% Frequency response of g_T
[GT, Wfreq] = freqz(g_T, 1, 1024, 'whole');
magGT_dB = 20*log10(abs(GT)/max(abs(GT)));
%% Plot results
figure;
subplot(3,1,1);
stem(0:N-1, g_T, 'filled');
title('Impulse Response of Transmit Filter g_T(n)');
xlabel('n'); ylabel('g_T[n]');
subplot(3,1,2);
plot(Wfreq/pi, magGT_dB, 'LineWidth', 1.5);
title('Normalized Magnitude Response of g_T(f)');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('Magnitude (dB)'); grid on;
subplot(3,1,3);
stem(imp_resp_cascade, 'filled');
title('Impulse Response of Cascade (g_T * g_R)');
xlabel('n'); ylabel('Amplitude');
```

- Output :
  - Photos :



🔹 Conclusion :

1. **Parameters:**
   - N = 31 ➜ Filter length, odd for symmetry.
   - T = 1 ➜ Symbol period.

- `W = 1/(2T)` → Bandwidth of the filter.
- `n` → Discrete-time index vector.

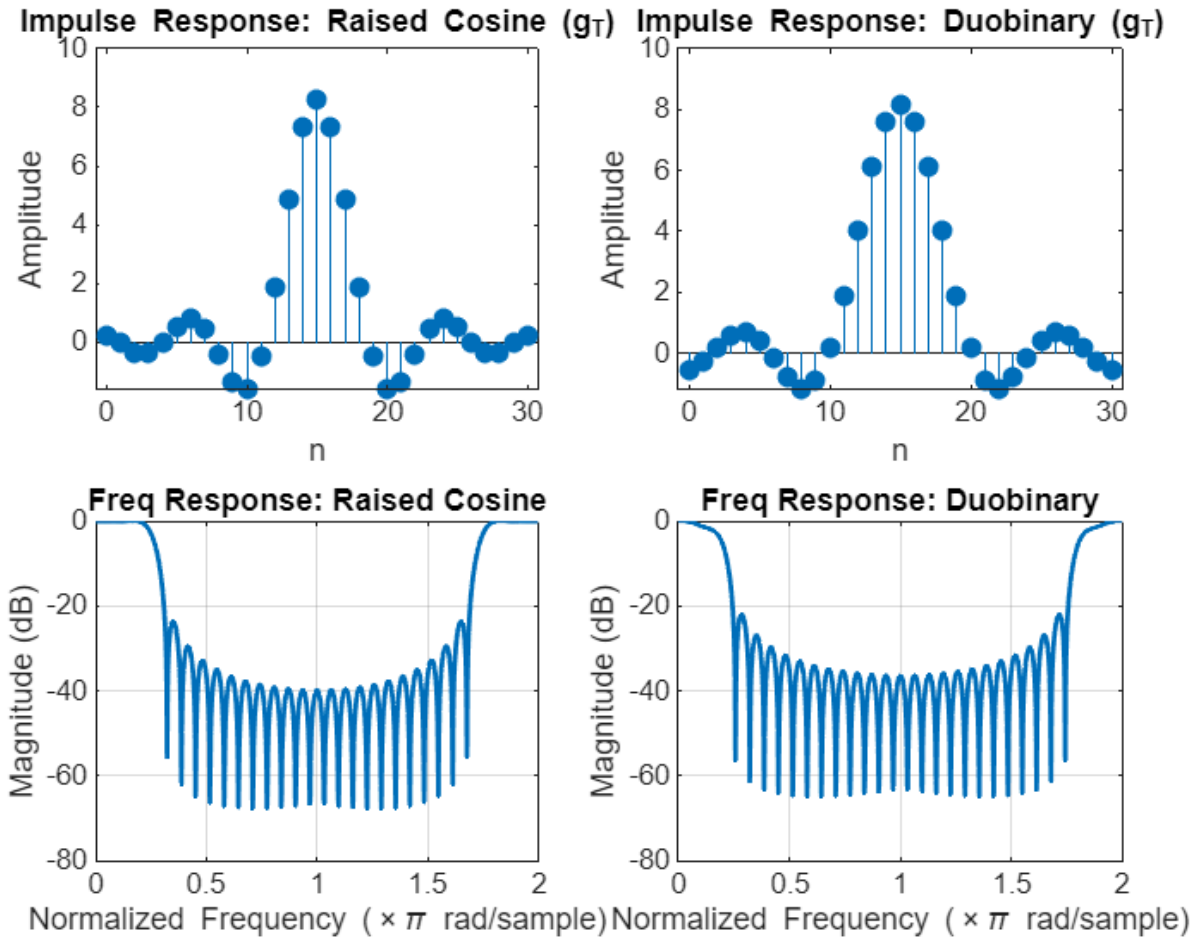
2. **Transmit Filter `g_T(n)`:**
   - Constructed by summing over frequency samples (`m`).
   - Spectrum is shaped using a **cosine weighting function** inside the passband.
   - Impulse response is real and symmetric → ensures **linear phase**.
3. **Receiver Filter `g_R`:**
   - Same as the transmit filter (matched filter).
4. **Cascade Response:**
   - Convolution of `g_T` with `g_R` gives the **overall system impulse response**.
5. **Frequency Response:**
   - Computed using `freqz`.
   - Normalized magnitude plotted in **dB**.

