औद्योगिक प्रशिक्षण के लिए राष्ट्रीय संस्थान

National Institute for Industrial Training

One Premier Organization with Non Profit Status | Registered Under Govt. of WB
Empanelled Under Planning Commission Govt. of India
Inspired By: National Task Force on IT & SD Government of India

THE PROJECT REPORT SUBMITTED TO

NATIONAL INSTITUTE FOR INDUSTRIAL TRAINING

KOLKATA

WEST BENGAL

SUMMER TRAINING – 2016

NAME: TANMOY PURKAIT

COLLEGE: GOVT. COLLEGE OF ENGINEERING AND CERAMIC TECHNOLOGY

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING

DATE OF SUBMISSION: 20.07.2016

PROJECTS ON ADVANCED NETWORKING WITH ROUTING AND SWITCHING

THE FOLLOWING PROJECTS WERE CONFIGURED ON CISCO PACKET TRACER STUDENT V 6.2

1) CONFIGURING ROUTING PROTOCOLS WITH EIGRP
2) CONFIGURING ROUTING PROTOCOLS WITH OSPF
3) CONFIGURING TELEPHONY SERVICE

INDEX

ACKNOWLEDGEMENT

The projects on “Advanced Networking with Routing and Switching” were prepared by Tanmoy Purkait under the sincere supervision of **Mr. SAYANTAN CHAKROBORTY**, Technical Director, Chairman (Non-Profit Organisation). I am using this opportunity to express my deepest gratitude and special thanks to him for guiding us on the correct path, motivating us and helping us complete our training together in merriment.

INTRODUCTION

A network consists of two or more computers that are linked in order to share resources, exchange files, or allow electronic communications. The computers on a network may be linked through cables, telephone lines, radio waves, satellites, or infrared light beams.

Networking helps us study and analyse the communication process among various computing devices or computer systems that are linked, or networked, together to exchange information and share resources.

When looking at networking basics, understanding the way a network operates is the first step to understanding routing and switching. The network operates by connecting computers and peripherals using two pieces of equipment; switches and routers. Switches and routers, essential networking basics, enable the devices that are connected to your network to communicate with each other, as well as with other networks.

Though they look quite similar, routers and switches perform very different functions in a network.

Switches are used to connect multiple devices on the same network within a building or campus. For example, a switch can connect your computers, printers and servers, creating a network of shared resources. The switch, one aspect of your networking basics, would serve as a controller, allowing the various devices to share information and talk to each other. Through information sharing and resource allocation, switches save you money and increase productivity.

Routers, the second valuable component of your networking basics, are used to tie multiple networks together. For example, you would use a router to connect your networked computers to the Internet and thereby share an Internet connection among many users. The router will act as a dispatcher, choosing the best route for your information to travel so that you receive it quickly.

Routers analyse the data being sent over a network, change how it is packaged, and send it to another network, or over a different type of network. They connect your business to the outside world, protect your information from security threats, and can even decide which computers get priority over others.

Cisco Packet Tracer is a powerful network simulation program that allows students to experiment with network behaviour and ask “what if” questions. Packet Tracer provides simulation, visualization, authoring, assessment, and collaboration capabilities and facilitates the teaching and learning of complex technology concepts.

Packet Tracer supplements physical equipment in the classroom by allowing students to create a network with an almost unlimited number of devices, encouraging practice, discovery, and troubleshooting. The simulation-based learning environment helps students develop 21st century skills such as decision making, creative and critical thinking, and problem solving. Packet Tracer allows instructors to easily teach and demonstrate complex technical concepts and networking systems design.

CONFIGURING ROUTING PROTOCOLS WITH EIGRP

EIGRP stands for Enhanced Interior Gateway Routing Protocol. It is a Cisco-proprietary routing protocol for TCP/IP. It is based on IGRP routing protocol. It has several enhancements to scale the enterprise size network.

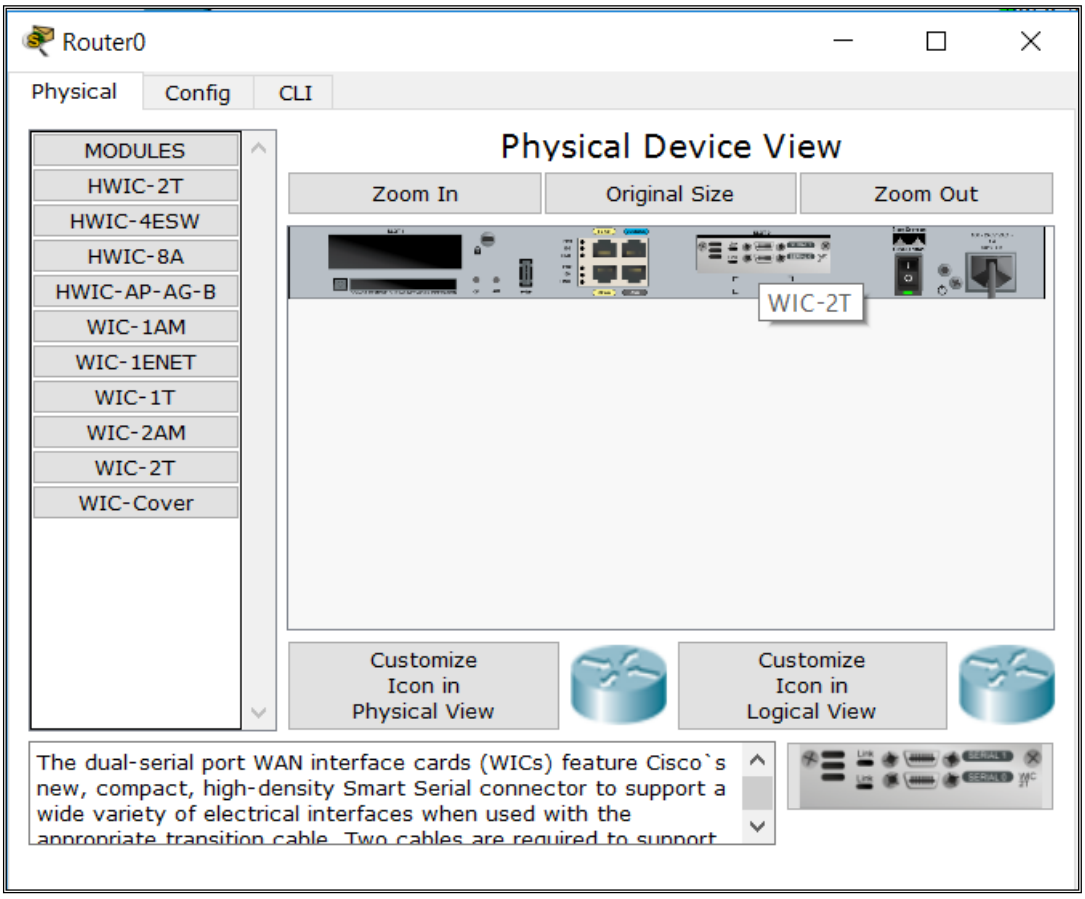
A topology as demonstrated in the picture is created using Cisco Packet Tracer Student v6.2.

Eight CISCO 1841 Routers are connected together across its serial interfaces via Serial DCE and DTE cables. Each router has a CISCO 2950-24 switch attached to it via fast Ethernet interfaces with straight-through cables. Generic devices (PCs) are further connected to the switches with straight-through cables.

Configuring each Router:

1. Double click on each Router
2. Under the Physical tab, we find the back side of the router where we need to power off the router and add a WAN Interface Card (WIC) and then turn the router back on. In this case, we use WIC-2T for the purpose.

WIC-2T: The dual-serial port WAN interface cards (WICs) feature Cisco's new, compact, high-density Smart Serial connector to support a wide variety of electrical interfaces when used with the appropriate transition cable. Two cables are required to support the two ports on the WIC. Each port on a WIC is a different physical interface and can support different protocols such as Point-to-Point protocol (PPP) or Frame Relay and Data Terminal Equipment/Data Communications Equipment (DTE/DCE).

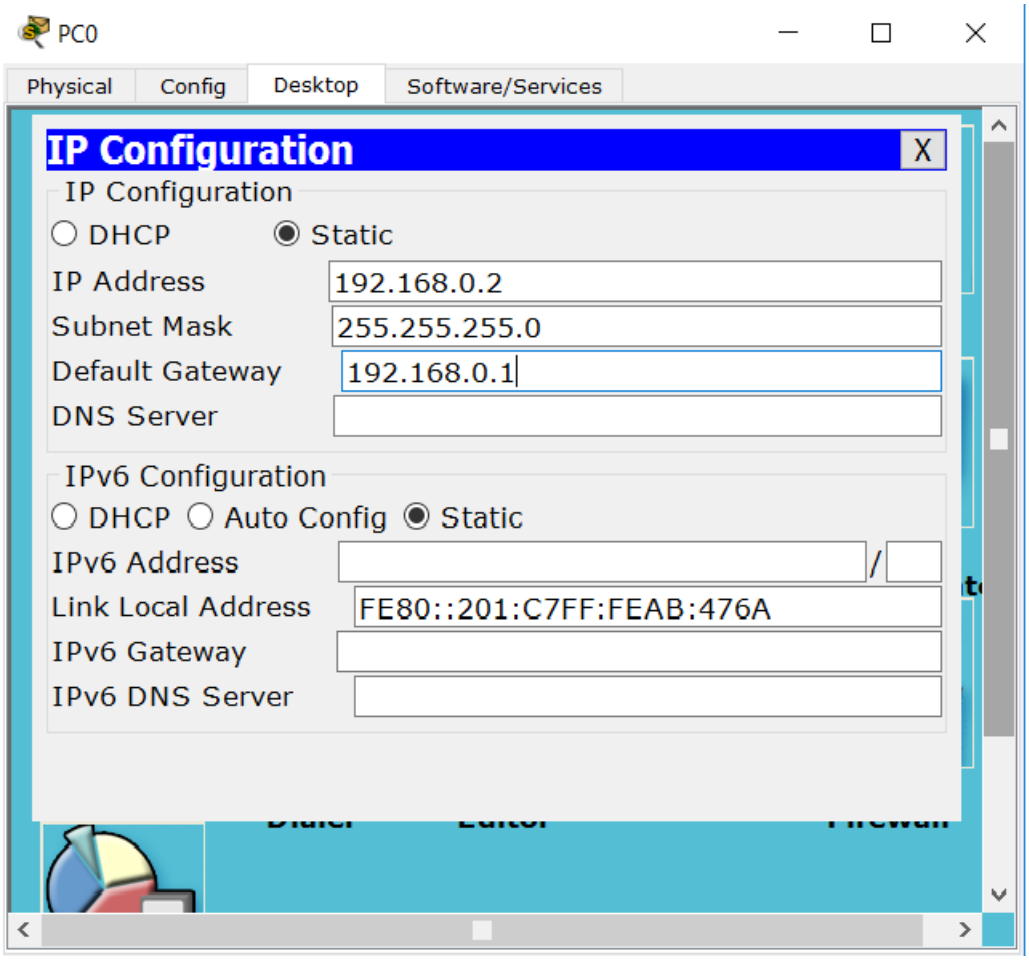


Assigning IP addresses to PCs:

1. Double click each PC
2. Under the Desktop tab, click on IP Configuration.
3. Assign IP address and Default Gateway. Leave the Subnet Mask as default.

