**Practical-1**

**Aim: Understanding the Sensor Node Hardware.**

Description: Components A wireless sensor network (WSN) is a hardware and software package that typically consists of four parts:

1. ‘Sensors’ connected to each node by a wired connection. In our case, we use sensors that can measure soil moisture, electrical conductivity, soil temperature, water pressure, flow rate, or a range of weather variables (light, air temperature, wind, humidity, etc.). Figure 2. One of many sensors that can be connected to a node, this EC-5 sensor (Decagon Devices, Inc. Pullman, WA) measures volumetric water content (soil moisture).
2. ‘Nodes’ collect the data from sensors and transmit that to a ‘base station’ computer using a one way (in the case of monitoring) or two-way (in the case of monitoring and control) radio. Nodes can simply monitor environmental and soil conditions or can be used to make control decisions. For example, some nodes have the capability to control an electric valve, such as an irrigation valve.
3. ‘Base Station’ computer connects the system to the internet, so that data collected by the nodes, then transmitted to the base station computer, can be viewed anywhere an internet connection is available.
4. ‘Graphical User Interface’ is the web-based software package, that allows the data collected by sensors to be viewed. The software is also used to set irrigation parameters.

Not every WSN will have all four components, but to get optimal functionality the systems developed as part of this project do. A very simple WSN example that many can relate to is that of the wireless environmental monitoring system used by the National Weather Service (NWS). You have probably seen these at a local airport or school. In this case, sensors measure environmental conditions and send this data to a node that wirelessly transmits the data using a cell signal or wireless signal to a base-station computer where NWS employees (and you) can view the current temperature (or rainfall/dew point, wind, etc.) via a website or application (‘app’).

**Practical-2 Aim: Exploring and understanding.**

1. TinyOS computational concepts : Event, Commands and Task
2. nesC model
3. nesC Components

a) Shockfish TinyNode

* Slow CPU
* 8 MHz Texas Instruments MSP430 microcontroller
* Little memory
* 10 KByte RAM, 48 KByte ROM, 512 Kbyte external flash
* Short-range radio
* 868 MHz Xemics XE1205 ultra-low power wireless transceiver
* Light sensor, temperature and humidity sensors + Extension Board TinyNode 584 Exchange of a sensor data
* Two sensor nodes are used for this task
* One node periodically samples its light sensor and broadcasts the sensor reading over its radio
* The other node listens for radio messages and signals if it is getting brighter or darker
* Brighter → The green LED of the receiver is set
* Darker → The red LED of the receiver is set
* No significant change → The yellow LED is set TinyOS:
* TinyOS is an operating system for sensor nodes
* Open source project with a strong academic background
* Hardware drivers, libraries, tools, compiler
* TinyOS applications are written in nesC
* C dialect with extra features
* nesC compiler converts your application into plain C code Why using a new Operating System?
* Measure real-world phenomena Event-driven architecture
* Resource Contraints Hurry up and sleep!
* Adapt to changing technologies Modularity & re-use
* Applications spread over many small nodes Communication is fundamental
* Inaccessible location, critical operation Robustness NesC TinyOS/ Programming Model Programs are built out of components •Two types of components Modules: Implement program logic Configurations: Wire components together
* Components use and provide interfaces

•Components are wired together by connecting interface users with interface providers

**Practical-3**

**Aim: Create and simulate simple adhoc network.**

# Description:

An ad hoc network is a network that is composed of individual devices communicating with each other directly. The term implies spontaneous or impromptu construction because these networks often bypass the gatekeeping hardware or central access point such as a router.

Many ad hoc networks are local area networks where computers or other devices are enabled to send data directly to one another rather than going through a centralized access point.

# Steps:

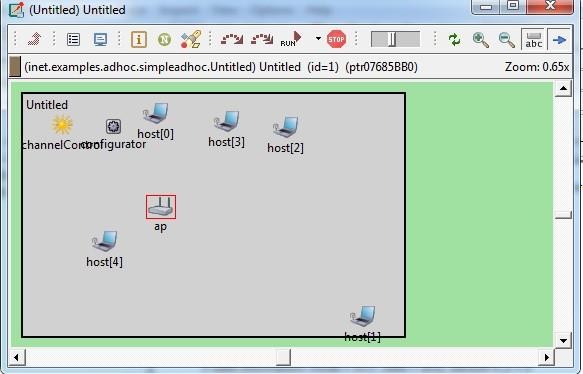
Then open inet folder->uder that open example->select(adhoc) create new folder as

# simpleAdhoc

Select and write click on **simpleAdhoc**->new->select(Network Description File)>give name(NET1)->Next->open dropdown of generated wizad and select new managed Mobility wireless Network Wizard->next->finish.