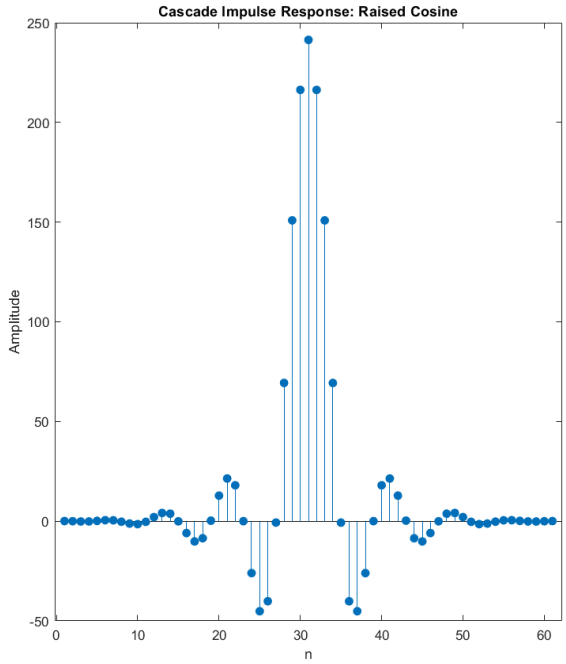
🔹 What the Output Shows

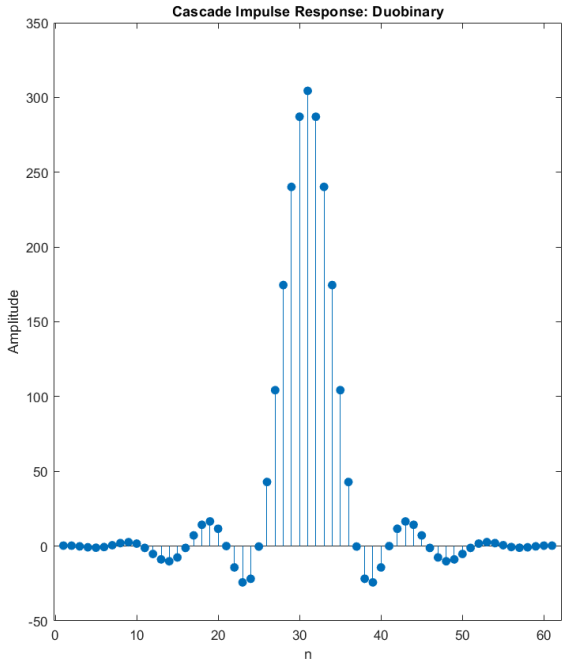
- **Figure (3 Subplots):**
  - **Impulse Response of g_T(n):**
    - Symmetric about the center.

        - Indicates **linear-phase FIR filter**.
        - Smooth decay on both sides of the main lobe.
    - **Normalized Magnitude Response of g_T(f):**
        - Passband is flat (good transmission within bandwidth).
        - Cosine roll-off shape defines the transition band.
        - Stopband attenuates unwanted frequencies → reduces interference.
    - **Cascade Impulse Response (g_T * g_R):**
        - Sharply peaked at the center.
        - Small sidelobes appear due to finite filter length.
        - Acts like a **Nyquist filter** → ensures minimal Inter-Symbol Interference (ISI).

- Photos of merge both problems :




Cascade Impulse Response: Raised Cosine
Amplitude
n
250
200
150
100
50
0
-50
0
10
20
30
40
50
60


Cascade Impulse Response: Duobinary
Amplitude
n
350
300
250
200
150
100
50
0
-50
0
10
20
30
40
50
60

Dhirubhai Ambani University

IT314: Computer Networks

Prof. Sanjary Srivastava

Lab 3: Configuring LAN Services Cluster

Group-93

Name	Student ID
Daiya Jeet	202301017
Vansh Padaliya	202301065
Kartik Vyas	202301003

1) **Assign IP addresses to each PC, grouping 2 PCs in one subnet and the 3rd PC in a different subnet.**
   - **Test connectivity within the PCs of same subnet.**
   - **Test connectivity within the PCs of two different subnets**

Commands :

Configure Network Interfaces on Each PC:

Step 1: Identify the network interfaces of each PC using the Command: ip a

Step 2: Assign IP addresses to each PC, grouping 2 PCs in one subnet and the 3rd PC in a different subnet.

- **For PC1 (subnet 1):**

```
sudo ip addr add 192.168.10.8/24 dev enp2s0
sudo ip link set enp2s0 up
```

- **For PC2 (subnet 1):**

```
sudo ip addr add 192.168.10.10/24 dev enp2s0
sudo ip link set enp2s0 up
```

- **For PC3 (subnet 2):**

```
sudo ip addr add 192.168.20.9/16 dev enp2s0
sudo ip link set enp2s0 up
```

Explanation: PC1 and PC2 are in the same subnet 192.168.10.0/24, allowing them to communicate directly with each other. PC3 is in a different subnet 192.168.20.0/24 and cannot directly communicate with PC1 and PC2 without routing.

