

Name: Harshal Jaywant Chavan

Slip no: 31

Roll no: 202124

Class: FYMCA-A

Practical 1:

Default Network Topology (Point-to-Point)

Code:

```
#include "ns3/core-module.h"
```

```
#include "ns3/network-module.h"
```

```
#include "ns3/internet-module.h"
```

```
#include "ns3/point-to-point-module.h"
```

```
#include "ns3/applications-module.h"
```

```
#include "ns3/netanim-module.h"
```

```
#include "ns3/mobility-module.h"
```

```
using namespace ns3;
```

```
NS_LOG_COMPONENT_DEFINE ("FirstScriptExample");
```

```
int
```

```
main (int argc, char *argv[])
```

```
{
```

```
    CommandLine cmd;
```

```
    cmd.Parse (argc, argv);
```

```
    Time::SetResolution (Time::NS);
```

```
    LogComponentEnable ("UdpEchoClientApplication", LOG_LEVEL_INFO);
```

```
    LogComponentEnable ("UdpEchoServerApplication", LOG_LEVEL_INFO);
```

```
NodeContainer nodes;
```

```
nodes.Create (2);
```

```
PointToPointHelper pointToPoint;
```

```
pointToPoint.SetDeviceAttribute ("DataRate", StringValue ("5Mbps"));
```

```
pointToPoint.SetChannelAttribute ("Delay", StringValue ("2ms"));
```

```
NetDeviceContainer devices;
```

```
devices = pointToPoint.Install (nodes);
```

```
InternetStackHelper stack;
```

```
stack.Install (nodes);
```

```
Ipv4AddressHelper address;
```

```
address.SetBase ("10.1.1.0", "255.255.255.0");
```

```
Ipv4InterfaceContainer interfaces = address.Assign (devices);
```

```
UdpEchoServerHelper echoServer (9);
```

```
ApplicationContainer serverApps = echoServer.Install (nodes.Get (1));
```

```
serverApps.Start (Seconds (1.0));
```

```
serverApps.Stop (Seconds (10.0));
```

```
UdpEchoClientHelper echoClient (interfaces.GetAddress (1), 9);
```

```
echoClient.SetAttribute ("MaxPackets", UIntegerValue (1));
```

```
echoClient.SetAttribute ("Interval", TimeValue (Seconds (1.0)));
```

```
echoClient.SetAttribute ("PacketSize", UIntegerValue (1024));
```

```
ApplicationContainer clientApps = echoClient.Install (nodes.Get (0));
```

```
clientApps.Start (Seconds (2.0));
```

```
clientApps.Stop (Seconds (10.0));
```

```
//For Net Anim
```

```
MobilityHelper mobility;
```

```
mobility.SetMobilityModel("ns3::ConstantPositionMobilityModel");
```

```
mobility.Install(nodes);
```

```
AnimationInterface anim("prac1.xml");
```

```
AnimationInterface::SetConstantPosition (nodes.Get(0), 10, 25);
```

```
AnimationInterface::SetConstantPosition(nodes.Get(1), 20,25);
```

```
anim.EnablePacketMetadata(true);
```

```
pointToPoint.EnablePcapAll("prac1");
```

```
Simulator::Run ();
```

```
Simulator::Destroy ();
```

```
return 0;
```

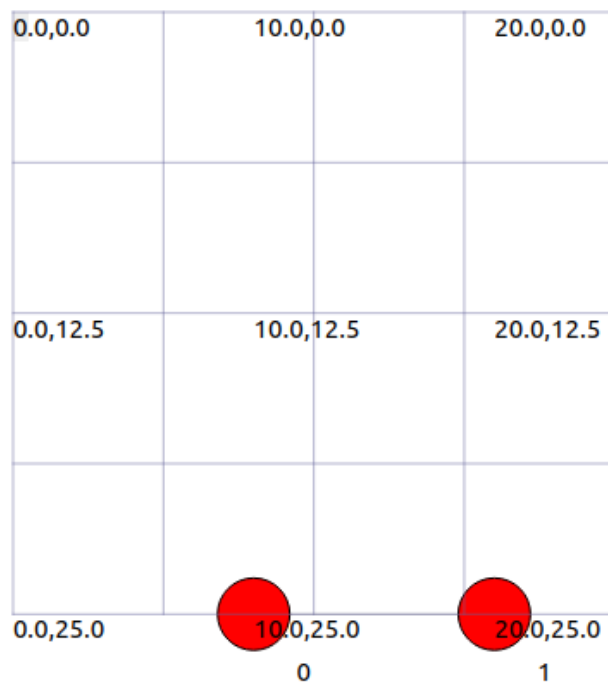
```
}
```

Output:

Run project

```
harshal-chavan@harshal-chavan:~/ns-allinone-3.29/ns-3.29$ ./waf --run prac1
Waf: Entering directory `/home/harshal-chavan/ns-allinone-3.29/ns-3.29/build'
[1791/1843] Compiling scratch/prac1.cc
[1822/1843] Linking build/scratch/prac1
Waf: Leaving directory `/home/harshal-chavan/ns-allinone-3.29/ns-3.29/build'
Build commands will be stored in build/compile_commands.json
'build' finished successfully (12.592s)
At time 2s client sent 1024 bytes to 10.1.1.2 port 9
At time 2.00369s server received 1024 bytes from 10.1.1.1 port 49153
At time 2.00369s server sent 1024 bytes to 10.1.1.1 port 49153
At time 2.00737s client received 1024 bytes from 10.1.1.2 port 9
harshal-chavan@harshal-chavan:~/ns-allinone-3.29/ns-3.29$
```

Xml file view in NetAnim



Export table

#

From Time

2

To Time

2.00569

Show Nodes

0:1

Show Graph

☐ Tcp

☐ Udp

☐ Ipv4

☐ Ipv6

☐ Icmp

☐ Wifi

☐ Ethernet

☐ Ppp

☐ Aodv

☐ Olsr

☐ Arp

Regex on meta data

.*

Submit

0

1

UDP 49153 > 9

Export Table

	From Id	To Id	Tx	Meta
1	0	1	2	UDP 49153 > 9
2	1	0	2.00369	UDP 9 > 49153

Wireshark file

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.1.1.1	10.1.1.2	UDP	1054	49153 → 9 Len=1024
2	0.000000	10.1.1.2	10.1.1.1	UDP	1054	9 → 49153 Len=1024

Frame 1: 1054 bytes on wire (8432 bits), 1054 bytes captured (8432 bits)

Point-to-Point Protocol

Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2

User Datagram Protocol, Src Port: 49153, Dst Port: 9

Data (1024 bytes)

0000 00 21 45 00 04 1c 00 00 00 00 40 11 00 00 0a 01 ..!E.....@.....

0010 01 01 0a 01 01 02 c0 01 00 09 04 08 00 00 00 00@.....

0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

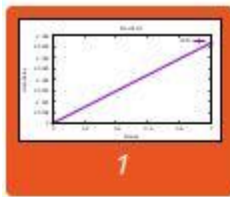
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

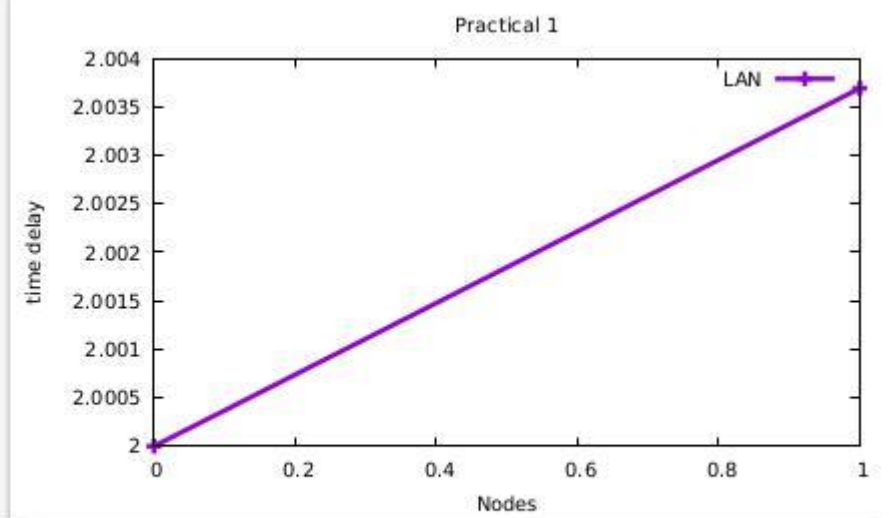
0080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

0090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00@.....