The IP address for each PC is mentioned beside each PC in the topology diagram. The Default Gateway for each PC is the IP address that is assigned to the Fast- Ethernet interface of the router to which it is connected via a switch. It is also mentioned in the topology diagram.

This is to be done for each PC on all the networks.



Assigning IP addresses to interfaces of each router:

1. Double click Router 0.
2. We will configure it under the CLI tab.

Three interfaces ***FastEthernet0/0, Serial0/0/0*** and ***Serial0/0/1*** of **Router0** are used in this topology. By default, interfaces on router remain administratively down during the start up.

We need to configure IP address and other parameters on interfaces before we could actually use them for routing. Interface mode is used to assign the IP address and other parameters. Interface mode can be accessed from global configuration mode. Following commands are used to access the global configuration mode.

Router 0:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
```

From global configuration mode we can enter in interface mode. From there we can configure the interface. Following commands will assign IP address on *FastEthernet0/0.*

```
Router(config)#int fa0/0
Router(config-if)#ip address 192.168.0.1 255.255.255.0
Router(config-if)#no shut
Router(config-if)#exit
```

int fa*0/0* command is used to enter in interface mode.

ip address *192.168.0.1 255.255.255.0* command will assign IP address to interface.

no shut command will bring the interface up.

exit command is used to return in global configuration mode

Serial interface needs an additional parameter called **clock rate**. Every serial cable has two ends DTE and DCE. This parameter is always configured at DCE end.

```
Router(config)#int s0/0/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#int s0/0/1
Router(config-if)#ip address 80.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit
```

int *s0/0/0* Command is used to enter in interface mode.

ip address 10.0.0.1 255.0.0.0 Command assigns IP address to interface.

clock rate *64000* In real life environment this parameter controls the data flow between serial links and need to be set at service provider's end. In lab environment we need not to worry about this value. We can use any valid rate here.

no shut Command brings interface up.

exit Command is used to return in global configuration mode.

We will use same commands to assign IP addresses on interfaces of remaining routers. We need to provide clock rate only on DCE side of serial interface.

Router 1:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 191.168.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)# int se0/0/0
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 2:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 190.169.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 20.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#int se0/0/1
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 3:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 193.165.0.1 255.255.255.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 30.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 40.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 4:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 172.16.0.1 255.255.0.0
Router(config-if)#no shut

Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 40.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 50.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 5:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 172.168.0.1 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 50.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 60.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut

Router(config-if)#exit
```

Router 6:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 190.168.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 60.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 70.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 7:

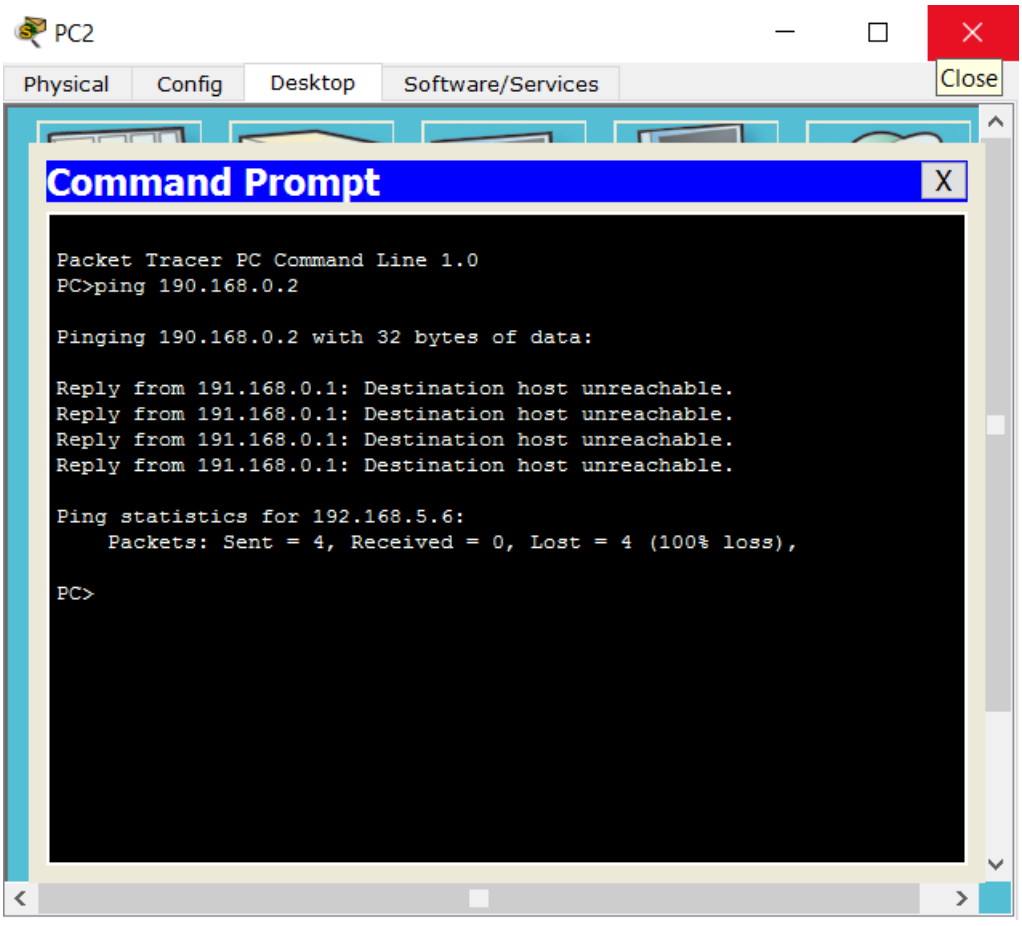
```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 191.169.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 70.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 80.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Now routers have information about the networks that they have on their own interfaces. Routers will not exchange this information between them on their own.

We can check it by pinging PC 10 from PC 2 by typing **ping 190.168.0.2** in the command prompt terminal on PC 2.



We need to implement EIGRP routing protocol that will insist them to share this information.

Configure EIGRP routing protocol:

Enabling EIGRP is a two-step process: -

1. Enable EIGRP routing protocol from global configuration mode.
2. Tell EIGRP which interfaces we want to include.

For these steps following commands are used respectively.

```
Router(config)# router eigrp autonomous_system_#
Router(config-router)# network IP_network_# [subnet_mask]
```

Router(config)# router eigrp autonomous_system_#

This command will enable EIGRP routing protocol in router. We can use any ASN (Autonomous System Number) from 1 to 65,535. In order to become EIGRP neighbours this number must be same on all participates.

Router(config-router)# network IP_network_# [subnet_mask]

This command allows us to specify the local interfaces which we want to include in EIGRP. Basically we define a range of addresses and router search for these addresses in local interfaces. If match found EIGRP will be enabled on that interface. Once enabled, EIGRP will start advertising about the connected subnets with that interface.

EIGRP configuration:

Let's implement the commands in our network.

Router 0:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 192.168.0.0 255.255.255.0
Router(config-router)#network 80.0.0.0 255.0.0.0
Router(config-router)#network 10.0.0.0 255.0.0.0
Router(config-router)#exit
```

Router 1:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 191.168.0.0 255.255.0.0
Router(config-router)#network 10.0.0.0 255.0.0.0
Router(config-router)#network 20.0.0.0 255.0.0.0
Router(config-router)#exit
```

Router 2:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 190.169.0.0 255.255.0.0
Router(config-router)#network 20.0.0.0 255.0.0.0
Router(config-router)#network 30.0.0.0 255.0.0.0
Router(config-router)#exit
```

Router 3:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 193.165.0.0 255.255.255.0
Router(config-router)#network 30.0.0.0 255.0.0.0
Router(config-router)#network 40.0.0.0 255.0.0.0
Router(config-router)#exit
```

Router 4:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 172.16.0.0 255.255.0.0
Router(config-router)#network 40.0.0.0 255.0.0.0
Router(config-router)#network 50.0.0.0 255.0.0.0
Router(config-router)#exit
```

Router 5:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 172.168.0.0 255.0.0.0
Router(config-router)#network 50.0.0.0 255.0.0.0
Router(config-router)#network 60.0.0.0 255.0.0.0
Router(config-router)#exit
```

Router 6:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 190.168.0.0 255.255.0.0
Router(config-router)#network 60.0.0.0 255.0.0.0
Router(config-router)#network 70.0.0.0 255.0.0.0
Router(config-router)#exit
```

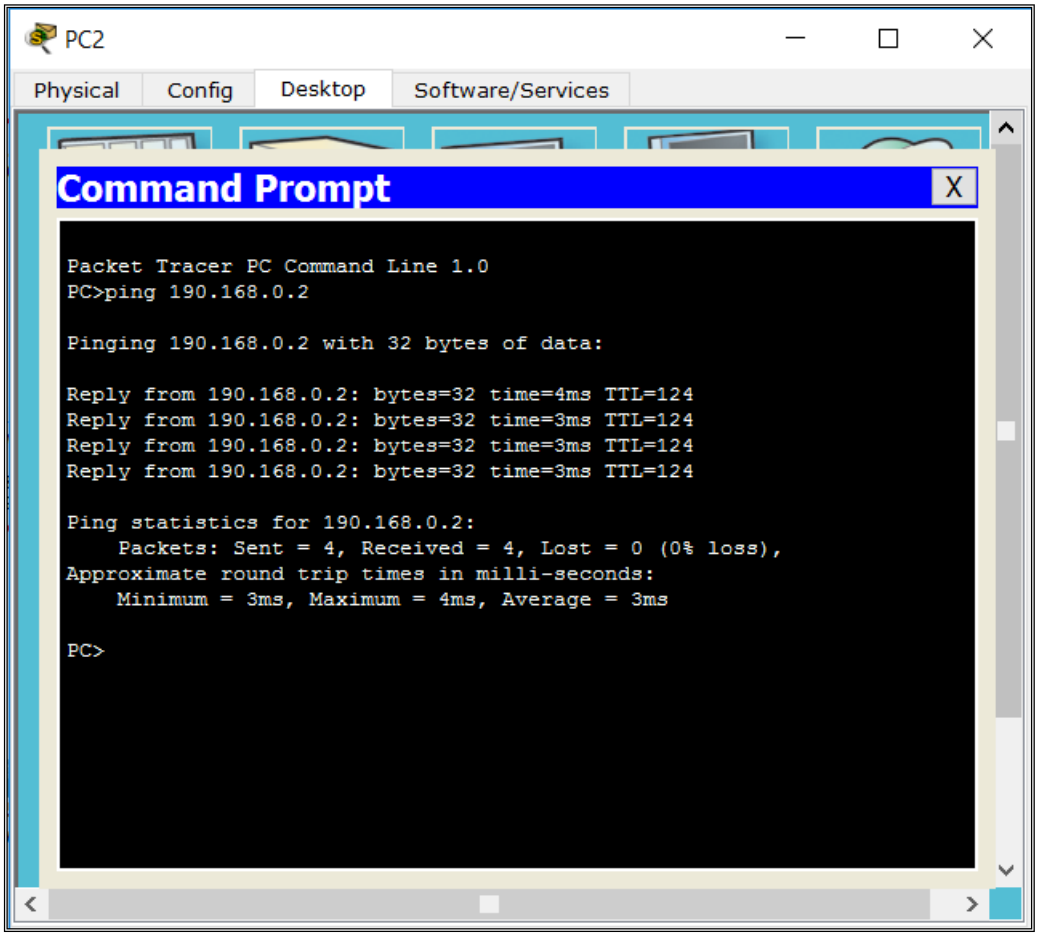
Router 7:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router eigrp 7
Router(config-router)#network 191.169.0.0 255.255.0.0
Router(config-router)#network 70.0.0.0 255.0.0.0
Router(config-router)#network 80.0.0.0 255.0.0.0
Router(config-router)#exit
```

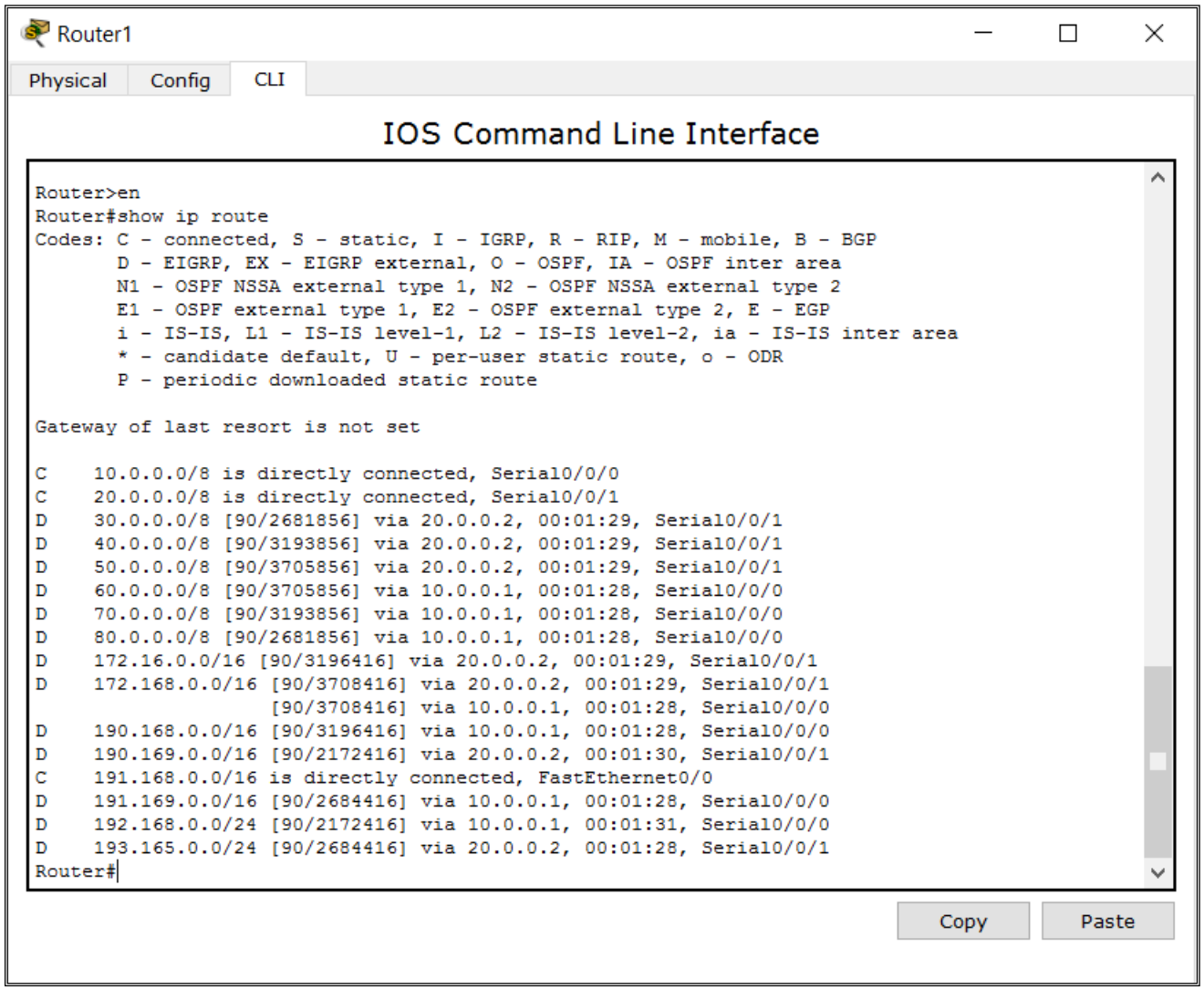
That's it. Our network is ready to take the advantage of EIGRP routing. To verify the setup, we will use ping command. **ping** command is used to test the connectivity between two devices.

Access the command prompt of PC2 and use ping command to test the connectivity from PC10.

ping 190.168.0.2



We can check the routing table by giving this command. For instance, we can check the routing table by **show ip route command** in Privileged mode.



CONFIGURING ROUTING PROTOCOLS WITH OSPF

OSPF stands for Open Shortest Path First. OSPF is a link state open standard based routing protocol. It was created in mid-1980. Since it is based on open standard, we can use it with any vendor's router.

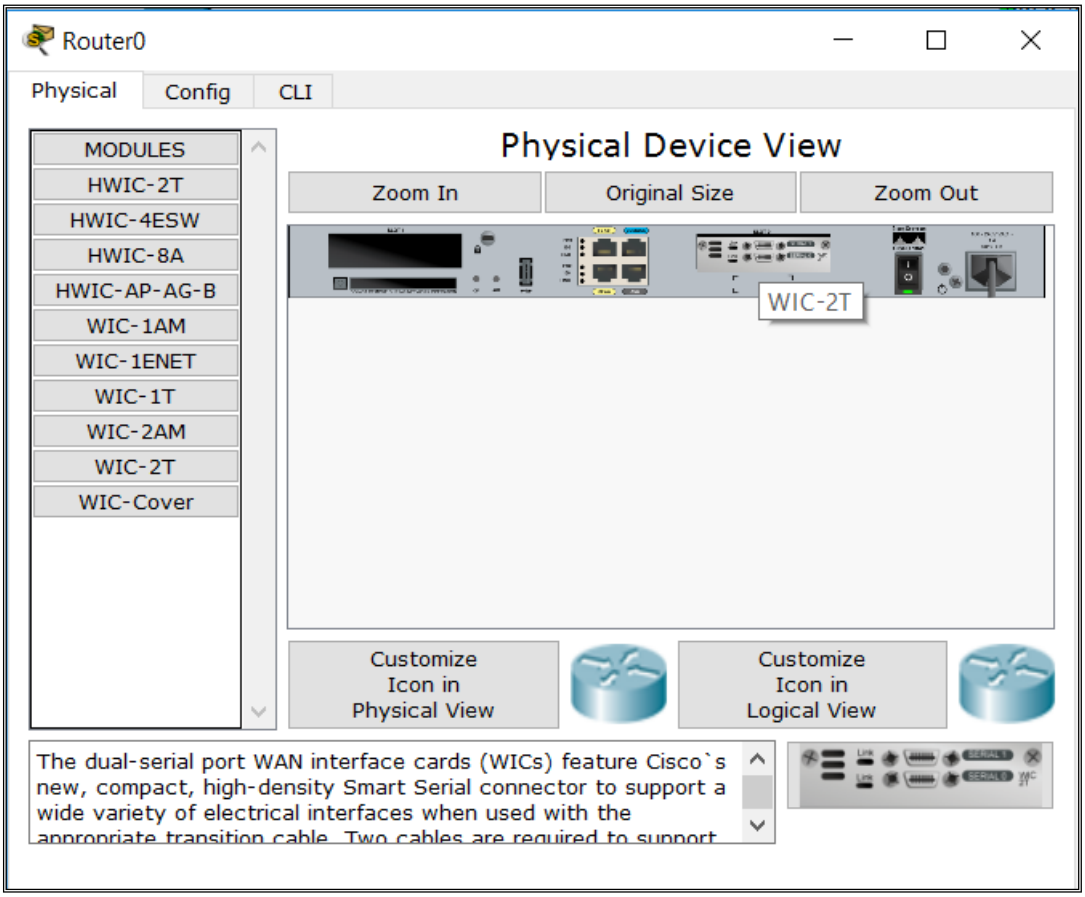
A topology as demonstrated in the picture is created using Cisco Packet Tracer Student v6.2.

Eight CISCO 1841 Routers are connected together across its serial interfaces via Serial DCE and DTE cables. Each router has a CISCO 2950-24 switch attached to it via fast Ethernet interfaces with straight-through cables. Generic devices (PCs) are further connected to the switches with straight-through cables.

Configuring each Router:

3. Double click on each Router
4. Under the Physical tab, we find the back side of the router where we need to power off the router and add a WAN Interface Card (WIC) and then turn the router back on. In this case, we use WIC-2T for the purpose.

WIC-2T: The dual-serial port WAN interface cards (WICs) feature Cisco's new, compact, high-density Smart Serial connector to support a wide variety of electrical interfaces when used with the appropriate transition cable. Two cables are required to support the two ports on the WIC. Each port on a WIC is a different physical interface and can support different protocols such as Point-to-Point protocol (PPP) or Frame Relay and Data Terminal Equipment/Data Communications Equipment (DTE/DCE).

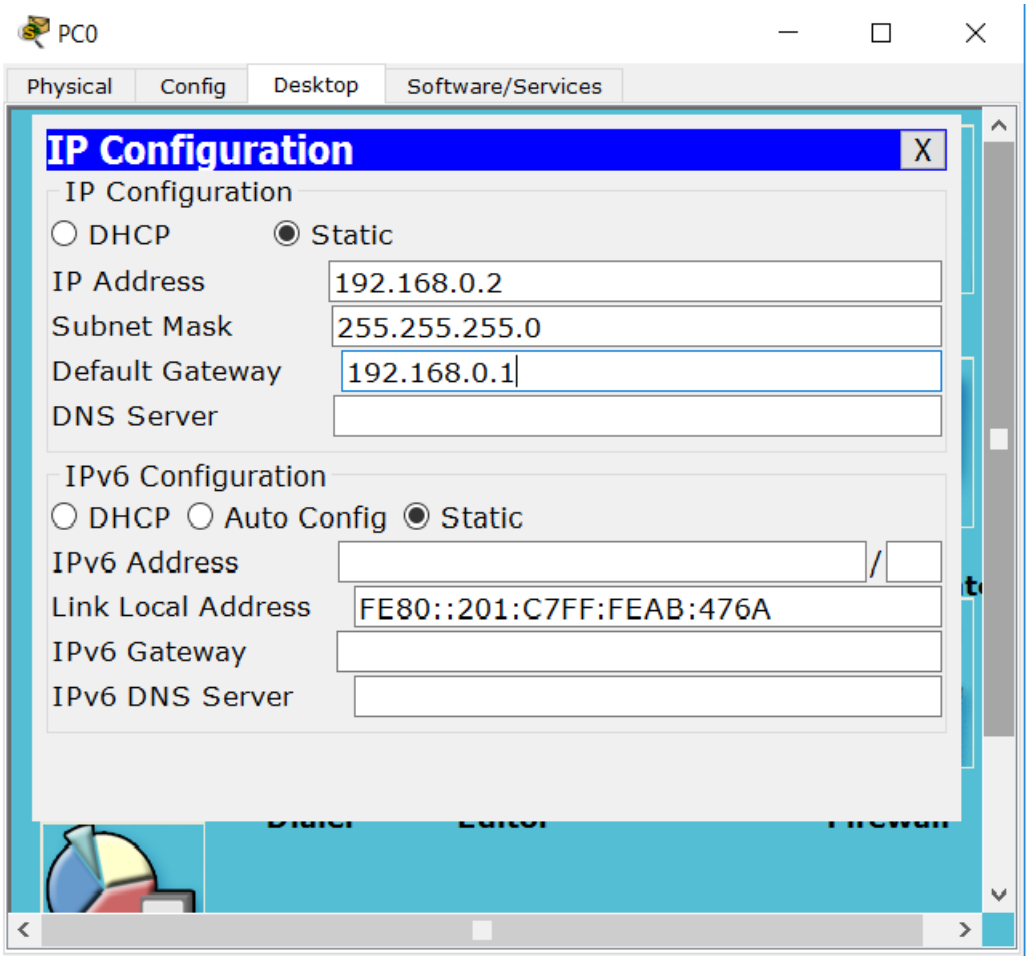


Assigning IP addresses to PCs:

4. Double click each PC
5. Under the Desktop tab, click on IP Configuration.
6. Assign IP address and Default Gateway. Leave the Subnet Mask as default.

The IP address for each PC is mentioned beside each PC in the topology diagram. The Default Gateway for each PC is the IP address that is assigned to the Fast- Ethernet interface of the router to which it is connected via a switch. It is also mentioned in the topology diagram.

This is to be done for each PC on all the networks.



Assigning IP addresses to interfaces of each router:

3. Double click Router 0.
4. We will configure it under the CLI tab.

Three interfaces ***FastEthernet0/0, Serial0/0/0*** and ***Serial0/0/1*** of **Router0** are used in this topology. By default, interfaces on router remain administratively down during the start up.

We need to configure IP address and other parameters on interfaces before we could actually use them for routing. Interface mode is used to assign the IP address and other parameters. Interface mode can be accessed from global configuration mode. Following commands are used to access the global configuration mode.

Router 0:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
```