Verifying Network Configuration:

Step 1: Check the assigned IP addresses:

Command: `ip a`

Step 2: Test connectivity within the same subnet:

1) From PC1, ping PC2:
   Eg: `ping 192.168.10.10`

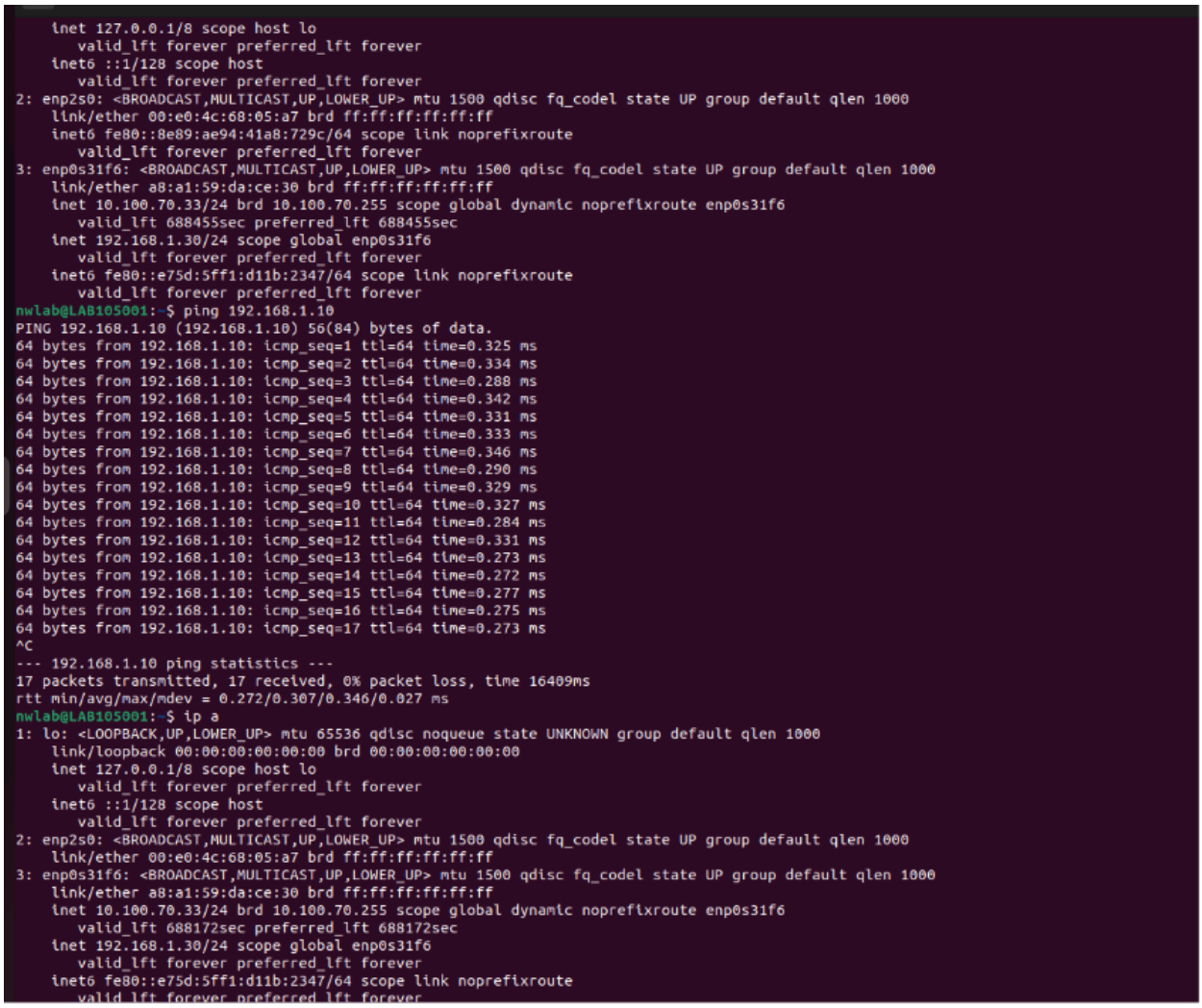
2) From PC2, ping PC1:
   Eg: `ping 192.168.10.8`

3) From PC3, ping PC1 or PC2 (should fail):
   Eg: `ping 192.168.10.10`

Output:

```
nwlab@LAB105001:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:e0:4c:68:2b:75 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::2e0:4cff:fe68:2b75/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether a8:a1:59:da:cd:60 brd ff:ff:ff:ff:ff:ff
    inet 10.100.70.49/24 brd 10.100.70.255 scope global dynamic noprefixroute enp0s31f6
       valid_lft 689390sec preferred_lft 689390sec
    inet6 fe80::8721:ace:cd34:6183/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
nwlab@LAB105001:~$ sudo ip addr add 192.168.2.9/16 dev enp2s0
[sudo] password for nwlab:
nwlab@LAB105001:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:e0:4c:68:2b:75 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.9/16 scope global enp2s0
       valid_lft forever preferred_lft forever
3: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether a8:a1:59:da:cd:60 brd ff:ff:ff:ff:ff:ff
    inet 10.100.70.49/24 brd 10.100.70.255 scope global dynamic noprefixroute enp0s31f6
       valid_lft 689137sec preferred_lft 689137sec
    inet6 fe80::8721:ace:cd34:6183/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
nwlab@LAB105001:~$ ping 192.168.1.10
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
From 192.168.2.9 icmp_seq=1 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.2.9 icmp_seq=2 Destination Host Unreachable
From 192.168.2.9 icmp_seq=3 Destination Host Unreachable
From 192.168.2.9 icmp_seq=5 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.2.9 icmp_seq=6 Destination Host Unreachable
From 192.168.2.9 icmp_seq=7 Destination Host Unreachable
From 192.168.2.9 icmp_seq=9 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.2.9 icmp_seq=10 Destination Host Unreachable
From 192.168.2.9 icmp_seq=11 Destination Host Unreachable
From 192.168.2.9 icmp_seq=13 Destination Host Unreachable
ping: sendmsg: No route to host
From 192.168.2.9 icmp_seq=14 Destination Host Unreachable
From 192.168.2.9 icmp_seq=15 Destination Host Unreachable
From 192.168.2.9 icmp_seq=17 Destination Host Unreachable
From 192.168.2.9 icmp_seq=18 Destination Host Unreachable
```

(Ping from different subnet)



(Ping from same Subnet)

Questions :

1) **Explain the differences between a LAN and a sub-LAN. Why would you use sub-LAN with LAN ?**
    - LAN: A Local Area Network connects devices within a small area (like an office or home)
    - Sub-LAN: A smaller division of a LAN created using subnetting.
    - We use Sub-LAN to improve security, reduce congestion and manage traffic better

2) **Explain the role of subnet masks in dividing networks. How does the subnet mask in this experiment determine which PCs are in the same or different sub-LANs ?**
    - Subnet mask separates the network part and host part of an IP address.
    - PCs with the same network portion (as per mask) are in the same sub-LAN.
    - Different network portions -> different sub-LANs.

3) **Given the configuration above, if PC3 needed to communicate with PC1 and PC2, what additional network configuration would be necessary?**
    - If PC3 is in a different sub-LAN, it needs a router (or default gateway) configured.
    - This allows communication between sub-LANs.

4) **What challenges might you face if you were to add more devices to each sub-LAN?**

- IP conflicts if addresses are not managed.
- Limited IP addresses within each sub-LAN.
- More broadcast traffic slowing performance.
- Need for better switches/routers as network grows

2)

Server Side:

- **Install the isc-dhcp-server package, which provides the DHCP server functionality needed to assign IP addresses to devices on your network.**
- **Configure the DHCP Server, i.e define the network settings for your DHCP server.'**
- **Start the DHCP Server Service, so that it can begin assigning IP addresses to clients on the specified subnet.**
- **Check the IP(which is assigned dynamically).**

Client Side:

- **Configure the network settings file of your system,and set up the 'enp2s0' network interface to use DHCP to obtain an IP address automatically. (check by 'ip a' for the interface name of your system)**
- **Restart Network manager and systemd-networkd services to apply the changes.**
- **Verify the address obtained from DHCP server**

COMMANDS

SERVER SIDE:

1) **Updating app packages:**

   COMMAND: `sudo apt update`

   Explanation: This command updates the list of available packages and their versions.

2) **Installing isc-dhcp-server package**

   COMMAND: `sudo apt install isc-dhcp-server`

   Explanation: This command installs the isc-dhcp-server package, which provides the DHCP server functionality needed to assign IP addresses to devices on your network.

3) **Checking isc-dhcp-server is installed correctly or not**
   a) **Check the version of the isc-dhcp-server**

      COMMAND : `dhcpd<space>- -version`

      OUTPUT : isc-dhcpd-4.4.1 (or any version)

   b) **Check the status of DHCP server**

      COMMAND : `sudo systemctl status isc-dhcp-server`

      If the ACTIVE: failed is showed, then you need to configure the config file

4) Configuring the DHCP server:

a) Enter the config file of dhcpd

COMMAND : `sudo nano<space>/etc/dhcp/dhcpd.conf`

Explanation: This opens the DHCP server's main configuration file in the nano text editor. Here, you will define the network settings for your DHCP server.

b) Enter the following configuration code:

```
subnet 192.168.10.0 netmask 255.255.255.0

{

range 192.168.10.0<space>192.168.10.150;

option routers 192.168.10.1;

option subnet-mask 255.255.255.0;

option domain-name-servers 192.168.10.1;

}
```

Save File : CTRL + O

c) Test the config file:

COMMAND: `sudo dhcpd -t -cf /etc/dhcp/dhcpd.conf`

Explanation: This command checks the syntax of the DHCP configuration file to ensure there are no errors before starting the server.

d) **Open default file:**

COMMAND: `sudo nano /etc/default/isc-dhcp-server`