Graph



1



Practical 2:

Program to simulate Hybrid Topology (Point-to-Point + Bus Topology)

Code:

```
#include "ns3/core-module.h"
#include "ns3/network-module.h"
#include "ns3/csma-module.h"
#include "ns3/internet-module.h"
#include "ns3/point-to-point-module.h"
#include "ns3/applications-module.h"
#include "ns3/ipv4-global-routing-helper.h"
#include "ns3/netanim-module.h"
#include "ns3/mobility-module.h"
```

```
// Default Network Topology
//
// 10.1.1.0
// n0 ----- n1  n2  n3  n4
// point-to-point |  |  |  |
//                =====
//                LAN 10.1.2.0
```

```
using namespace ns3;
```

```
NS_LOG_COMPONENT_DEFINE ("Mini-Project");
```

```
int
```

```
main (int argc, char *argv[])
```

```
{
```

```
    bool verbose = true;
```

```
uint32_t nCsma = 3;
```

```
CommandLine cmd;
```

```
cmd.AddValue ("nCsma", "Number of \"extra\" CSMA nodes/devices", nCsma);
```

```
cmd.AddValue ("verbose", "Tell echo applications to log if true", verbose);
```

```
cmd.Parse (argc,argv);
```

```
if (verbose)
```

```
{
```

```
    LogComponentEnable ("UdpEchoClientApplication", LOG_LEVEL_INFO);
```

```
    LogComponentEnable ("UdpEchoServerApplication", LOG_LEVEL_INFO);
```

```
}
```

```
nCsma = nCsma == 0 ? 1 : nCsma;
```

```
NodeContainer p2pNodes;
```

```
p2pNodes.Create (2);
```

```
NodeContainer csmaNodes;
```

```
csmaNodes.Add (p2pNodes.Get (1));
```

```
csmaNodes.Create (nCsma);
```

```
PointToPointHelper pointToPoint;
```

```
pointToPoint.SetDeviceAttribute ("DataRate", StringValue ("5Mbps"));
```

```
pointToPoint.SetChannelAttribute ("Delay", StringValue ("2ms"));
```

```
NetDeviceContainer p2pDevices;
```

```
p2pDevices = pointToPoint.Install (p2pNodes);
```

```
CsmaHelper csma;
```



```
csma.SetChannelAttribute ("DataRate", StringValue ("100Mbps"));
csma.SetChannelAttribute ("Delay", TimeValue (NanoSeconds (6560)));
```

```
NetDeviceContainer csmaDevices;
csmaDevices = csma.Install (csmaNodes);
```

```
InternetStackHelper stack;
stack.Install (p2pNodes.Get (0));
stack.Install (csmaNodes);
```

```
Ipv4AddressHelper address;
address.SetBase ("10.1.1.0", "255.255.255.0");
Ipv4InterfaceContainer p2pInterfaces;
p2pInterfaces = address.Assign (p2pDevices);
```

```
address.SetBase ("10.1.2.0", "255.255.255.0");
Ipv4InterfaceContainer csmaInterfaces;
csmaInterfaces = address.Assign (csmaDevices);
```

```
UdpEchoServerHelper echoServer (9);
```

```
ApplicationContainer serverApps = echoServer.Install (csmaNodes.Get (nCsmas));
serverApps.Start (Seconds (1.0));
serverApps.Stop (Seconds (10.0));
```

```
UdpEchoClientHelper echoClient (csmaInterfaces.GetAddress (nCsmas), 9);
echoClient.SetAttribute ("MaxPackets", UIntegerValue (1));
echoClient.SetAttribute ("Interval", TimeValue (Seconds (1.0)));
echoClient.SetAttribute ("PacketSize", UIntegerValue (1024));
```

```
ApplicationContainer clientApps = echoClient.Install (p2pNodes.Get (0));
```

```
clientApps.Start (Seconds (2.0));
```

```
clientApps.Stop (Seconds (10.0));
```

```
Ipv4GlobalRoutingHelper::PopulateRoutingTables ();
```

```
pointToPoint.EnablePcapAll ("mini-project-hybrid-topology");
```

```
csma.EnablePcap ("mini-project-hybrid-topology", csmaDevices.Get (1), true);
```

```
//For Net Anim
```

```
MobilityHelper mobility;
```

```
mobility.SetMobilityModel("ns3::ConstantPositionMobilityModel");
```

```
mobility.Install(p2pNodes);
```

```
mobility.Install(csmaNodes);
```

```
AnimationInterface anim("mini-project-hybrid-topology.xml");
```

```
AnimationInterface::SetConstantPosition (p2pNodes.Get(0), 10, 25);
```

```
AnimationInterface::SetConstantPosition(p2pNodes.Get(1), 20,25);
```

```
AnimationInterface::SetConstantPosition(csmaNodes.Get(1),40,25);
```

```
AnimationInterface::SetConstantPosition(csmaNodes.Get(2),50,25);
```

```
AnimationInterface::SetConstantPosition(csmaNodes.Get(3),60,25);
```

```
anim.EnablePacketMetadata(true);
```

```
Simulator::Run ();
```

```
Simulator::Destroy ();
```

```
return 0;
```