Select NET1.ned->right click->run as->Omnetpp

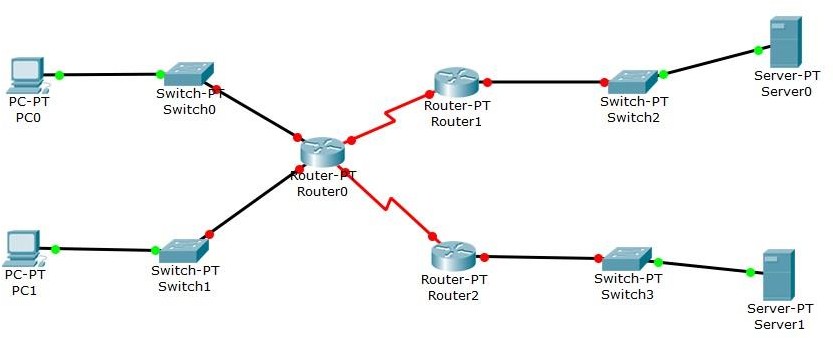
Again click on run in output box to see packet are travelling from one computer to other.



**Practical-4**

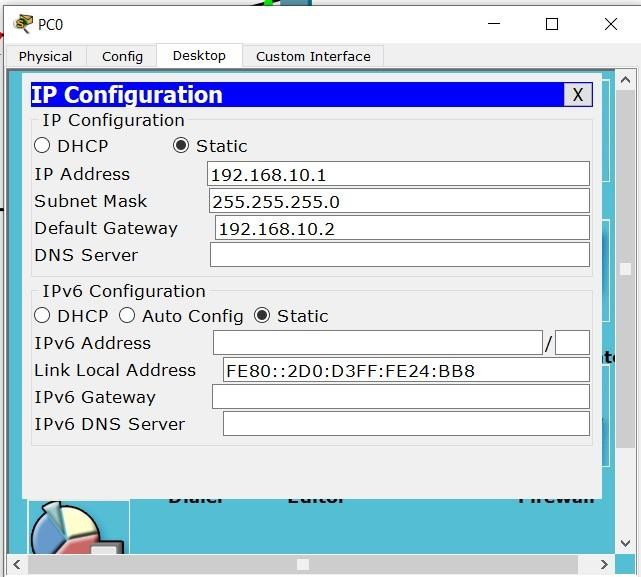
**Aim: Understanding, Reading and Analyzing Routing Table of a network.**

Make connection as shown below :



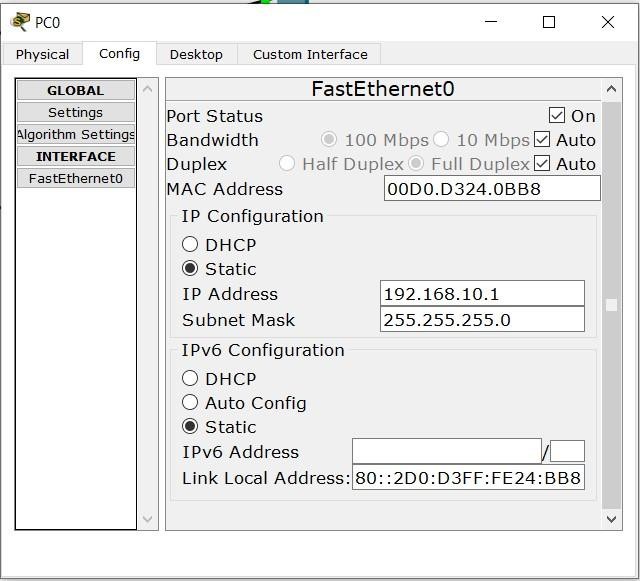
Click on **PC0** : --> Desktop --> IP configuration IP Configuration : Static

IP address : 192.168.10.1 Subnet Mask : 255.255.25.0 Default Gateway : 192.168.10.2



PC0 --> Config --> FastEthernet0 Check for IP address and Subnet mask

Port Status : Check ON



Click on **PC1** : --> Desktop --> IP Configuration IP Configuration : Static

IP address : 192.168.10.3 Subnet Mask : 255.255.255.0 Default Gateway : 192.168.10.4

PC1 --> Config --> FastEthernet0 Check for IP address and Subnet Mask

Click on RouterPT Router0 : --> Config-->FastEthernet0/0 IP address : 192.168.10.5

Subnet mask : 255.255.255.0

Config --> FastEthernet1/0 IP address : 192.168.11.1 Subnet Mask : 255.255.255.0 Port status : Check **ON**

Config --> Serial2/0

IP address : 192.168.12.1 Subnet Mask : 255.255.255.0 Port status : Check **ON**

Config --> Serial3/0

IP address : 192.168.13.1 Subnet Mask : 255.255.255.0 Port status : Check **ON**

Router-PT Router1

Config --> FastEthernet0/0 IP address : 198.168.10.6 Subnet Mask : 255.255.255.0 Port status : Check **ON** Config -->FastEthernet1/0

IP address : 192.168.11.2 Subnet Mask : 255.255.255.0 Port status : Check **ON**

Config -- Serial2/0

IP address : 192.168.12.2 Subnet Mask : 255.255.255.0 Port status : Check **ON**

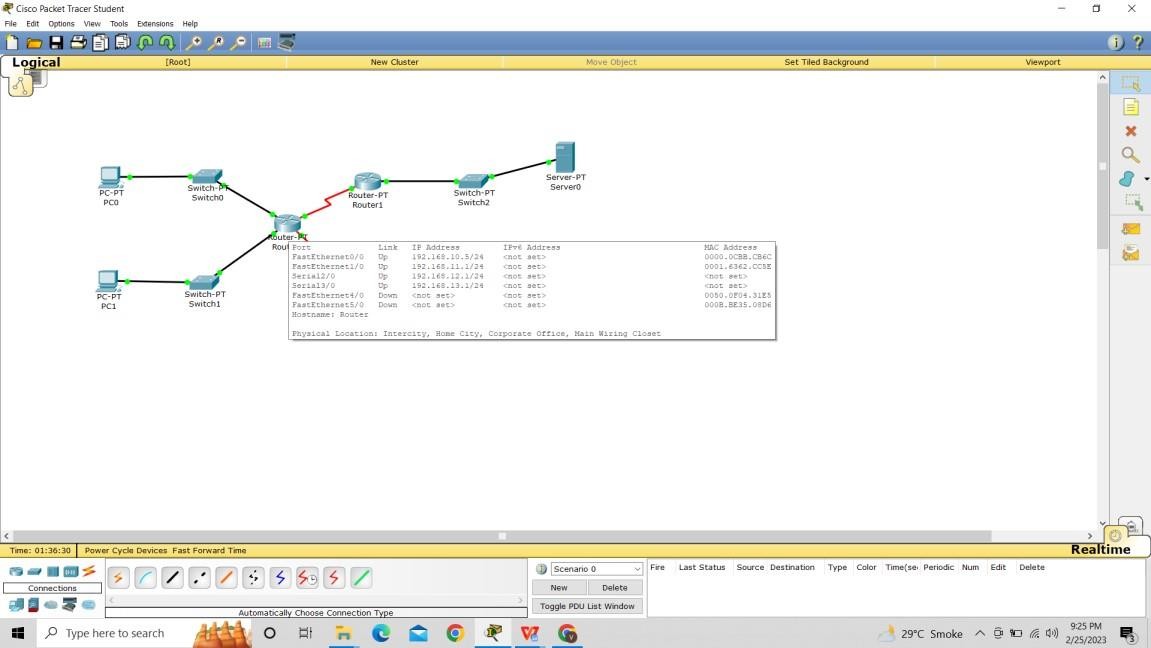
Click on Router-PT Router2 Config --> FastEthernet0/0 IP address : 192.168.11.7 Subnet Mask : 255.255.255.0 Port status : Check **ON**

Config --> FastEthernet1/0 IP address : 192.168.12.3 Subnet Mask : 255.255.255.0

Port status : Check **ON**

Config --> Serial2/0

IP address : 192.168.13.3 Subnet Mask : 255.255.255.0 Port status : Check **ON**



**Practical-5**

**Aim: Create a basic MANET implementation simulation for Packet animation and Packet Tracer.**

# Steps:

**Open omnetpp**

Write command (**omnetpp)** on command line of omnetpp

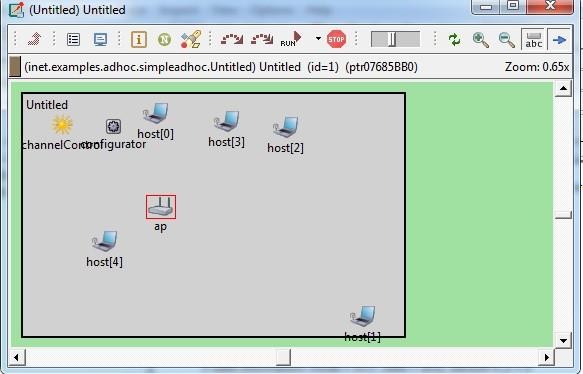
Then open inet folder->uder that open example->select(**Manetrouting**) create new folder as **simpleManetrouting**

Select and write click on **simpleManetrouting**->new->select(Network Description File)-

>give name(NET1)->Next->open dropdown of generated wizad and select new managed Mobility wireless Network Wizad->next->finish.

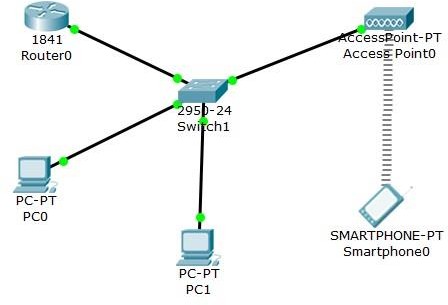
Select NET1.ned->right click->run as->Omnetpp

Again click on run in output box to see packet are travelling from one computer to other.



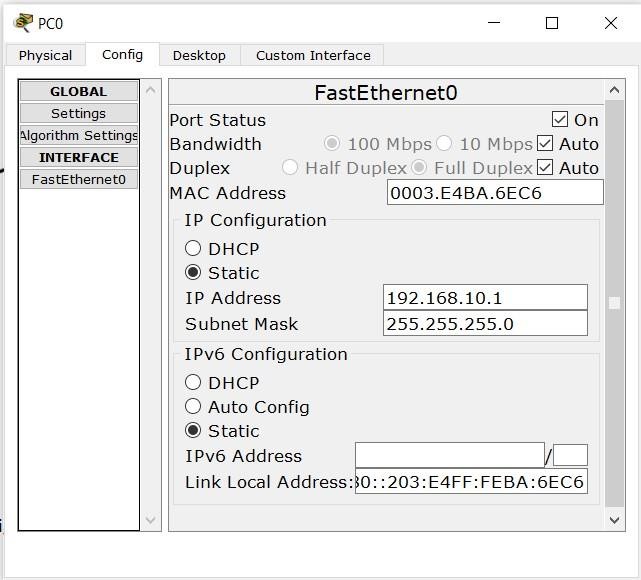
**Practical-6**

**Aim : Implement a Wireless sensor network simulation.**



Click on **PC-PT PC0** : --> Desktop --> IP configuration IP address : 192.168.10.1

Subnet Mask : 255.255.255.0

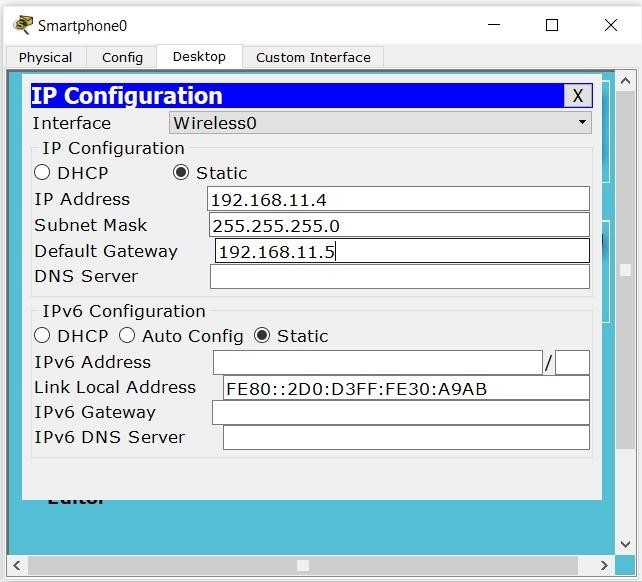
**PC-PT PC0** --> Config --> FastEthernet0 Check for IP address and Subne Mask Check **ON**

Click on **PC-PT PC1** --> Desktopn --> IP Configuration IP address : 192.168.11.2

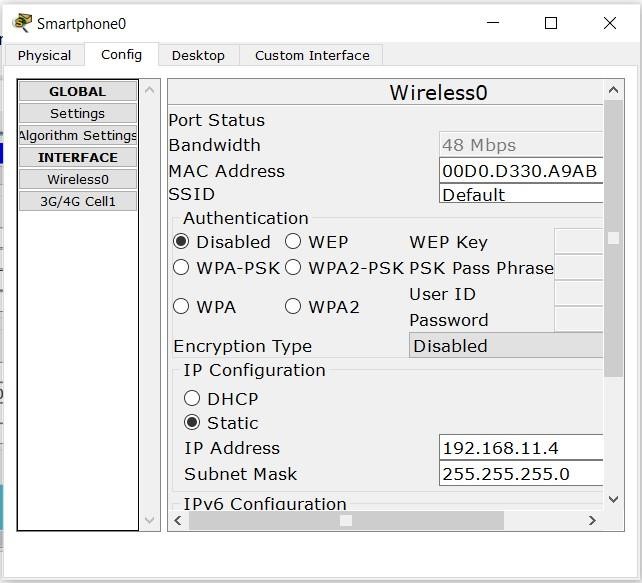
Subnet Mask : 255.255.255.0

**PC-PT PC1** --> Config --> FastEthernet0 Check for IP address and Subnet mask Check **ON**

Click on **SmartPhone0** --> Desktop --> IP configuration

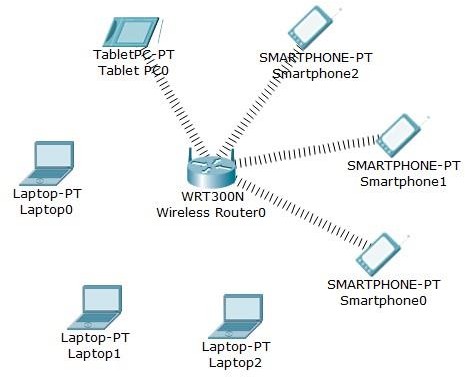


**Smartphone0** --> Config --> Wireless0

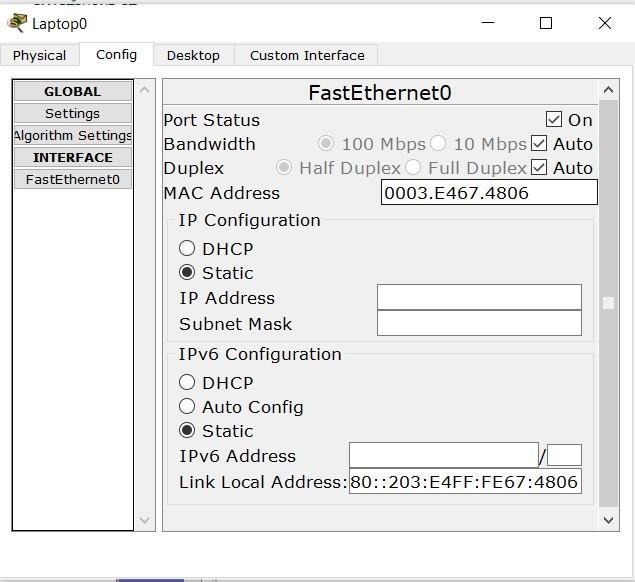


**Practical-7**

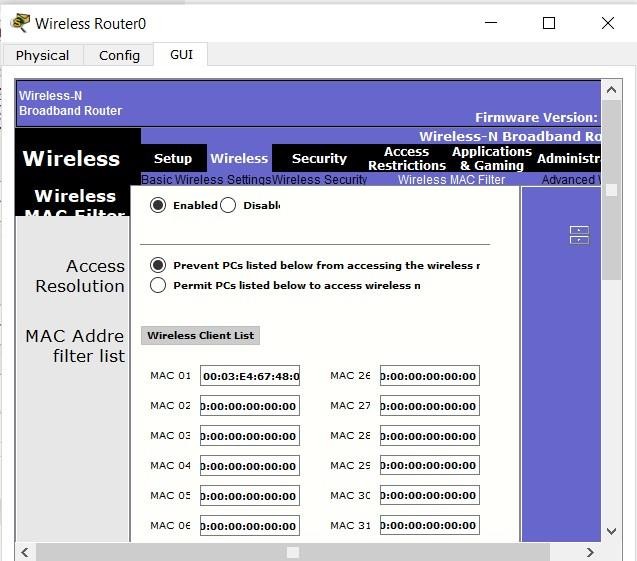
**Aim : Create MAC protocol simulation for wireless sensor Network.**



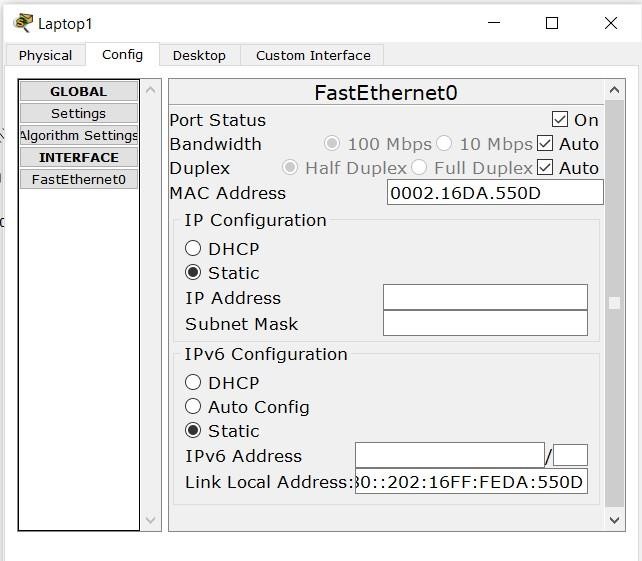
Click on **Laptop0** --> Config --> Copy the MAC address.



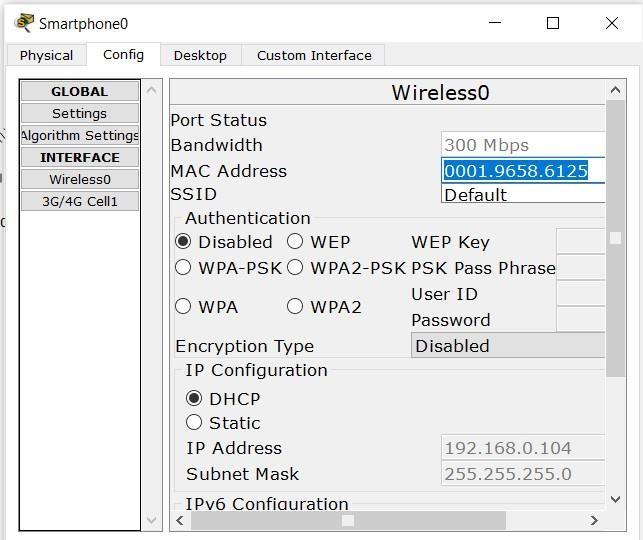
Wireless **Router0** --> GUI --> Wireless --> Wireless MAC Filter Click on Enabled --> Click on Permit PCs listed below to access wireless n MAC 01 : Paste the MAC address of Laptop0 --> Save Setting.



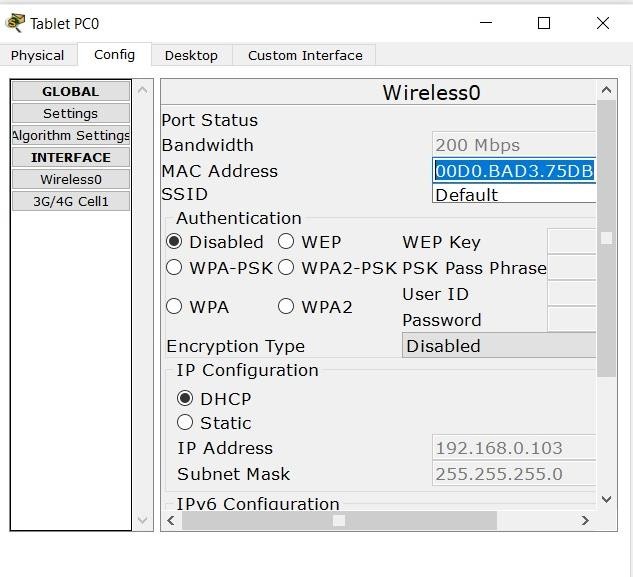
Click on **Laptop1** --> Config --> Copy the MAC address.



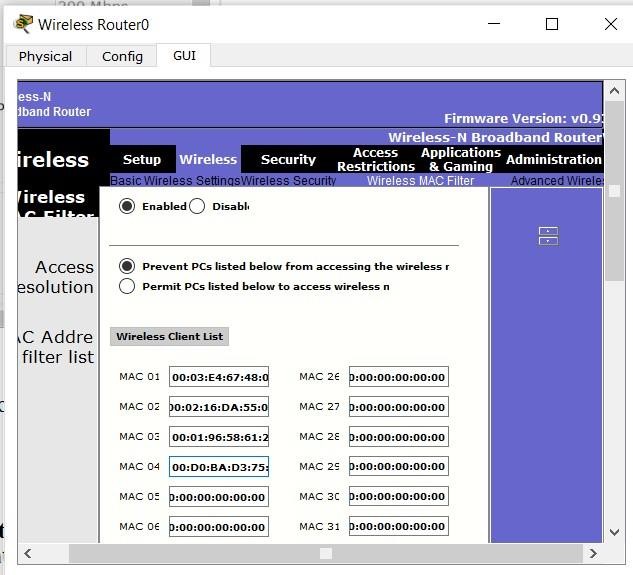
MAC 02 : Paste the MAC address of Laptop1 --> Save Setting. Click on **Smartphone0** --> Config --> Copy the MAC address.



MAC 03 : Paste the MAC address of Smartphone0--> Save Setting. Click on **Tablet PC0** --> Config --> Copy the MAC address.



MAC 04 : Paste the MAC address of Tablet PC0 --> Save Setting.

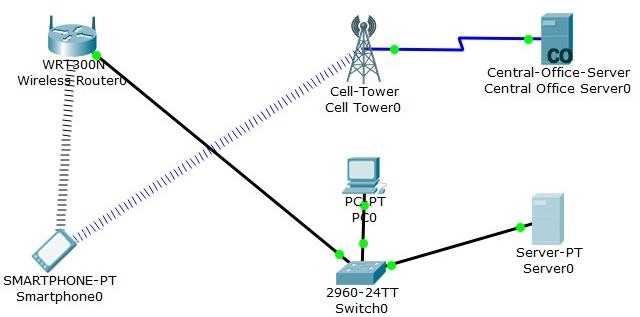


**Practical-8**

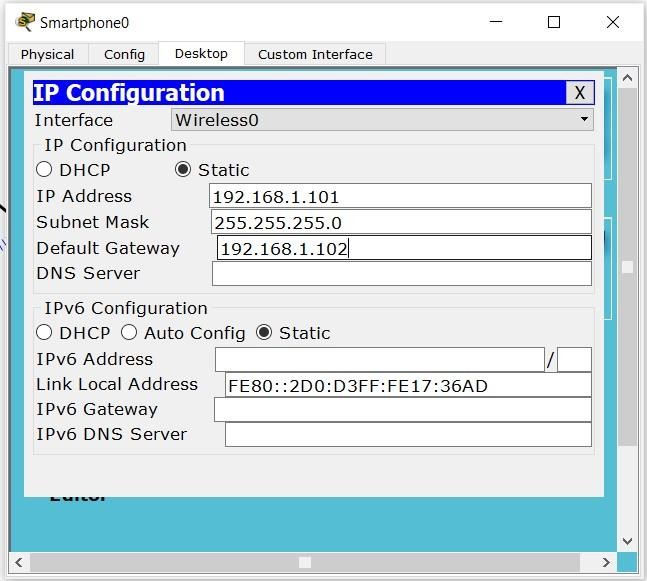
**Aim : Create a mobile network using Cell Tower, Central Office Server, Web browser and Web Server. Simulate connection between them.**

# Steps:

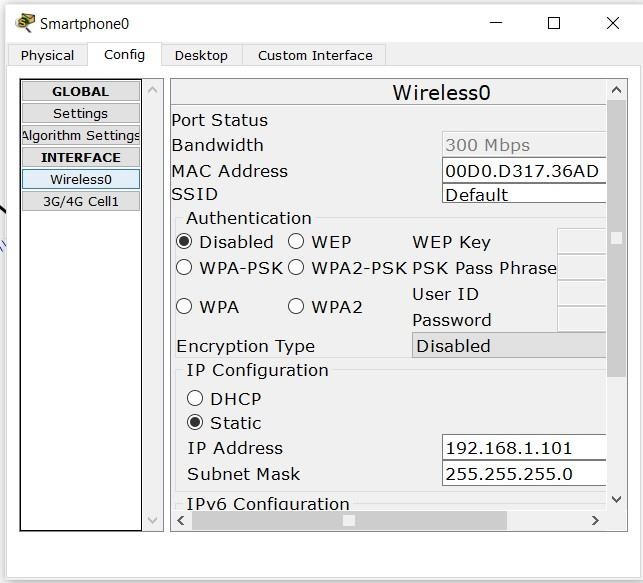
1. Take 1 router, 1 smartphone, Cell-tower, central server, Generic PC, Switch 2960 and generic server.
2. Connect them as shown in the output below.
3. Give IP addresses accordingly.
4. Lastly, send packets to verify the connections.



Click on **Smartphone0** --> Desktop --> IP Configuration



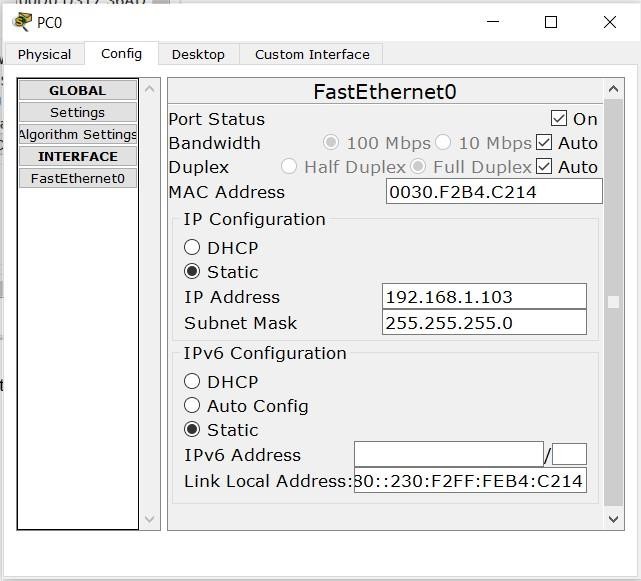
**Smartphone0** --> Config --> Wireless0 Check for IP address and Subnet Mask



Click on **PC0** --> Desktop --> IP Configuration Click on static

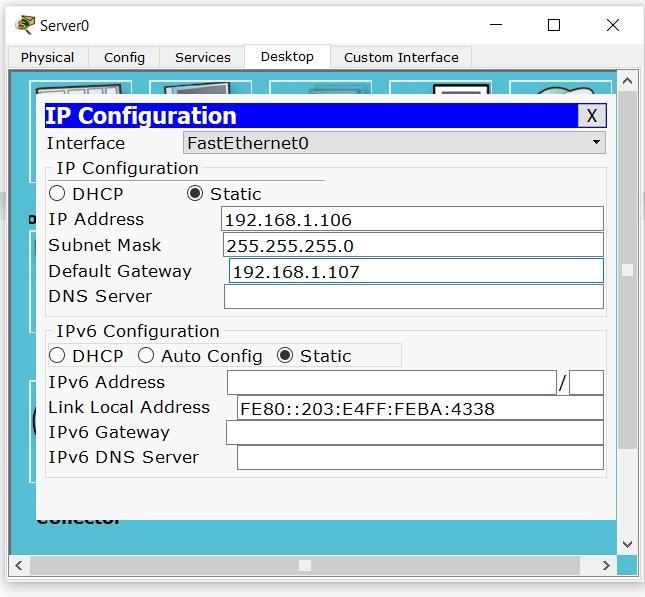
IP address : 192.168.1.103 Subnet Mask : 255.255.255.0 Default Gateway : 192.168.1.104

Click on **PC0** --> Config --> FastEthernet0 Check for IP address and Subnet Mask



Click on **Server0** --> Desktop --> IP Configuration Click on Static

IP address : 192.168.1.106 Subnet Mask : 255.255.255.0 Default Gateway : 192.168.1.107



**Server0** --> Config --> FastEthernet0 Check for IP address and Subnet Mask.