ENTER: INTERFACESv4="enp2s0" (Replace empty “” with enp2s0)

Explanation: This file defines the network interface that the DHCP server should listen on. Replace the empty quotes with enp2s0 to specify the network interface that the DHCP server will use to serve IP addresses.

5) Start the service

COMMAND: `sudo systemctl start isc-dhcp-server`

Explanation: This command starts the DHCP server so it can begin assigning IP addresses to clients on the specified Subnet

6) Check the status again (Command at point 3-b.)

If the DHCP port is disabled, enable it with the following Command:

COMMAND : `sudo systemctl enable isc-dhcp-server`

Explanation: This command ensures that the DHCP server starts Automatically on boot.

7) Check the IP

COMMAND : `ip a`

Explanation: If the ip for enp2s0 is absent, it means the network interface does not have an IP assigned, so now reassign the IP Manually

8) Reassign IP

COMMAND: `sudo ip addr add 192.168.10.29/24 dev enp2s0`

RECOMMENDED IP: 192.168.10.<System No.>

9) Restart the DHCP service

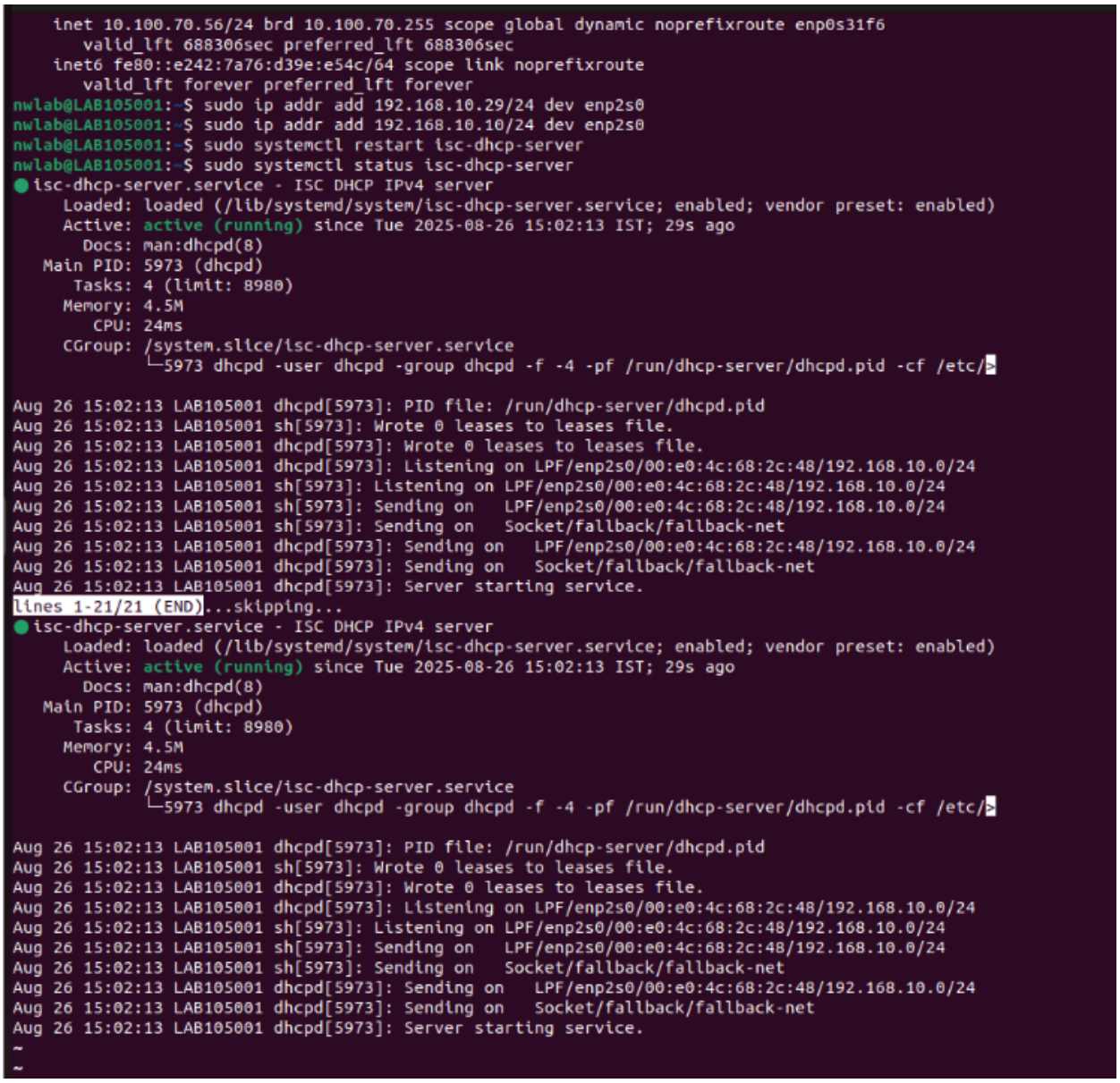
COMMAND: `sudo systemctl restart isc-dhcp-server`

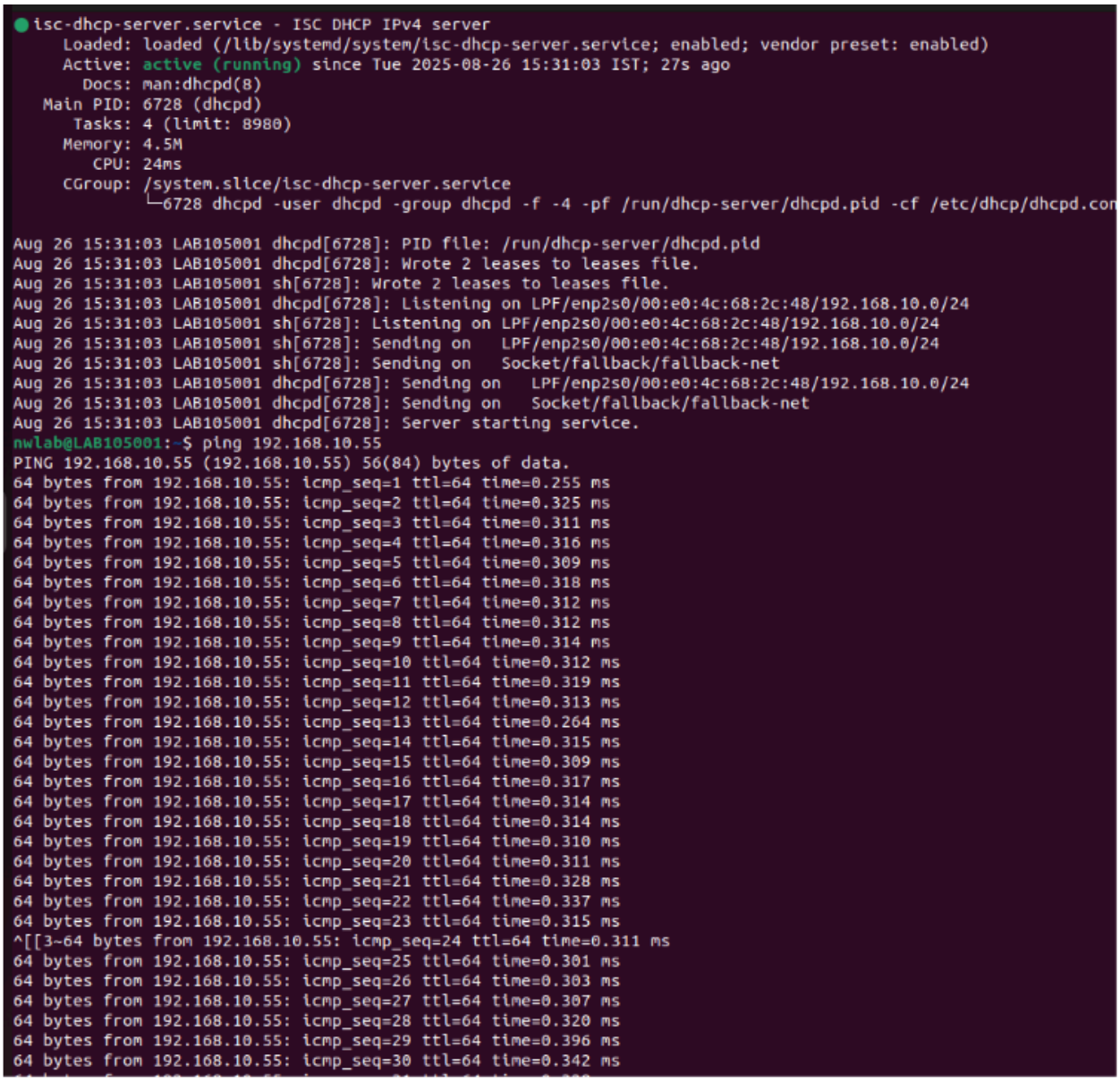
Explanation: Restart the DHCP server to apply any changes made during the configuration process

10) Check the status again (Command at point 3-b.)

11) ACTIVE: failed? Check whether the system was used as a DHCP client, then undo the process you did for DHCP client. (i.e do it for DHCP Server from the beginning

OUTPUT:


inet 10.100.70.56/24 brd 10.100.70.255 scope global dynamic noprefixroute enp0s31f6
valid_lft 688306sec preferred_lft 688306sec
inet6 fe80::e242:7a76:d39e:e54c/64 scope link noprefixroute
valid_lft forever preferred_lft forever
nwlab@LAB105001:~$ sudo ip addr add 192.168.10.29/24 dev enp2s0
nwlab@LAB105001:~$ sudo ip addr add 192.168.10.10/24 dev enp2s0
nwlab@LAB105001:~$ sudo systemctl restart isc-dhcp-server
nwlab@LAB105001:~$ sudo systemctl status isc-dhcp-server
isc-dhcp-server.service - ISC DHCP IPv4 server
Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
Active: active (running) since Tue 2025-08-26 15:02:13 IST; 29s ago
Docs: man:dhcpd(8)
Main PID: 5973 (dhcpd)
Tasks: 4 (limit: 8980)
Memory: 4.5M
CPU: 24ms
CGroup: /system.slice/isc-dhcp-server.service
└─5973 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/
Aug 26 15:02:13 LAB105001 dhcpd[5973]: PID file: /run/dhcp-server/dhcpd.pid
Aug 26 15:02:13 LAB105001 sh[5973]: Wrote 0 leases to leases file.