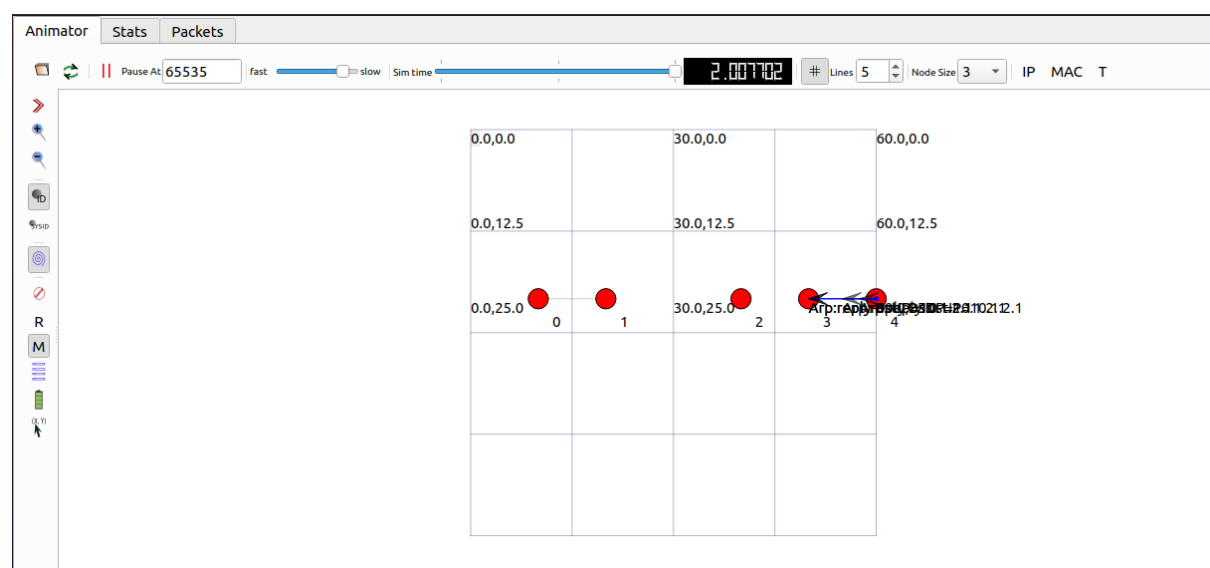
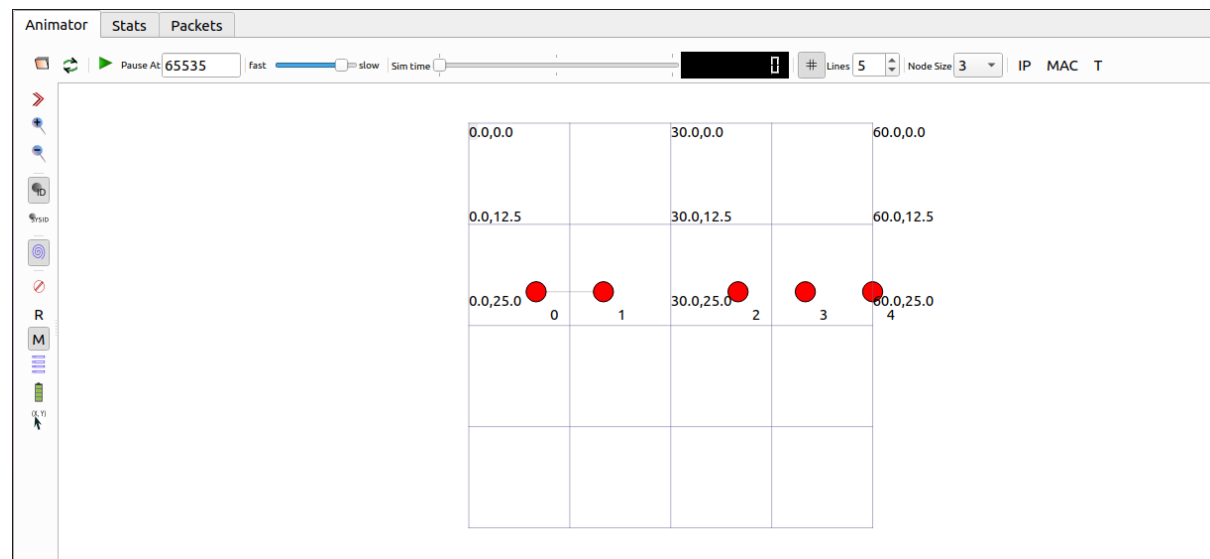
```
}
```