From global configuration mode we can enter in interface mode. From there we can configure the interface. Following commands will assign IP address on *FastEthernet0/0.*

```
Router(config)#int fa0/0
Router(config-if)#ip address 192.168.0.1 255.255.255.0
Router(config-if)#no shut
Router(config-if)#exit
```

int fa*0/0* command is used to enter in interface mode.

ip address *192.168.0.1 255.255.255.0* command will assign IP address to interface.

no shut command will bring the interface up.

exit command is used to return in global configuration mode

Serial interface needs an additional parameter called **clock rate**. Every serial cable has two ends DTE and DCE. This parameter is always configured at DCE end.

```
Router(config)#int s0/0/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#int s0/0/1
Router(config-if)#ip address 80.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit
```

int *s0/0/0* Command is used to enter in interface mode.

ip address 10.0.0.1 255.0.0.0 Command assigns IP address to interface.

clock rate *64000* In real life environment this parameter controls the data flow between serial links and need to be set at service provider's end. In lab environment we need not to worry about this value. We can use any valid rate here.

no shut Command brings interface up.

exit Command is used to return in global configuration mode.

We will use same commands to assign IP addresses on interfaces of remaining routers. We need to provide clock rate only on DCE side of serial interface.

Router 1:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 191.168.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)# int se0/0/0
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 2:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 190.169.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 20.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#int se0/0/1
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 3:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 193.165.0.1 255.255.255.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 30.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 40.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 4:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 172.16.0.1 255.255.0.0
Router(config-if)#no shut

Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 40.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 50.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 5:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 172.168.0.1 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 50.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 60.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut

Router(config-if)#exit
```

Router 6:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 190.168.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 60.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 70.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Router 7:

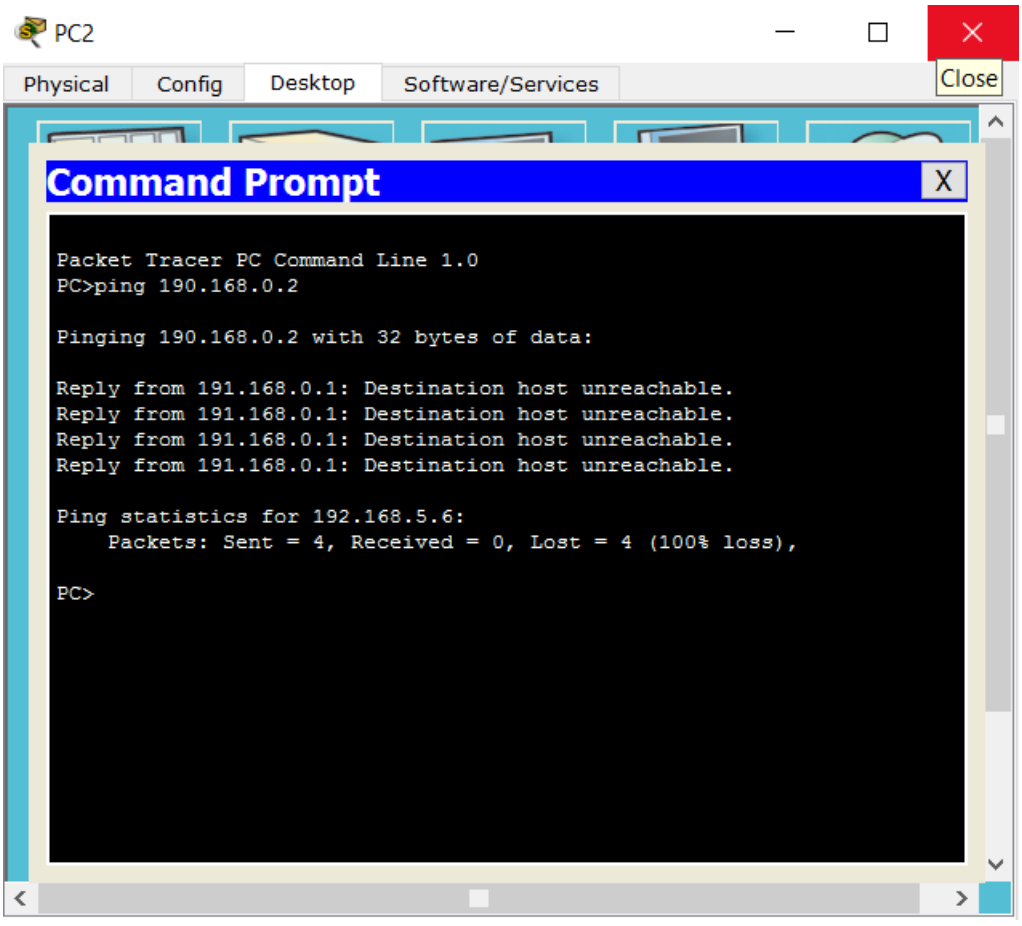
```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip address 191.169.0.1 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/0
Router(config-if)#ip address 70.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int se0/0/1
Router(config-if)#ip address 80.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shut
Router(config-if)#exit
```

Now routers have information about the networks that they have on their own interfaces. Routers will not exchange this information between them on their own.

We can check it by pinging PC 10 from PC 2 by typing **ping 190.168.0.2** in the command prompt terminal on PC 2.



We need to implement OSPF routing protocol that will insist them to share this information.

Configure OSPF routing protocol:

Enabling OSPF is a two-step process: -

3. Enable OSPF routing protocol from global configuration mode.
4. Tell OSPF which interfaces we want to include.

For these steps following commands are used respectively.

```
Router(config)# router ospf process_ID

Router(config-router)# network IP_network_# [wild card mask] area [area number]
```

Router(config)# router ospf process ID

This command will enable OSPF routing protocol in router. Process ID is a positive integer. We can use any number from 1 to 65,535. Process ID is locally significant. We can run multiple OSPF process on same router. Process ID is used to differentiate between them. Process ID need not to match on all routers.

Router(config-router)# network IP_network_# [wildcard_mask] area [area number]

Network command allows us to specify the interfaces which we want to include in OSPF process. This command accepts three arguments network number, wildcard mask and area number.

OSPF configuration:

Let's implement the commands in our network.

Router 0:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 192.168.0.0 0.0.0.255 area 0
Router(config-router)#network 80.0.0.0 0.255.255.255 area 0
Router(config-router)#network 10.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

Router 1:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 191.168.0.0 0.0.255.255 area 0
Router(config-router)#network 10.0.0.0 0.255.255.255 area 0
Router(config-router)#network 20.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

Router 2:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 190.169.0.0 0.0.255.255 area 0
Router(config-router)#network 20.0.0.0 0.255.255.255 area 0
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

Router 3:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 193.165.0.0 0.0.0.255 area 0
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#network 40.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

Router 4:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 172.16.0.0 0.0.255.255 area 0
Router(config-router)#network 40.0.0.0 0.255.255.255 area 0
Router(config-router)#network 50.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

Router 5:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 172.168.0.0 0.255.255.255 area 0
Router(config-router)#network 50.0.0.0 0.255.255.255 area 0
Router(config-router)#network 60.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

Router 6:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 190.168.0.0 0.0.255.255 area 0
Router(config-router)#network 60.0.0.0 0.255.255.255 area 0
Router(config-router)#network 70.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

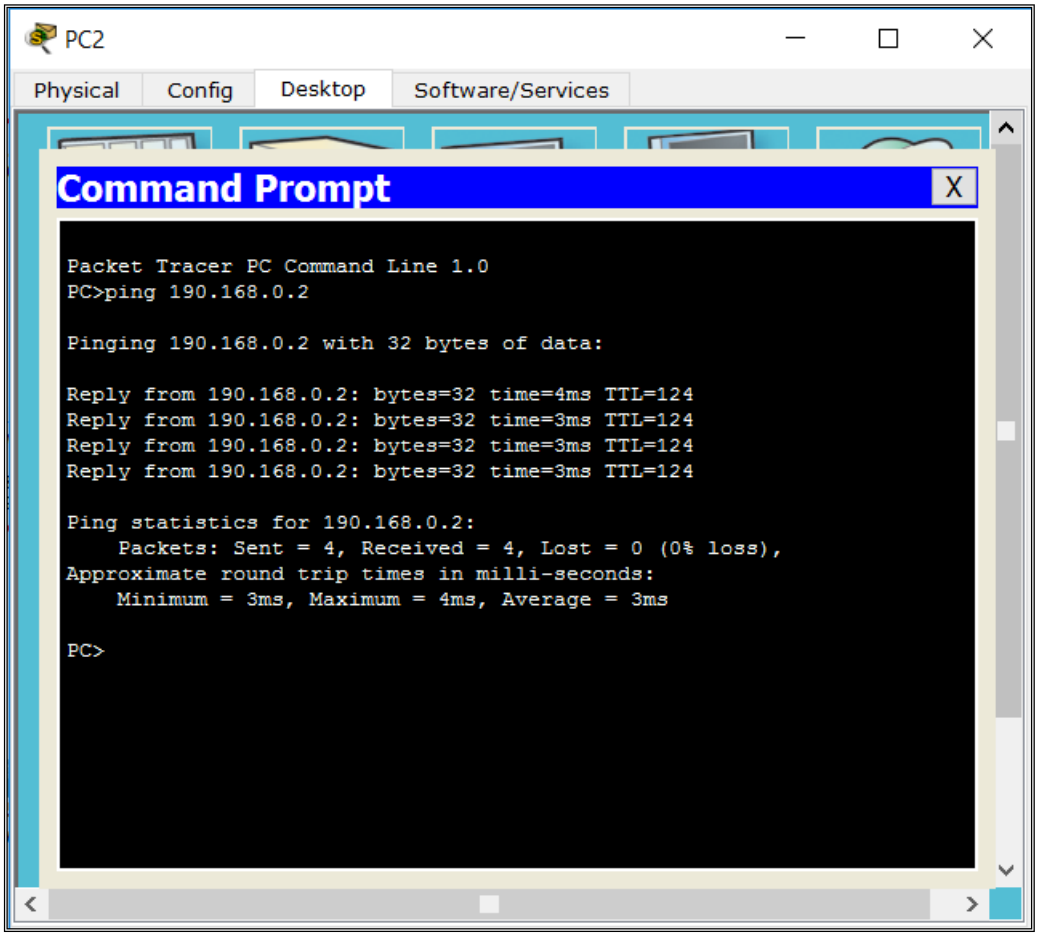
Router 7:

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 191.169.0.0 0.0.255.255 area 0
Router(config-router)#network 70.0.0.0 0.255.255.255 area 0
Router(config-router)#network 80.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
```

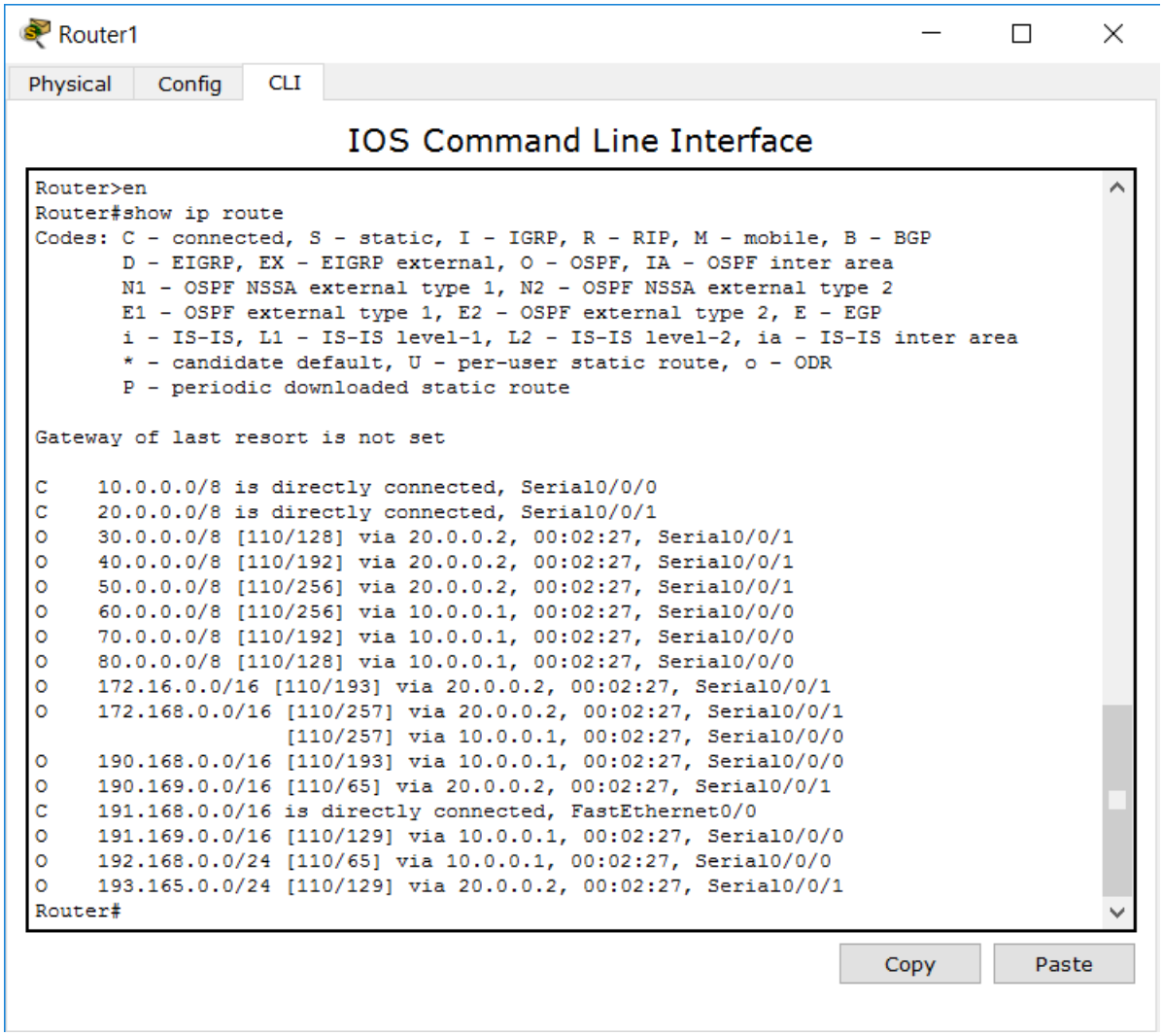
That's it. Our network is ready to take the advantage of OSPF routing. To verify the setup, we will use ping command. **ping** command is used to test the connectivity between two devices.

Access the command prompt of PC2 and use ping command to test the connectivity from PC10.

ping 190.168.0.2

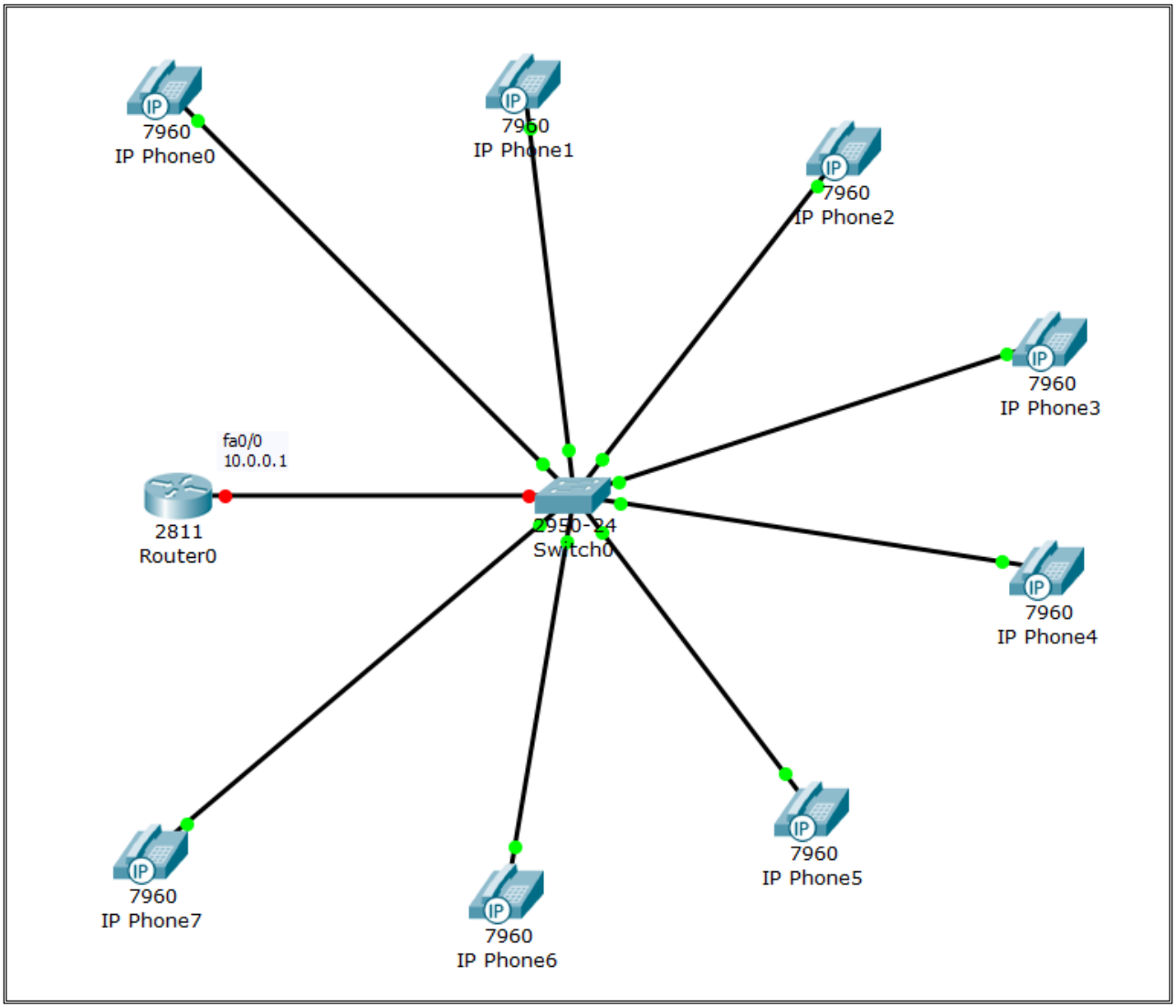


We can check the routing table by giving this command. For instance, we can check the routing table by **show ip route command** in Privileged mode.



CONFIGURING TELEPHONY SERVICE

A topology as demonstrated in the picture is created using Cisco Packet Tracer Student v6.2.

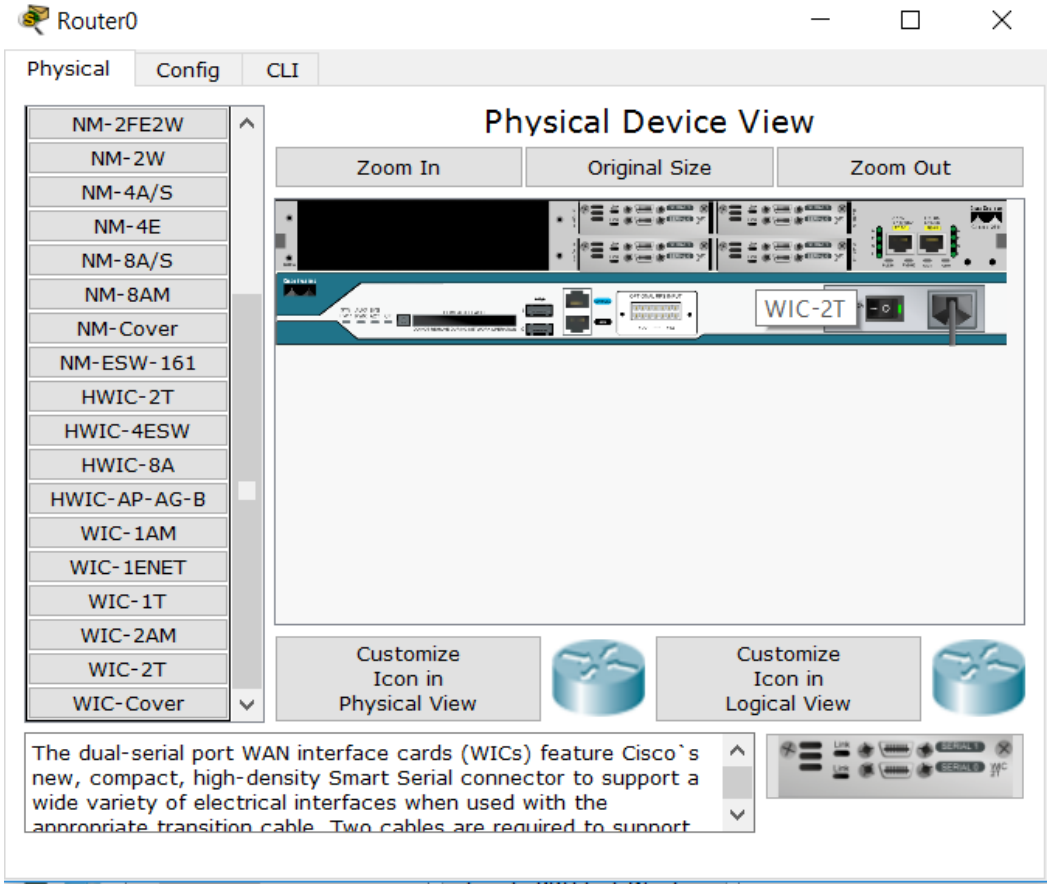


A CISCO 2811 router has a CISCO 2950-24 switch attached to it via fast Ethernet interface with straight-through cable. Eight IP Phones are further connected to the switch with straight-through cables across its other fast-ethernet interfaces.

Configuring the Router:

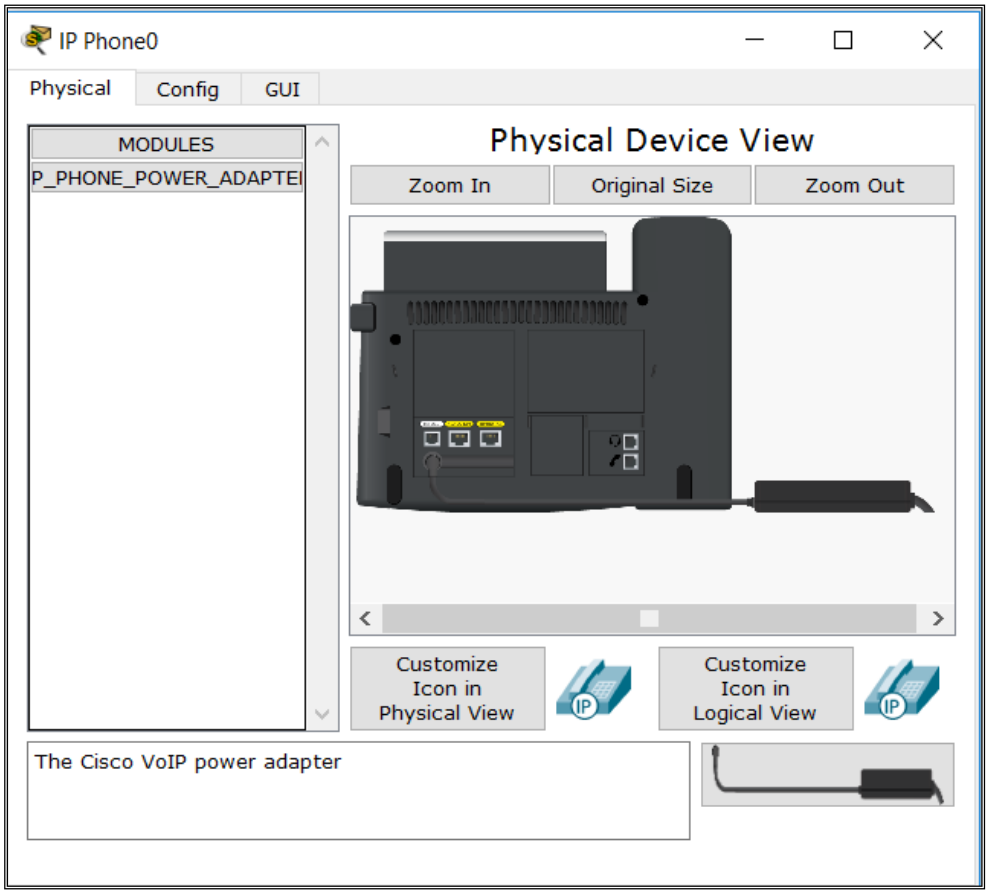
5. Click on the Router
6. Under the Physical tab, we find the back side of the router where we need to power off the router and add 4 Wan Interface Cards (WICs) and then turn the router back on. In this case, we use four WIC-2T for the purpose.

WIC-2T: The dual-serial port WAN interface cards (WICs) feature Cisco`s new, compact, high-density Smart Serial connector to support a wide variety of electrical interfaces when used with the appropriate transition cable. Two cables are required to support the two ports on the WIC. Each port on a WIC is a different physical interface and can support different protocols such as Point-to-Point protocol (PPP) or Frame Relay and Data Terminal Equipment/Data Communications Equipment (DTE/DCE).



Connecting Power Adapter to each IP Phone:

1. Click on each IP Phone
2. Under the Physical tab, connect the power adapter to the back port of the IP Phone.



Assigning IP address to fast Ethernet interface of the router:

5. Double click Router 0.
6. We will configure it under the CLI tab.

One interface ***FastEthernet0/0*** of the Router is used in this topology. By default, interfaces on router remain administratively down during the start up.

We need to configure IP address and other parameters on interfaces before we could actually use them for routing. Interface mode is used to assign the IP address and other parameters. Interface mode can be accessed from global configuration mode. Following commands are used to access the global configuration mode.

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
```

From global configuration mode we can enter in interface mode. From there we can configure the interface. Following commands will assign IP address on *FastEthernet0/0.*

```
Router(config)#int fa0/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#exit
```

int fa*0/0* command is used to enter in interface mode.

ip address *10.0.0.1 255.0.0.0* command will assign IP address to interface.

exit command is used to return in global configuration mode

Configuring DHCP server:

A DHCP server is used to assign IP addresses to the IP Phones.

A TFTP server is used to allow the phones to get the firmware and certain configurations files from the router, this is done by the option 150 of the DHCP.

```
Router(config)#ip dhcp pool VOICE
Router(dhcp-config)#network 10.0.0.0 255.0.0.0
Router(dhcp-config)#default-router 10.0.0.1
Router(dhcp-config)#option 150 ip 10.0.0.1
Router(dhcp-config)#exit
```

Router(config)#ip dhcp pool VOICE: Giving a name to the DHCP pool, in this case, the "VOICE" pool is for the phones only

Router(dhcp-config)#network 10.0.0.0 255.0.0.0: The network that is covered by the DHCP

Router(dhcp-config)#default-router 10.0.0.1: The default router for this IP distribution

Router(dhcp-config)#option 150 ip 10.0.0.1: Configuration of the TFTP

Router(dhcp-config)#exit: command is used to return in global configuration mode.

Configuring the Call Manager Express itself on the router in CLI:

```
Router(config)#telephony-service
Router(config-telephony)#max-ephones 8
Router(config-telephony)#max-dn 8
Router(config-telephony)#ip source-address 10.0.0.1 port 2000
Router(config-telephony)#auto assign 1 to 8
Router(config-telephony)#exit
```

Router(config)#telephony-service: Enters to the telephony services

Router(config-telephony)#max-ephones 8: max number of electronic phones

Router(config-telephony)#max-dn 8: max number of phone lines or directory numbers

Router(config-telephony)#ip source-address 10.0.0.1 port 2000: The IP address of the router where the telephones will be registered and the source address where the DHCP and TFTP services are running, which will be the router itself. And the port used for the phones, the default one is 2000.

Router(config-telephony)#auto assign 1 to 8: to automatically register the ephones, in this case it is from phone 1 to 8

Router(config-telephony)#exit: command is used to return in global configuration mode.

Configuring the Switch:

We will configure a VLAN for voice traffic. As we configured 8 max IP Phones, we will assign 8 interfaces of the switch to the Voice VLAN.

The first interface (fa0/1) of the switch is connected to the router, so we will exclude it on this VLAN configuration.

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int range fa0/2-9
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport voice vlan 1
Switch(config-if-range)#exit
```

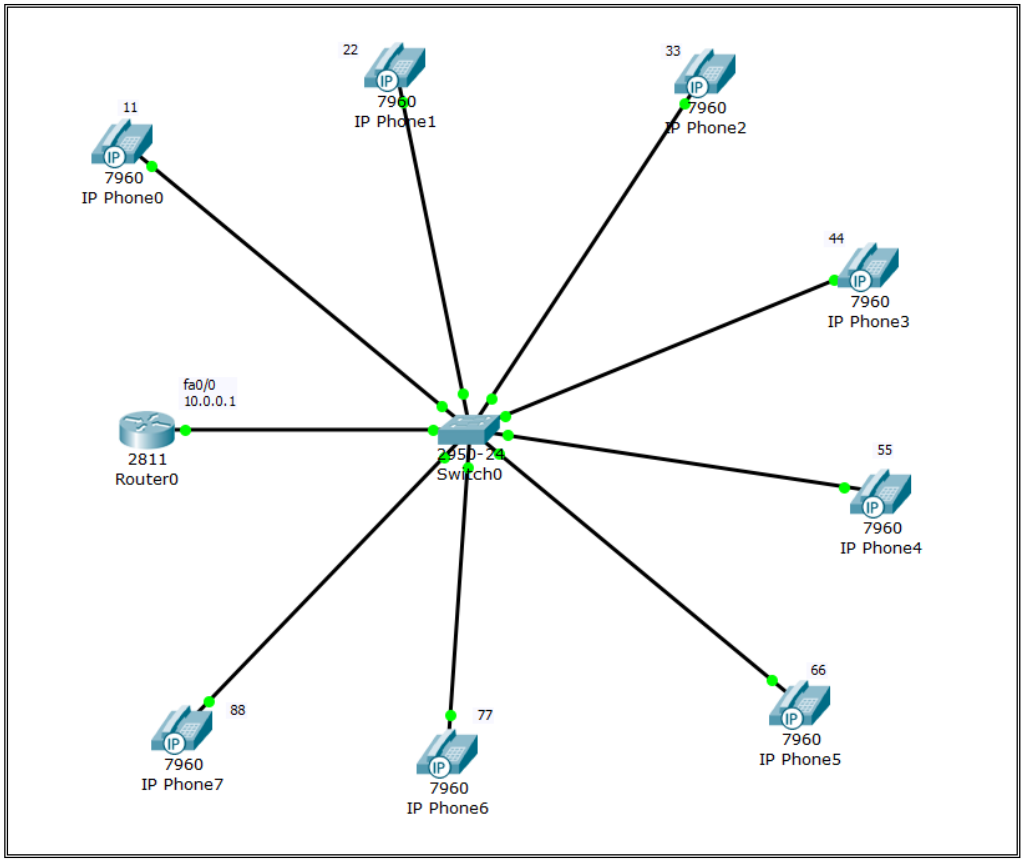
Now is time to give numbers to the lines that will be used on the IP Phones:

We have configured 8 max IP Phones, so we will configure 8 max numbers for the lines.

In the CLI of the Router we will give the following commands:

```
Router(config)#ephone-dn 1
Router(config-ephone-dn)#number 11
Router(config-ephone-dn)#ephone-dn 2
Router(config-ephone-dn)#number 22
Router(config-ephone-dn)#ephone-dn 3
Router(config-ephone-dn)#number 33
Router(config-ephone-dn)#ephone-dn 4
Router(config-ephone-dn)#number 44
Router(config-ephone-dn)#ephone-dn 5
Router(config-ephone-dn)#number 55
Router(config-ephone-dn)#ephone-dn 6
Router(config-ephone-dn)#number 66
Router(config-ephone-dn)#ephone-dn 7
Router(config-ephone-dn)#number 77
Router(config-ephone-dn)#ephone-dn 8
Router(config-ephone-dn)#number 88
Router(config-ephone-dn)#int fa0/0
Router(config-if)#no shut
```

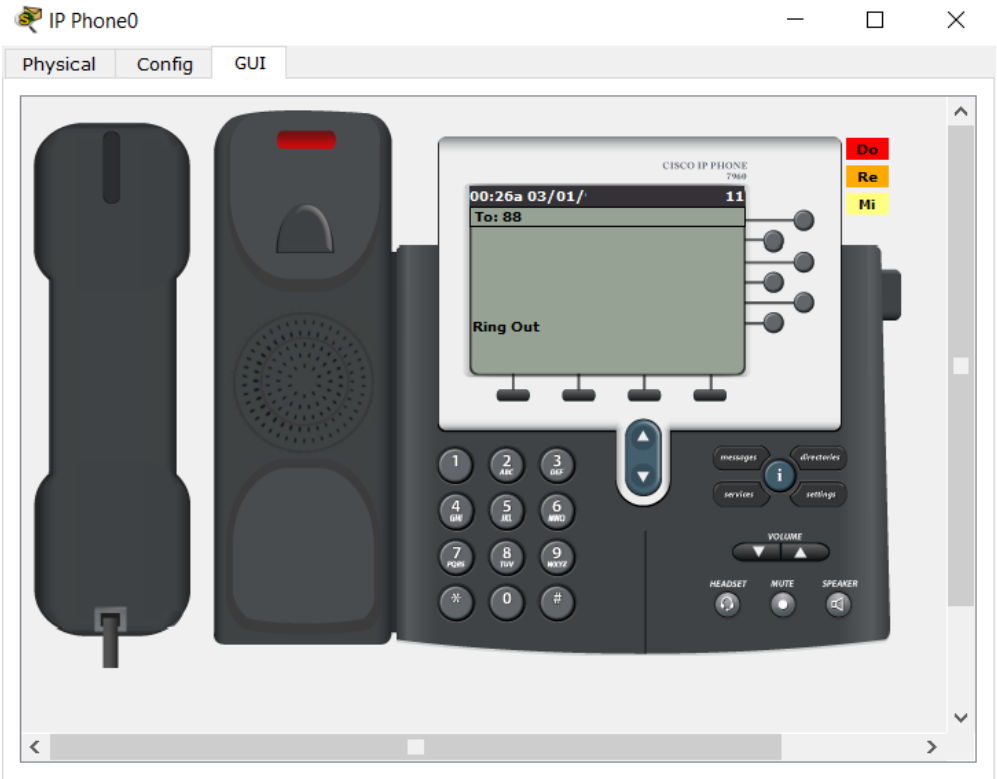
The only confirmation we need is that the IP Phones were successfully registered on the Call Manger Express (This process can take a while). After successful configuration:



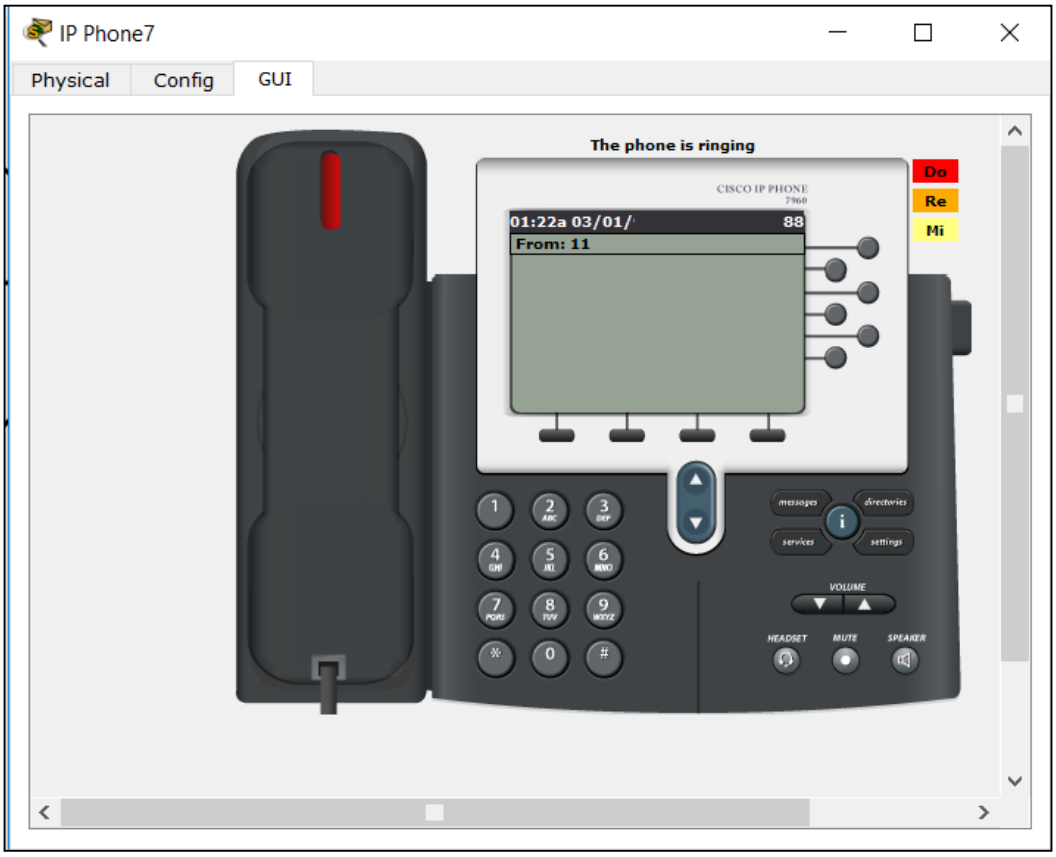
The Graphical User Interface of an IP Phone after successful configuration:

Now let's try calling IP Phone 7 (88) from IP Phone 0 (11):

- Dial 88 and press Enter



IP Phone 7 is ringing:



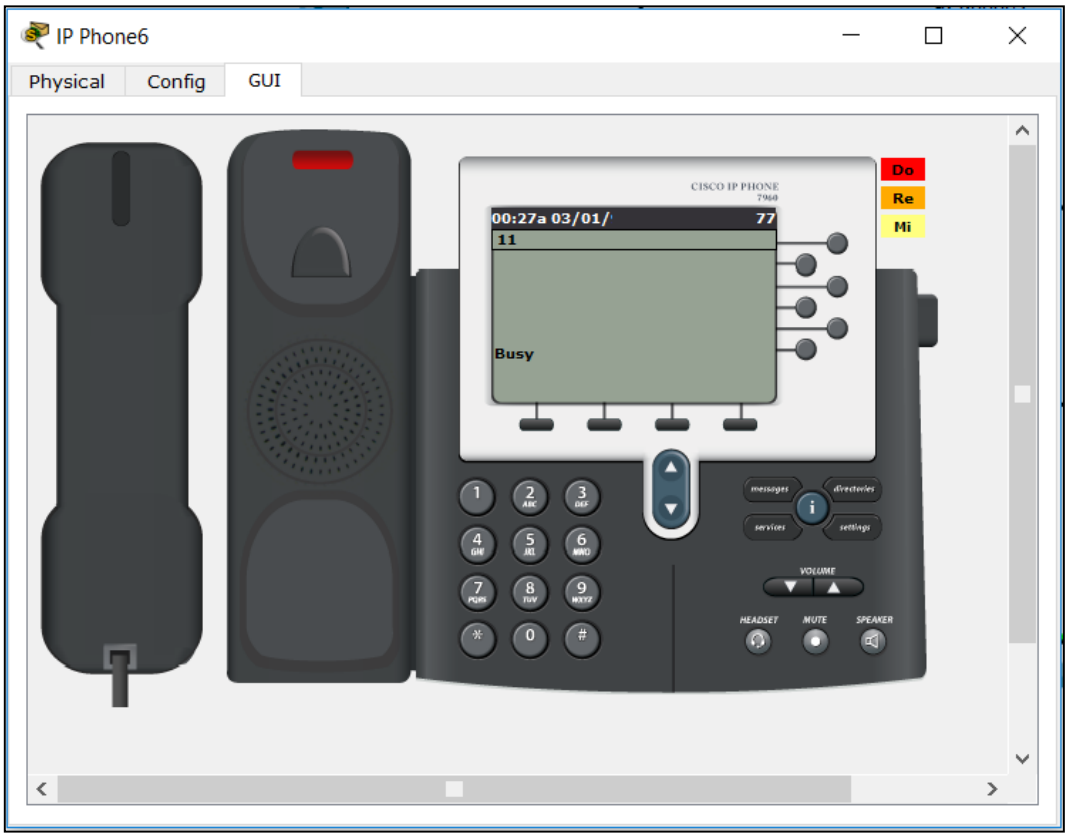
- Press Enter to receive the call

IP Phone 1 and 7 are connected.

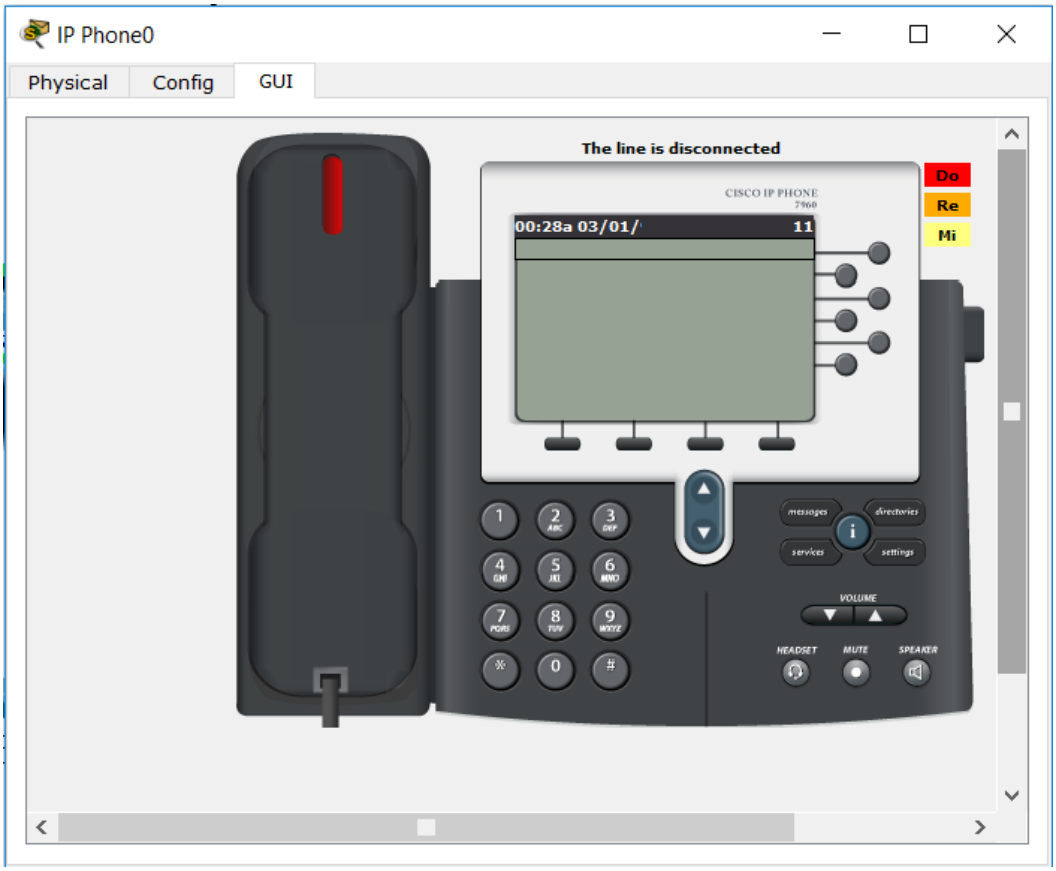
Let’s try calling IP Phone 1 from IP Phone 6, while IP Phone 1 and IP Phone 7 are connected.

- Dial 11 and press Enter.

A “Busy” message pops up.



- To end ongoing call, click on the receiver:



BIBLIOGRAPHY

For doing the projects I have taken help from the following sources:

Website:

- http://computernetworkingnotes.com/cisco/ccna-study-guide/

Teacher:

- Mr. SAYANTAN CHAKROBORTY

  Technical Director,

  Chairman (Non-Profit Organisation)