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Wrote 0 leases to leases file.
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Listening on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 sh[5973]: Listening on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 sh[5973]: Sending on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 sh[5973]: Sending on Socket/fallback/fallback-net
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Sending on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Sending on Socket/fallback/fallback-net
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Server starting service.
lines 1-21/21 (END)...skipping...
isc-dhcp-server.service - ISC DHCP IPv4 server
Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
Active: active (running) since Tue 2025-08-26 15:02:13 IST; 29s ago
Docs: man:dhcpd(8)
Main PID: 5973 (dhcpd)
Tasks: 4 (limit: 8980)
Memory: 4.5M
CPU: 24ms
CGroup: /system.slice/isc-dhcp-server.service
└─5973 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/
Aug 26 15:02:13 LAB105001 dhcpd[5973]: PID file: /run/dhcp-server/dhcpd.pid
Aug 26 15:02:13 LAB105001 sh[5973]: Wrote 0 leases to leases file.
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Wrote 0 leases to leases file.
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Listening on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 sh[5973]: Listening on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 sh[5973]: Sending on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 sh[5973]: Sending on Socket/fallback/fallback-net
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Sending on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Sending on Socket/fallback/fallback-net
Aug 26 15:02:13 LAB105001 dhcpd[5973]: Server starting service.


isc-dhcp-server.service - ISC DHCP IPv4 server
Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
Active: active (running) since Tue 2025-08-26 15:31:03 IST; 27s ago
Docs: man:dhcpd(8)
Main PID: 6728 (dhcpd)
Tasks: 4 (limit: 8980)
Memory: 4.5M
CPU: 24ms
CGroup: /system.slice/isc-dhcp-server.service
└─6728 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dhcp/dhcpd.con
Aug 26 15:31:03 LAB105001 dhcpd[6728]: PID file: /run/dhcp-server/dhcpd.pid
Aug 26 15:31:03 LAB105001 dhcpd[6728]: Wrote 2 leases to leases file.
Aug 26 15:31:03 LAB105001 sh[6728]: Wrote 2 leases to leases file.
Aug 26 15:31:03 LAB105001 dhcpd[6728]: Listening on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:31:03 LAB105001 sh[6728]: Listening on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:31:03 LAB105001 sh[6728]: Sending on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:31:03 LAB105001 sh[6728]: Sending on Socket/fallback/fallback-net
Aug 26 15:31:03 LAB105001 dhcpd[6728]: Sending on LPF/enp2s0/00:e0:4c:68:2c:48/192.168.10.0/24
Aug 26 15:31:03 LAB105001 dhcpd[6728]: Sending on Socket/fallback/fallback-net
Aug 26 15:31:03 LAB105001 dhcpd[6728]: Server starting service.
nwlab@LAB105001:~$ ping 192.168.10.55
PING 192.168.10.55 (192.168.10.55) 56(84) bytes of data.
64 bytes from 192.168.10.55: icmp_seq=1 ttl=64 time=0.255 ms
64 bytes from 192.168.10.55: icmp_seq=2 ttl=64 time=0.325 ms
64 bytes from 192.168.10.55: icmp_seq=3 ttl=64 time=0.311 ms
64 bytes from 192.168.10.55: icmp_seq=4 ttl=64 time=0.316 ms
64 bytes from 192.168.10.55: icmp_seq=5 ttl=64 time=0.309 ms
64 bytes from 192.168.10.55: icmp_seq=6 ttl=64 time=0.318 ms
64 bytes from 192.168.10.55: icmp_seq=7 ttl=64 time=0.312 ms
64 bytes from 192.168.10.55: icmp_seq=8 ttl=64 time=0.312 ms
64 bytes from 192.168.10.55: icmp_seq=9 ttl=64 time=0.314 ms
64 bytes from 192.168.10.55: icmp_seq=10 ttl=64 time=0.312 ms
64 bytes from 192.168.10.55: icmp_seq=11 ttl=64 time=0.319 ms
64 bytes from 192.168.10.55: icmp_seq=12 ttl=64 time=0.313 ms
64 bytes from 192.168.10.55: icmp_seq=13 ttl=64 time=0.264 ms
64 bytes from 192.168.10.55: icmp_seq=14 ttl=64 time=0.315 ms
64 bytes from 192.168.10.55: icmp_seq=15 ttl=64 time=0.309 ms
64 bytes from 192.168.10.55: icmp_seq=16 ttl=64 time=0.317 ms
64 bytes from 192.168.10.55: icmp_seq=17 ttl=64 time=0.314 ms
64 bytes from 192.168.10.55: icmp_seq=18 ttl=64 time=0.314 ms
64 bytes from 192.168.10.55: icmp_seq=19 ttl=64 time=0.310 ms
64 bytes from 192.168.10.55: icmp_seq=20 ttl=64 time=0.311 ms
64 bytes from 192.168.10.55: icmp_seq=21 ttl=64 time=0.328 ms
64 bytes from 192.168.10.55: icmp_seq=22 ttl=64 time=0.337 ms
64 bytes from 192.168.10.55: icmp_seq=23 ttl=64 time=0.315 ms
^[[3~64 bytes from 192.168.10.55: icmp_seq=24 ttl=64 time=0.311 ms
64 bytes from 192.168.10.55: icmp_seq=25 ttl=64 time=0.301 ms
64 bytes from 192.168.10.55: icmp_seq=26 ttl=64 time=0.303 ms
64 bytes from 192.168.10.55: icmp_seq=27 ttl=64 time=0.307 ms
64 bytes from 192.168.10.55: icmp_seq=28 ttl=64 time=0.320 ms
64 bytes from 192.168.10.55: icmp_seq=29 ttl=64 time=0.396 ms
64 bytes from 192.168.10.55: icmp_seq=30 ttl=64 time=0.342 ms

CLIENT SIDE

1. **Enter the file '/etc/network/interfaces**

   COMMAND: `sudo nano /etc/network/interfaces`

   Explanation: This file defines the network settings for your system. For a DHCP client, you need to configure it to obtain an IP address automatically from a DHCP server

2. **Content to add in conf file**

   ```
   auto enp2s0
   ```

   ```
   iface enp2s0 inet dhcp
   ```

   Explanation: This configuration sets up the enp2s0 interface to use DHCP to obtain an IP address automatically.

3. **Restart Network manager and systemd-networkd services to apply the changes**

   COMMAND: `sudo systemctl restart NetworkManager`

   COMMAND: `sudo systemctl restart systemd-networkd`

4. **Check status of both of above:**

   COMMAND : `sudo systemctl status NetworkManager`

   COMMAND : `sudo systemctl status systemd-networkd`

5. **Manually obtain IP Address**

   For flushing out the already assigned/engaged IP and requesting new IP manually.

   COMMAND : `sudo dhclient -r`

   COMMAND : `sudo dhclient enp2s0`

   Explanation: This command manually requests an IP address

   from the DHCP server for the enp2s0 interface.

6. **Verify the address obtained from DHCP server**

   COMMAND : `ip addr show`

7. **To check the network interface status**

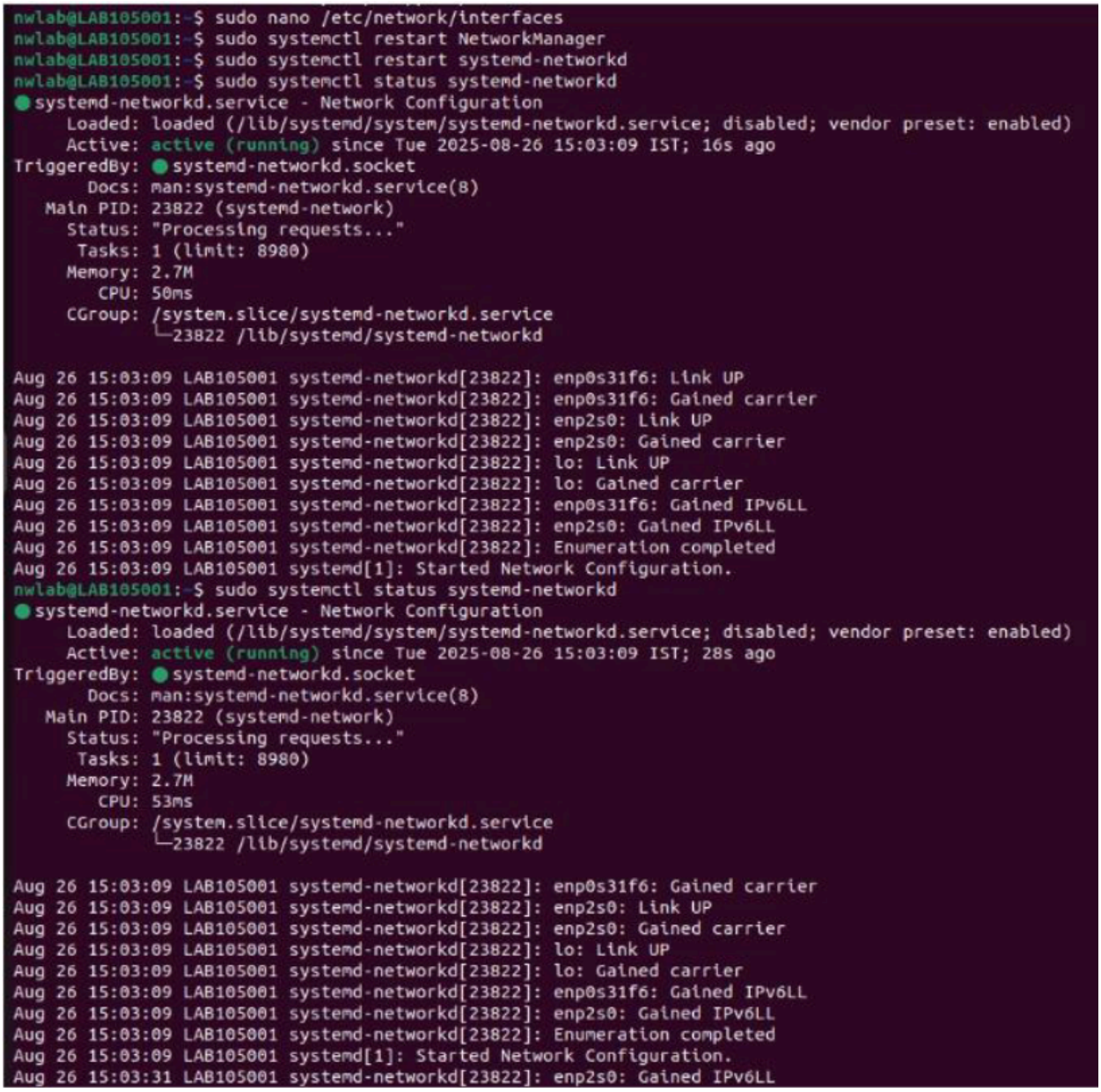
   COMMAND : `ip link show`

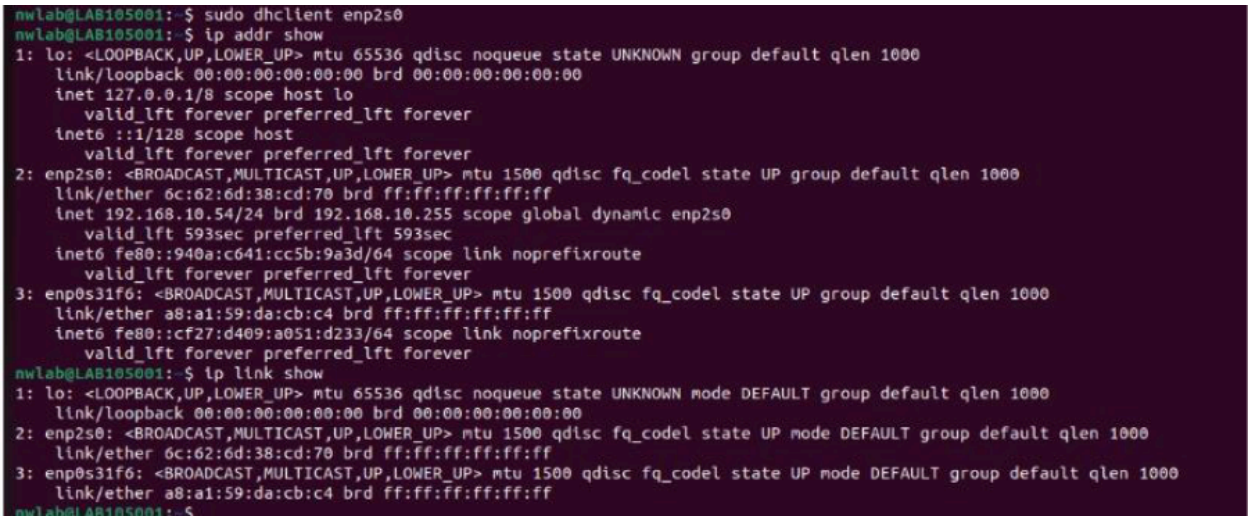
   Explanation: This command shows the status of the network

   interface, which can help determine if the interface is up and running.

   It’s working depends on the network connectivity

OUTPUT:


nwlab@LAB105001:~$ sudo nano /etc/network/interfaces
nwlab@LAB105001:~$ sudo systemctl restart NetworkManager
nwlab@LAB105001:~$ sudo systemctl restart systemd-networkd
nwlab@LAB105001:~$ sudo systemctl status systemd-networkd
● systemd-networkd.service - Network Configuration
Loaded: loaded (/lib/systemd/system/systemd-networkd.service; disabled; vendor preset: enabled)
Active: active (running) since Tue 2025-08-26 15:03:09 IST; 16s ago
TriggeredBy: ● systemd-networkd.socket
Docs: man:systemd-networkd.service(8)
Main PID: 23822 (systemd-network)
Status: "Processing requests..."
Tasks: 1 (limit: 8980)
Memory: 2.7M
CPU: 50ms
CGroup: /system.slice/systemd-networkd.service
└─23822 /lib/systemd/systemd-networkd
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp0s31f6: Link UP
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp0s31f6: Gained carrier
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp2s0: Link UP
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp2s0: Gained carrier
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: lo: Link UP
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: lo: Gained carrier
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp0s31f6: Gained IPv6LL
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp2s0: Gained IPv6LL
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: Enumeration completed
Aug 26 15:03:09 LAB105001 systemd[1]: Started Network Configuration.
nwlab@LAB105001:~$ sudo systemctl status systemd-networkd
● systemd-networkd.service - Network Configuration
Loaded: loaded (/lib/systemd/system/systemd-networkd.service; disabled; vendor preset: enabled)
Active: active (running) since Tue 2025-08-26 15:03:09 IST; 28s ago
TriggeredBy: ● systemd-networkd.socket
Docs: man:systemd-networkd.service(8)
Main PID: 23822 (systemd-network)
Status: "Processing requests..."
Tasks: 1 (limit: 8980)
Memory: 2.7M
CPU: 53ms
CGroup: /system.slice/systemd-networkd.service
└─23822 /lib/systemd/systemd-networkd
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp0s31f6: Gained carrier
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp2s0: Link UP
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp2s0: Gained carrier
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: lo: Link UP
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: lo: Gained carrier
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp0s31f6: Gained IPv6LL
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: enp2s0: Gained IPv6LL
Aug 26 15:03:09 LAB105001 systemd-networkd[23822]: Enumeration completed
Aug 26 15:03:09 LAB105001 systemd[1]: Started Network Configuration.
Aug 26 15:03:31 LAB105001 systemd-networkd[23822]: enp2s0: Gained IPv6LL


nwlab@LAB105001:~$ sudo dhclient enp2s0
nwlab@LAB105001:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
link/ether 6c:62:6d:38:cd:70 brd ff:ff:ff:ff:ff:ff
inet 192.168.10.54/24 brd 192.168.10.255 scope global dynamic enp2s0
valid_lft 593sec preferred_lft 593sec
inet6 fe80::940a:c641:cc5b:9a3d/64 scope link noprefixroute
valid_lft forever preferred_lft forever
3: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
link/ether a8:a1:59:da:cb:c4 brd ff:ff:ff:ff:ff:ff
inet6 fe80::cf27:d409:a051:d233/64 scope link noprefixroute
valid_lft forever preferred_lft forever
nwlab@LAB105001:~$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group default qlen 1000
link/ether 6c:62:6d:38:cd:70 brd ff:ff:ff:ff:ff:ff
3: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group default qlen 1000
link/ether a8:a1:59:da:cb:c4 brd ff:ff:ff:ff:ff:ff

QUESTIONS:

1) **What is the difference between static IP address and dynamic**
   - Static IP: Manually assigned, fixed, does not change.
   - Dynamic IP: Automatically assigned by DHCP server, changes after lease time.

2) **What do you mean by lease time of dynamically assigned IP address? How its value is governed?**
   - Lease time is the duration for which a DHCP server assigns an IP to a device and it is governed by DHCP server configuration (can be minutes, hours, or **days)**

3) **How can one bind an IP address to a MAC address?**
   - Using DHCP reservation (also called static binding), we can configure the DHCP server so that whenever a device with a specific MAC address connects, it always gets the same IP address.