Output:

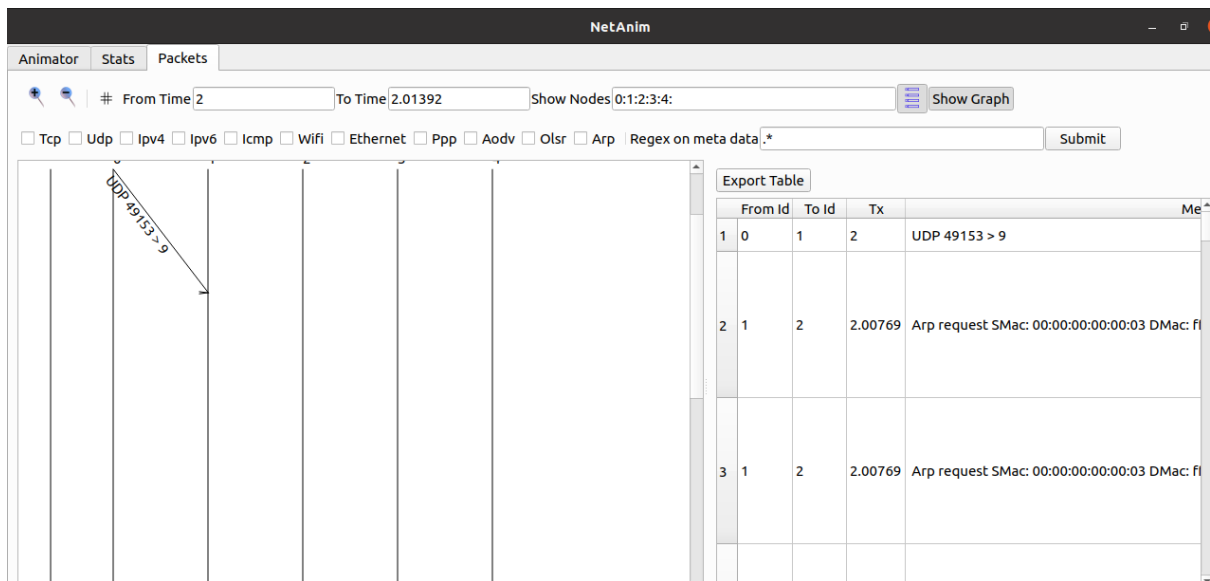
Run project

```
harshal-chavan@harshal-chavan:~/ns-allinone-3.29/ns-3.29$ ./waf --run mini-project
Waf: Entering directory `/home/harshal-chavan/ns-allinone-3.29/ns-3.29/build'
[1788/1841] Compiling scratch/mini-project.cc
[1801/1841] Linking build/scratch/mini-project
Waf: Leaving directory `/home/harshal-chavan/ns-allinone-3.29/ns-3.29/build'
Build commands will be stored in build/compile_commands.json
'build' finished successfully (2m44.709s)
At time 2s client sent 1024 bytes to 10.1.2.4 port 9
At time 2.0078s server received 1024 bytes from 10.1.1.1 port 49153
At time 2.0078s server sent 1024 bytes to 10.1.1.1 port 49153
At time 2.01761s client received 1024 bytes from 10.1.2.4 port 9
harshal-chavan@harshal-chavan:~/ns-allinone-3.29/ns-3.29$
```

Xml file view in NetAnim



Export table



Exported data.txt file

```

1 From Id      To Id  Tx      Meta
2 0            1      2      UDP 49153 > 9
3 1            2      2.00769  ARP request SMac: 00:00:00:00:00:03 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.1 DstIp : 10.1.2.4
4 1            2      2.00769  ARP request SMac: 00:00:00:00:00:03 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.1 DstIp : 10.1.2.4
5 1            3      2.00769  ARP request SMac: 00:00:00:00:00:03 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.1 DstIp : 10.1.2.4
6 1            3      2.00769  ARP request SMac: 00:00:00:00:00:03 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.1 DstIp : 10.1.2.4
7 1            4      2.00769  ARP request SMac: 00:00:00:00:00:03 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.1 DstIp : 10.1.2.4
8 1            4      2.00769  ARP request SMac: 00:00:00:00:00:03 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.1 DstIp : 10.1.2.4
9 4            1      2.00777  ARP reply SMac: 00:00:00:00:00:06 DMac: 00:00:00:00:00:03 SrcIp : 10.1.2.4 DstIp : 10.1.2.1
10 4           1      2.00777  ARP reply SMac: 00:00:00:00:00:06 DMac: 00:00:00:00:00:03 SrcIp : 10.1.2.4 DstIp : 10.1.2.1
11 4           2      2.00777  ARP reply SMac: 00:00:00:00:00:06 DMac: 00:00:00:00:00:03 SrcIp : 10.1.2.4 DstIp : 10.1.2.1
12 4           3      2.00777  ARP reply SMac: 00:00:00:00:00:06 DMac: 00:00:00:00:00:03 SrcIp : 10.1.2.4 DstIp : 10.1.2.1
13 1           2      2.00771  UDP 49153 > 9
14 1           3      2.00771  UDP 49153 > 9
15 1           4      2.00771  UDP 49153 > 9
16 1           4      2.00771  UDP 49153 > 9
17 4           1      2.0138  ARP request SMac: 00:00:00:00:00:06 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.4 DstIp : 10.1.2.1
18 4           1      2.0138  ARP request SMac: 00:00:00:00:00:06 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.4 DstIp : 10.1.2.1
19 4           2      2.0138  ARP request SMac: 00:00:00:00:00:06 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.4 DstIp : 10.1.2.1
20 4           2      2.0138  ARP request SMac: 00:00:00:00:00:06 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.4 DstIp : 10.1.2.1
21 4           3      2.0138  ARP request SMac: 00:00:00:00:00:06 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.4 DstIp : 10.1.2.1
22 4           3      2.0138  ARP request SMac: 00:00:00:00:00:06 DMac: ff:ff:ff:ff:ff:ff SrcIp : 10.1.2.4 DstIp : 10.1.2.1
23 1           2      2.01382  ARP reply SMac: 00:00:00:00:00:03 DMac: 00:00:00:00:00:06 SrcIp : 10.1.2.1 DstIp : 10.1.2.4
24 1           3      2.01382  ARP reply SMac: 00:00:00:00:00:03 DMac: 00:00:00:00:00:06 SrcIp : 10.1.2.1 DstIp : 10.1.2.4
25 1           4      2.01382  ARP reply SMac: 00:00:00:00:00:03 DMac: 00:00:00:00:00:06 SrcIp : 10.1.2.1 DstIp : 10.1.2.4
26 1           4      2.01382  ARP reply SMac: 00:00:00:00:00:03 DMac: 00:00:00:00:00:06 SrcIp : 10.1.2.1 DstIp : 10.1.2.4

```

.plt file

```

1 set terminal pdf
2 set output "mini-project.pdf"
3 set title "P2P and Bus Topology"
4 set xlabel "Nodes"
5 set ylabel "time delay"
6 plot "data.txt" using 1:3 with linespoint title "LAN" lw 4

```

Wireshark file

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	00:00:00_00:00:03	Broadcast	ARP	64	who has 10.1.2.4? Tell 10.1.2.1
2	0.000012	00:00:00_00:00:06	00:00:00_00:00:03	ARP	64	10.1.2.4 is at 00:00:00:00:00:06
3	0.000105	10.1.1.1	10.1.2.4	UDP	1070	49153 → 9 Len=1024
4	0.000117	00:00:00_00:00:06	Broadcast	ARP	64	who has 10.1.2.1? Tell 10.1.2.4
5	0.000130	00:00:00_00:00:03	00:00:00_00:00:06	ARP	64	10.1.2.1 is at 00:00:00:00:00:03
6	0.000223	10.1.2.4	10.1.1.1	UDP	1070	9 → 49153 Len=1024

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
 Ethernet II, Src: 00:00:00_00:00:03 (00:00:00:00:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 Address Resolution Protocol (request)

```

0000  ff ff ff ff ff 00 00 00 00 03 08 06 00 01  .....
0010  08 00 06 04 00 01 00 00 00 00 03 0a 01 02 01  .....
0020  ff ff ff ff ff 0a 01 02 04 00 00 00 00 00 00  .....
0030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
  
```

Graph:

