III B.Tech ECE II Sem – RF SYSTEM DESIGN (20A04605P)

List of Experiments

1. **Design of λ/2, λ/4 microstrip transmission line.**
2. **Design and characterization of Micro strip patch antenna.**
3. **Analyse of a Microstrip Transmission Line and standing wave pattern at various frequencies**
4. **Measure the S parameter of a Microstrip Transmission Line and plot the normalised impedance on a smith chart.**
5. **Design of microstrip inductor and capacitor.**
6. **Design of impedance matching network.**
7. **Design and characterization of RF BJT Amplifier and LNA.**
8. **Design of low pass, high pass, band pass and band stop filter at RF.**
9. **Design and characterization of RF Mixer.**
10. **Design and characterization of VCO.**
11. **Design and simulate a Schottky Diode and RF Switch.**
12. **Analyse and measure the gain of a Power Amplifier and equalize its gain using an Equalizer.**

Introduction to HFSS

RF SYSTEM DESIGN HFSS

HFSS is a high-performance full-wave electromagnetic (EM) field simulator for arbitrary 3D volumetric passive device modeling that takes advantage of the familiar Microsoft Windows graphical user interface. It integrates simulation, visualization, solid modeling, and automation in an easy-to-learn environment where solutions to your 3D EM problems are quickly and accurately obtained. Ansoft HFSS employs the Finite Element Method (FEM), adaptive meshing, and brilliant graphics to give you unparalleled performance and insight to all of your 3D EM problems. Ansoft HFSS can be used to calculate parameters such as S Parameters, Resonant Frequency, and Fields.

HFSS USES

Typical uses include:

- **Package Modeling** BGA, QFP,
  Flip-Chip.
- **PCB Board Modeling**
  Power/Ground planes, Mesh Grid Grounds, Backplanes.
- **Silicon/GaAs**

  Spiral Inductors, Transformers.
- **EMC/EMI**

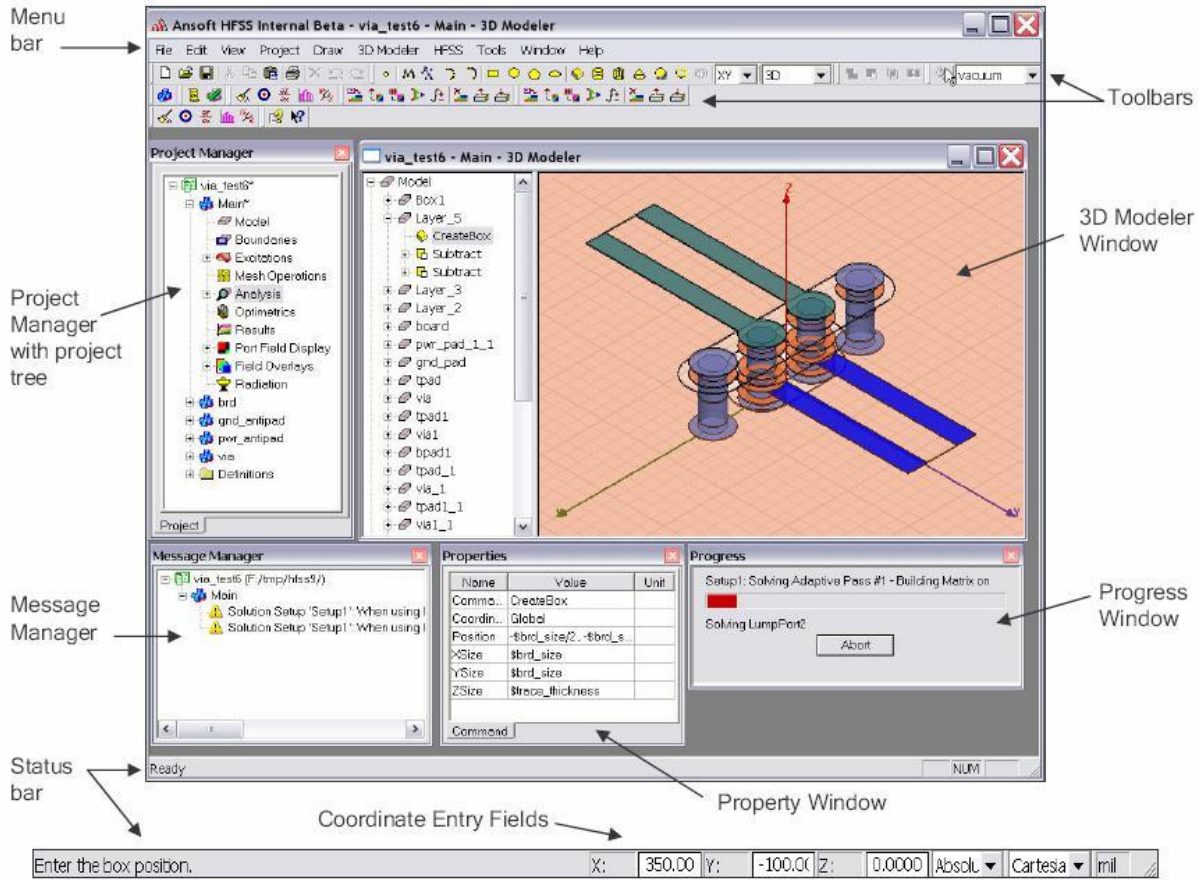
  Shield Enclosures, Coupling, Near- or Far-Field Radiation
- **Antennas/Mobile Communications**

  Patches, Dipoles, Horns, Conformal Cell Phone Antennas, Quadrafilar Helix, Specific Absorption Rate(SAR), Infinite Arrays, Radar Cross Section(RCS), Frequency Selective Surfaces(FSS).

The Ansoft HFSS window has several optional panels:

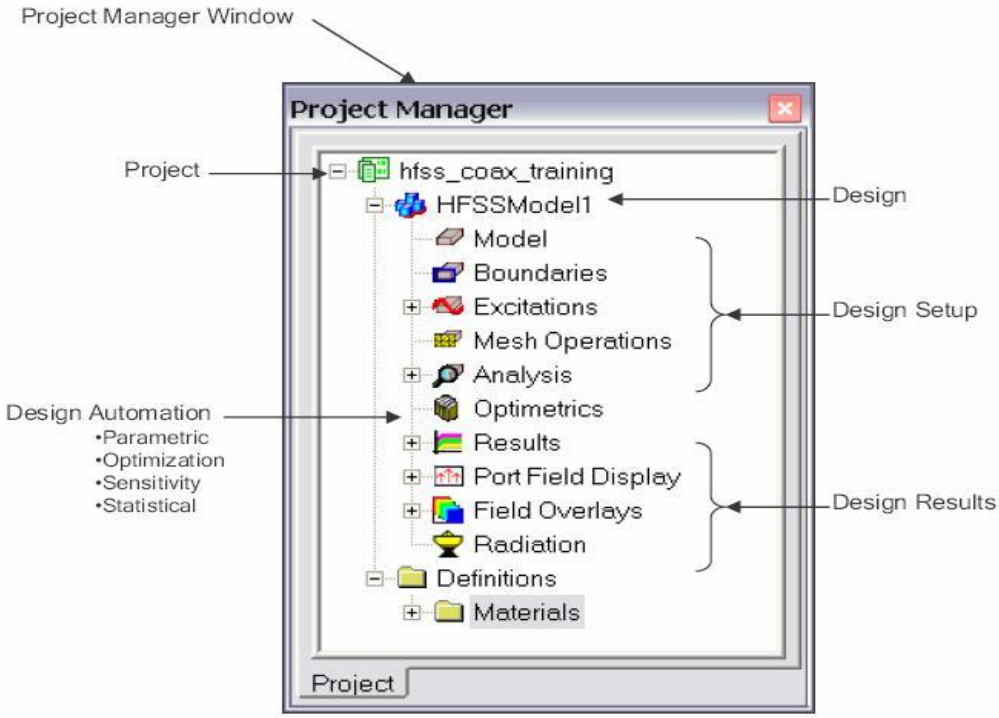
i. Project Manager
ii. Message Manager
iii. Property Window
iv. Progress Window
v. 3D Modeler Window

These above managers and windows are shown in Fig (1) and their details are given in coming sections.



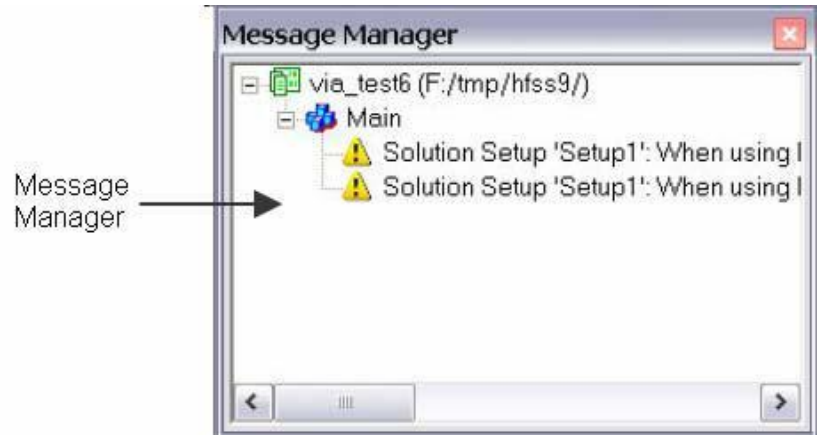
Project Manager

A Project Manager which contains a design tree which lists the structure of the project is shown in Fig (2).



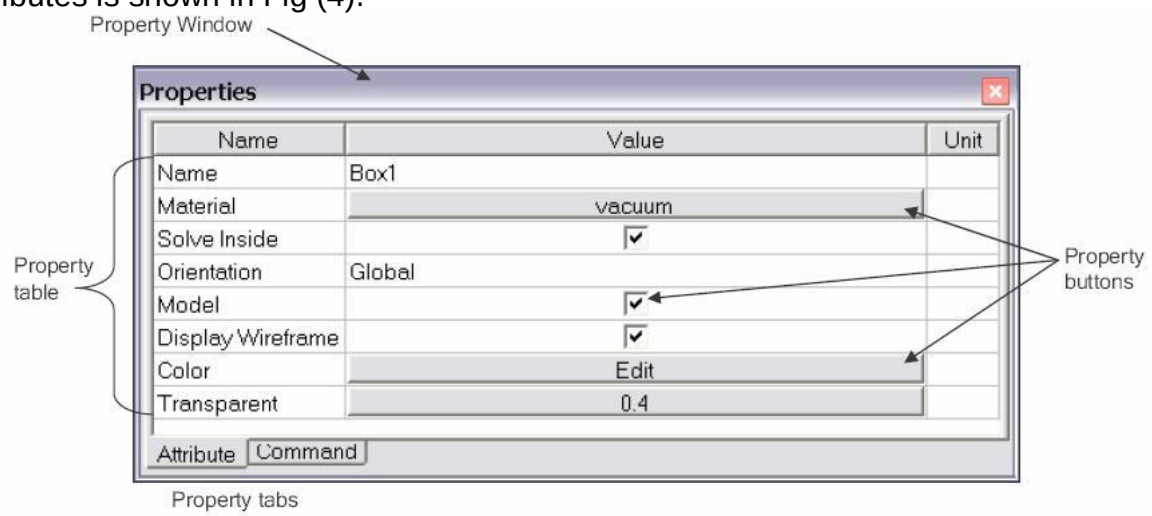
Message Manager

A Message Manager that allows you to view any errors or warnings that occur before you begin a simulation is shown in Fig (3).



Property Window

A Property Window that displays and allows you to change model parameters or attributes is shown in Fig (4).

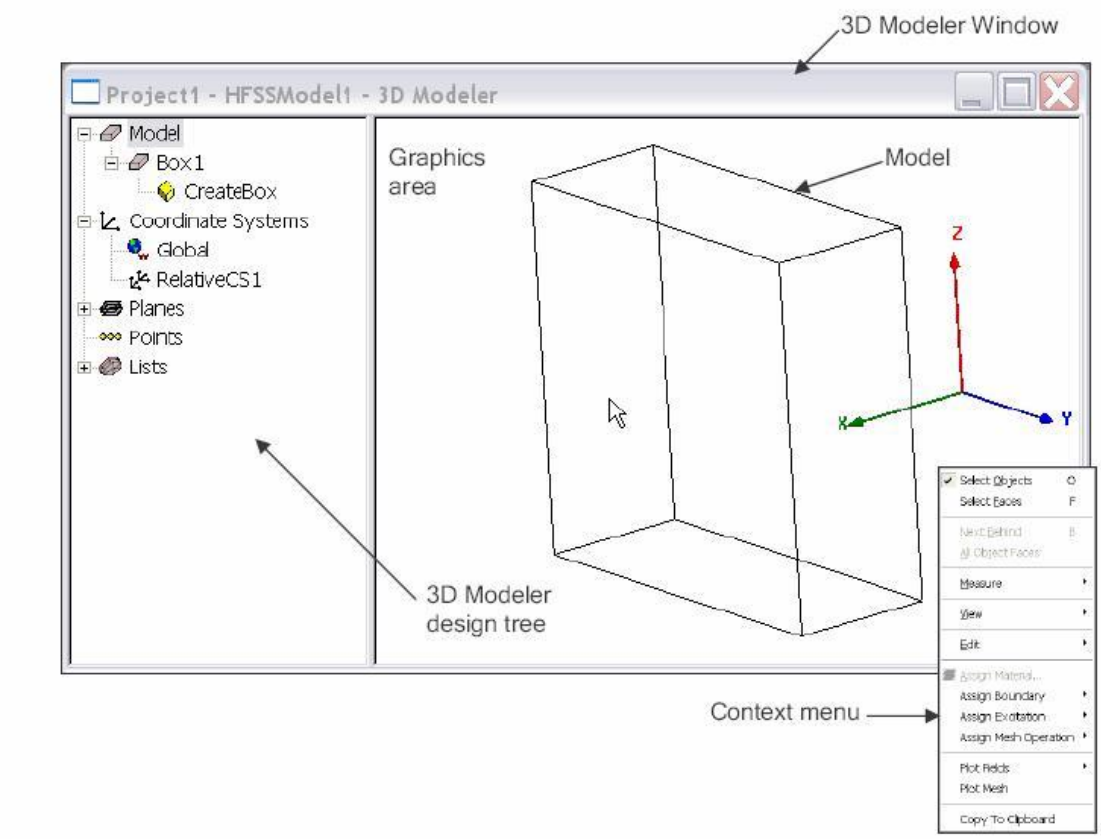


Progress Window

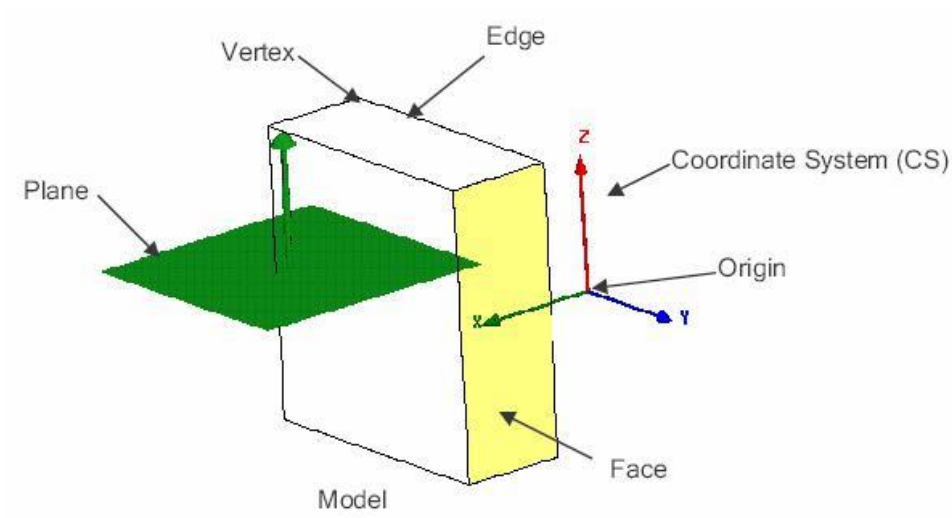
A Progress Window that displays solution progress is shown in Fig (5).

3D Modeler Window

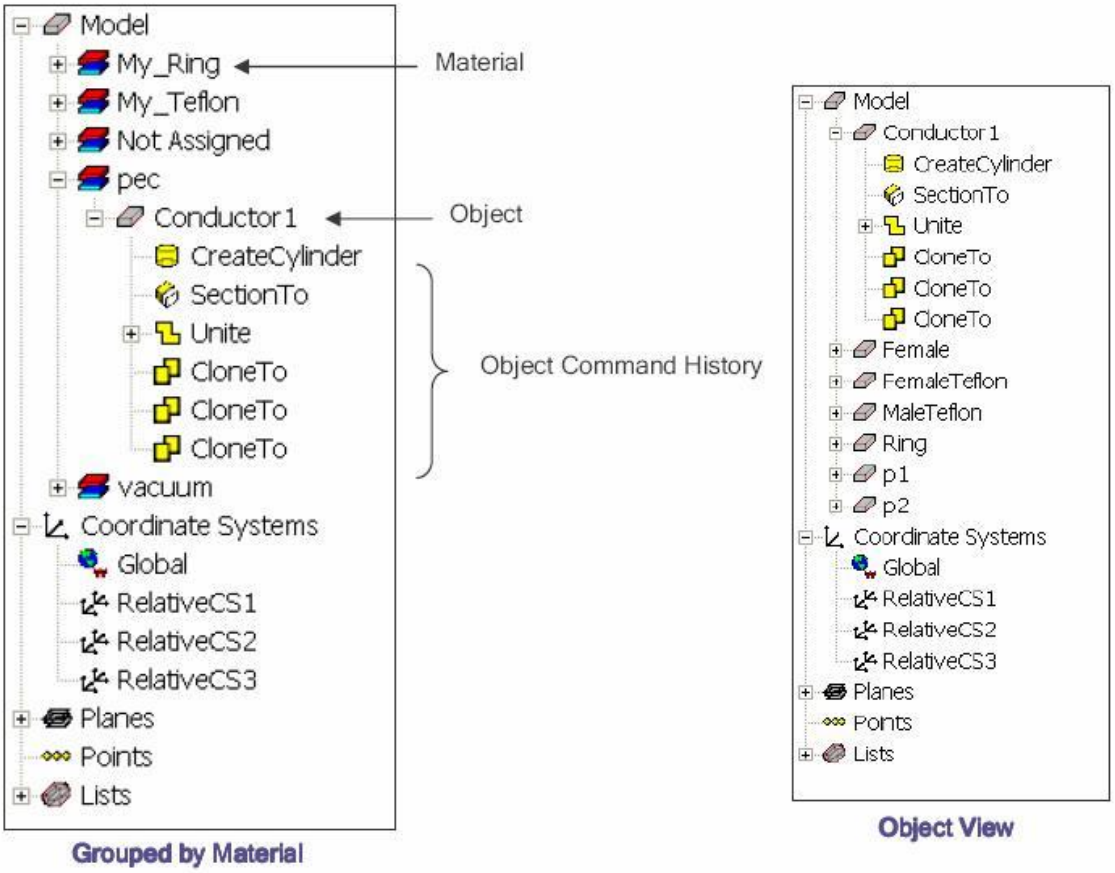
A 3D Modeler Window which contains the model and model tree for the active design is shown in Fig (6), model and model tree are shown in Fig (7) and Fig (8) respectively.



Ansoft HFSS 3D Modeler Window.



3D Modeler Design Tree.

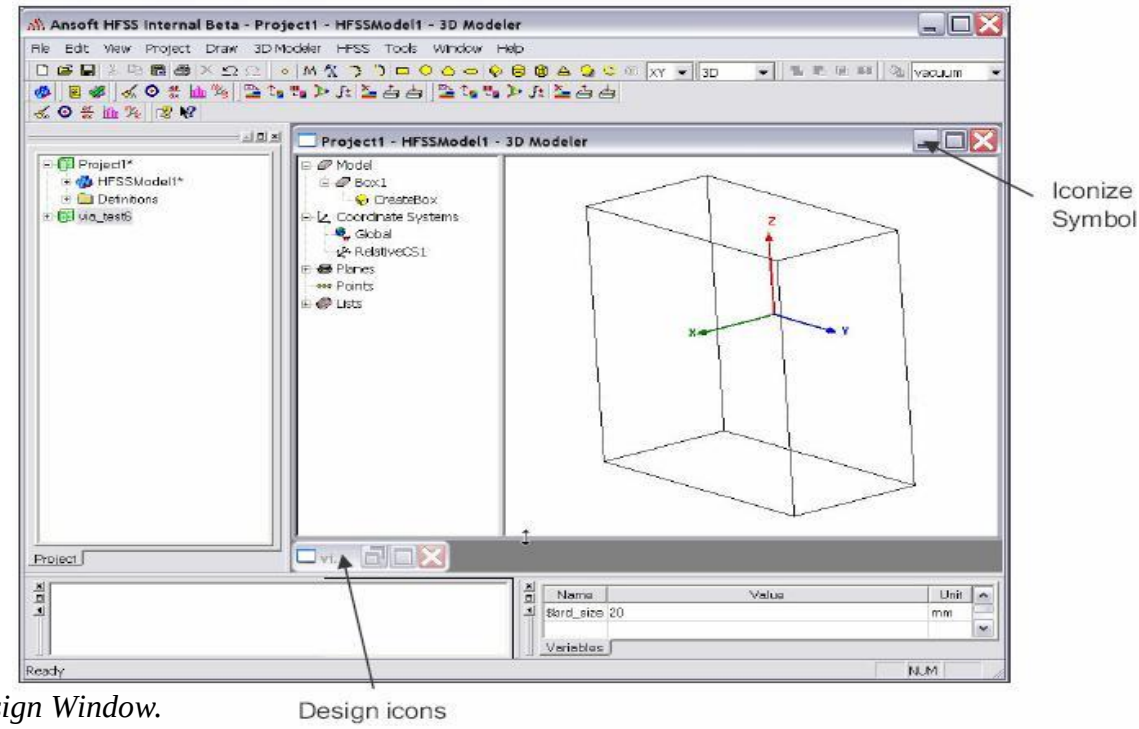


Design Windows

In the Ansoft HFSS Desktop, each project can have multiple designs and each design is displayed in a separate window. You can have multiple projects and design windows open at the same time. Also, you can have multiple views of the same design visible at the same time.

To arrange the windows, you can drag them by the title bar, and resize them by dragging a corner or border. Also, you can select one of the following menu options: Window >Cascade, Window >Tile Vertically, or Window > Tile Horizontally.

To organize your Ansoft HFSS window, you can iconize open designs. Click the Iconize ** symbol in the upper right corner of the document border. An icon appears in the lower part of the Ansoft HFSS window. If the icon is not visible, it may be behind another open document. Resize any open documents as necessary. Select the menu item Window > Arrange Icons to arrange them at the bottom of the Ansoft HFSS window. Select the menu item Window > Close All to close all open design. You are prompted to Save unsaved designs.

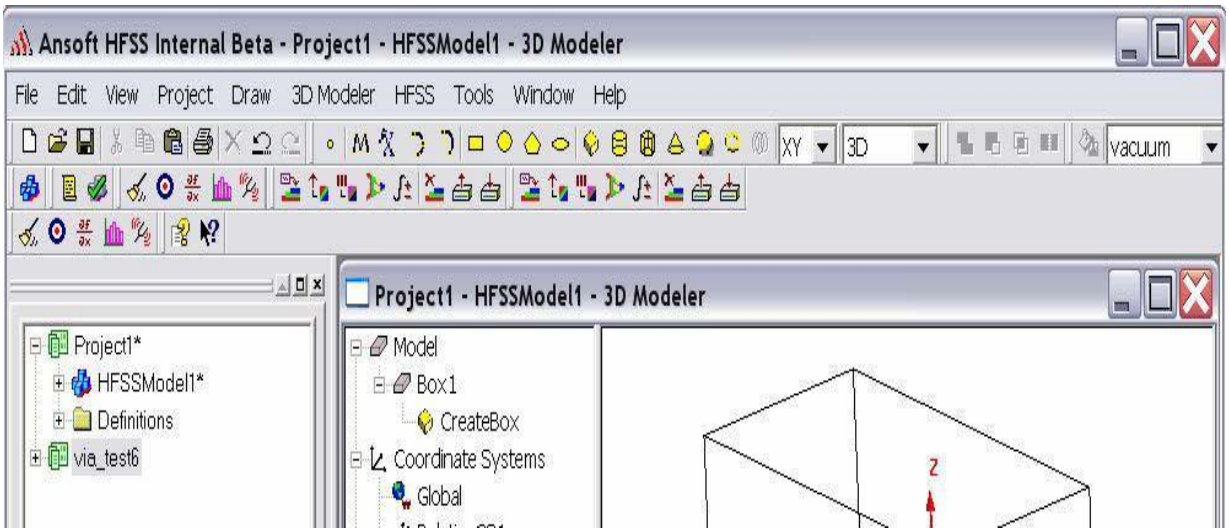


Design Window.

Toolbars

The toolbar buttons are shortcuts for frequently used commands. Most of the available toolbars are displayed in this illustration of the Ansoft HFSS initial screen, but your Ansoft HFSS window probably will not be arranged this way.

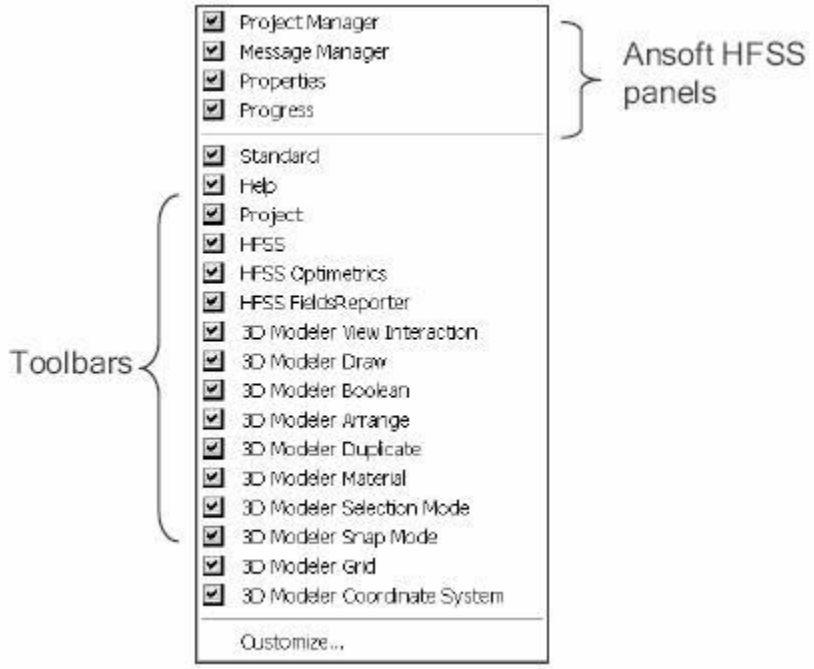
You can customize your toolbar display in a way that is convenient for you. Some toolbars are always displayed; other toolbars display automatically when you select a document of the related type. For example, when you select a 2D report from the project tree, the 2D report toolbar displays, as shown in Fig (9)



Ansoft HFSS Toolbars.

To display or hide individual toolbars

- Right-click the Ansoft HFSS window frame.
  - A list of all the toolbars is displayed. The toolbars with a check mark beside them are visible; the toolbars without a check mark are hidden. Click the toolbar name to turn its display on or off.
- To make changes to the toolbars, select the menu item Tools > Customize.



Ansoft HFSS Panels and Toolbars.

Customize and Arrange Toolbars

To customize toolbars:

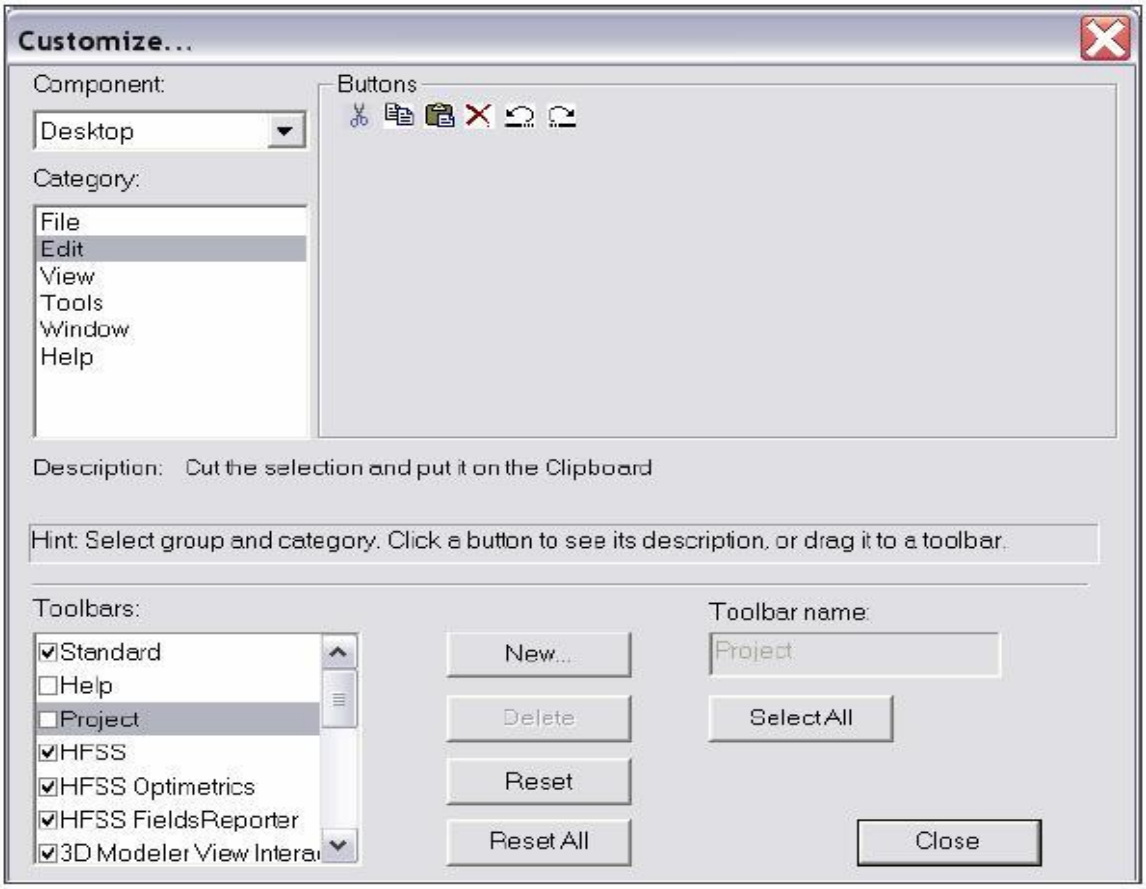
- Select the menu item Tools > Customize, or right-click the Ansoft HFSS window frame and click Customize at the bottom of the toolbar list.
- In the Customize dialog, you can do the following: ○

  View a Description of the toolbar commands.

  i. Select an item from the Component pull-down list.
  ii. Select an item from the Category list.
  iii. Using the mouse click on the Buttons to display the Description.
  iv. Click the Close button when you are finished.

Toggle the visibility of toolbars

i. From the Toolbar list, toggle the check boxes to control the visibility of the toolbars.
ii. Click the Close button when you are finished.



Ansoft HFSS customize.

Ansoft HFSS Desktop

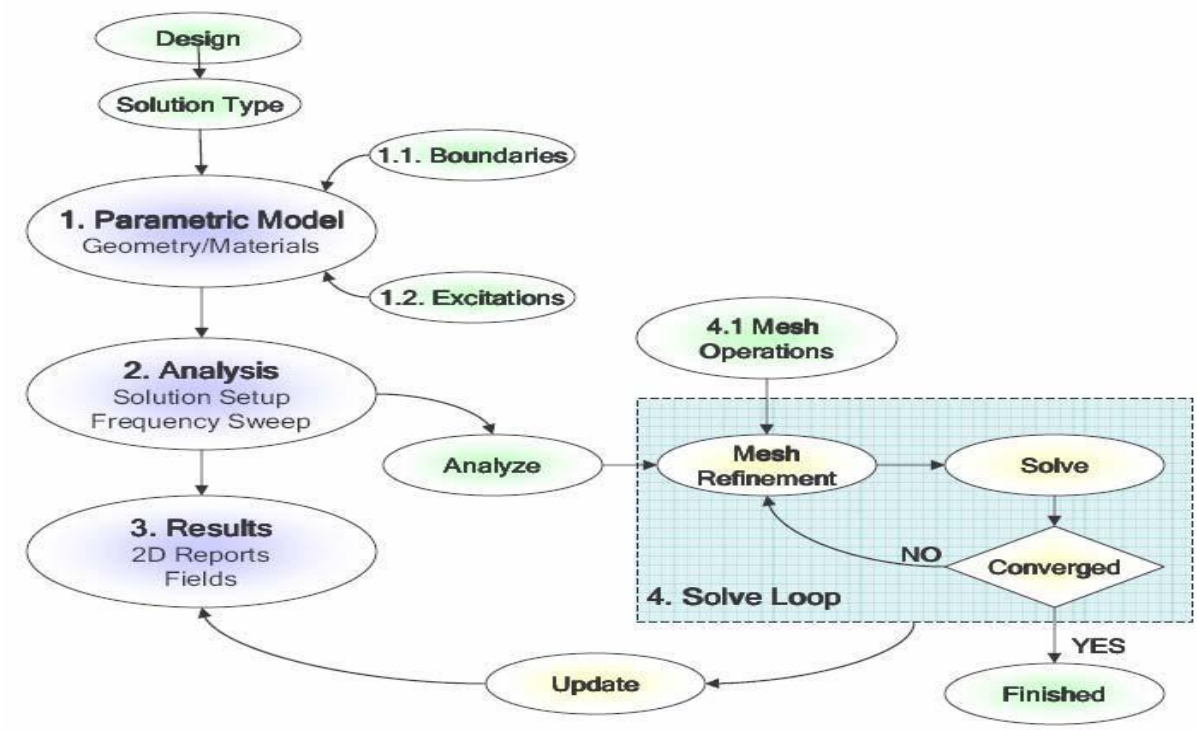
The Ansoft HFSS Desktop provides an intuitive, easy-to-use interface for developing passive RF device models. Creating designs, involves the following:

i. **Parametric Model Generation** – creating the geometry, boundaries and excitations.

ii. **Analysis Setup** – defining solution setup and frequency sweeps.
iii. **Results** – creating 2D reports and field plots.
iv. **Solve Loop** - the solution process is fully automated.

To understand how these processes co-exist, examine the illustration shown In Fig(12).

Ansoft HFSS Desktop.



Opening a HFSS project

This section describes how to open a new or existing project.

Opening a New project

To open a new project:

i. In an Ansoft HFSS window, select the menu item File > New.

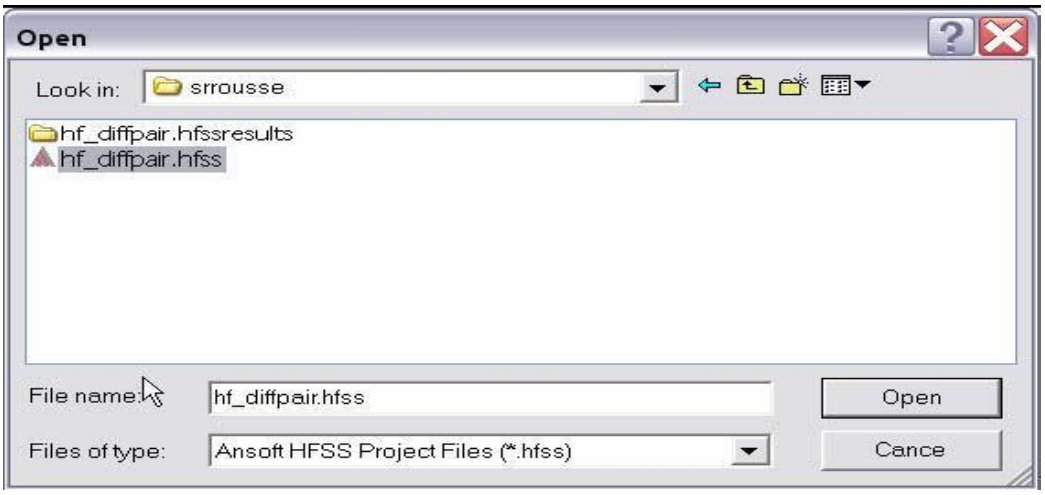
ii. Select the menu Project > Insert HFSS Design.

Opening an Existing HFSS project

To open an existing project:

i. In an Ansoft HFSS window, select the menu File > Open. Use the Open dialog to select the project.

ii. Click Open to open the project



Opening a HFSS project.

Opening an Existing Project from Explorer

You can open a project directly from the Microsoft Windows Explorer. To open a project from Windows Explorer, do one of the following:

i. Double-click on the name of the project in Windows Explorer.

ii. Right-click the name of the project in Windows Explorer and select Open from the shortcut menu.

Set Solution Type

This section describes how to set the Solution Type. The Solution Type defines the type of results, how the excitations are defined, and the convergence. The following Solution Types are available:

i. **Driven Modal** - calculates the modal-based S-parameters. The S-matrix solutions will be expressed in terms of the incident and reflected powers of waveguide modes.

ii. **Driven Terminal** - calculates the terminal-based S-parameters of multiconductor transmission line ports. The S-matrix solutions will be expressed in terms of terminal voltages and currents.

iii. **Eignemode** – calculate the eigenmodes, or resonances, of a structure. The Eigenmode solver finds the resonant frequencies of the structure and the fields at those resonant frequencies.

Convergence

i. **Driven Modal** – Delta S for modal S-Parameters. This was the only convergence method available for Driven Solutions in previous versions.

ii. **Driven Terminal New** – Delta S for the single-ended or differential nodal S-Parameters.

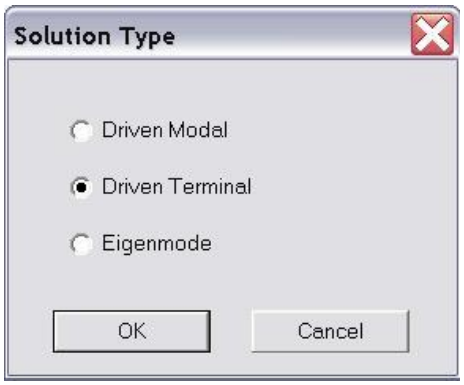
iii. **Eigenmode** - Delta F

To set the solution type:

Select the menu item HFSS > Solution Type Solution Type Window:

- Choose one of the following:
  - Driven Modal
  - Driven Terminal
  - Eigenmode

- Click the OK button

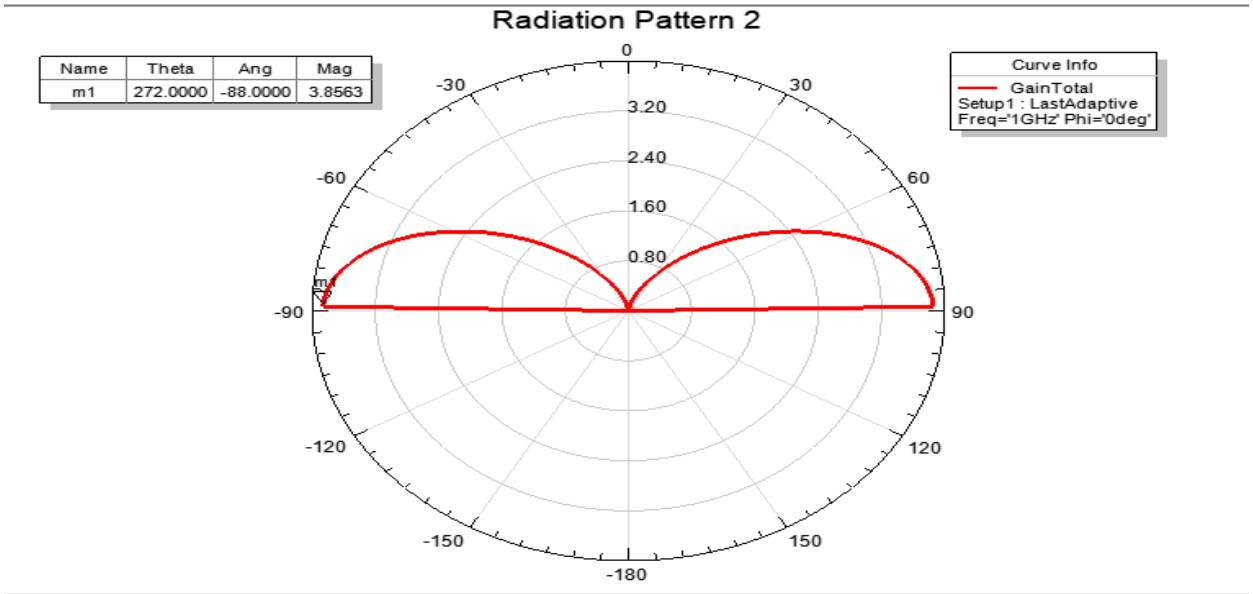


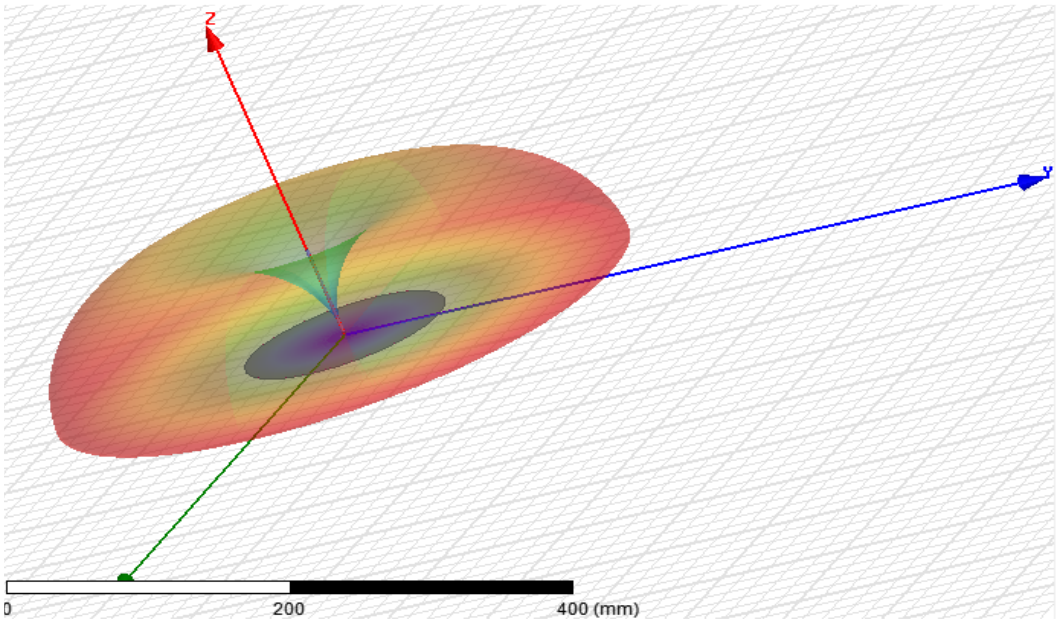
Solution Type.

Parametric Model Creation

The Ansoft HFSS 3D Modeler is designed for ease of use and flexibility. The power of the 3D Modeler is in its unique ability to create fully parametric designs without editing complex macros/model history.

The purpose of this section is to provide an overview of the 3D Modeling capabilities. By understanding the basic concepts outlined here you will be able to quickly take advantage of the full feature set offered by the 3D Parametric Modeler.





Exp.No: 01 **Date:**

MICROSRIP TRANSMISSION LINE

a). Aim: Design of λ/2, microstrip transmission line.

Apparatus Required:

1. **Computer**
2. **Hfss software.**

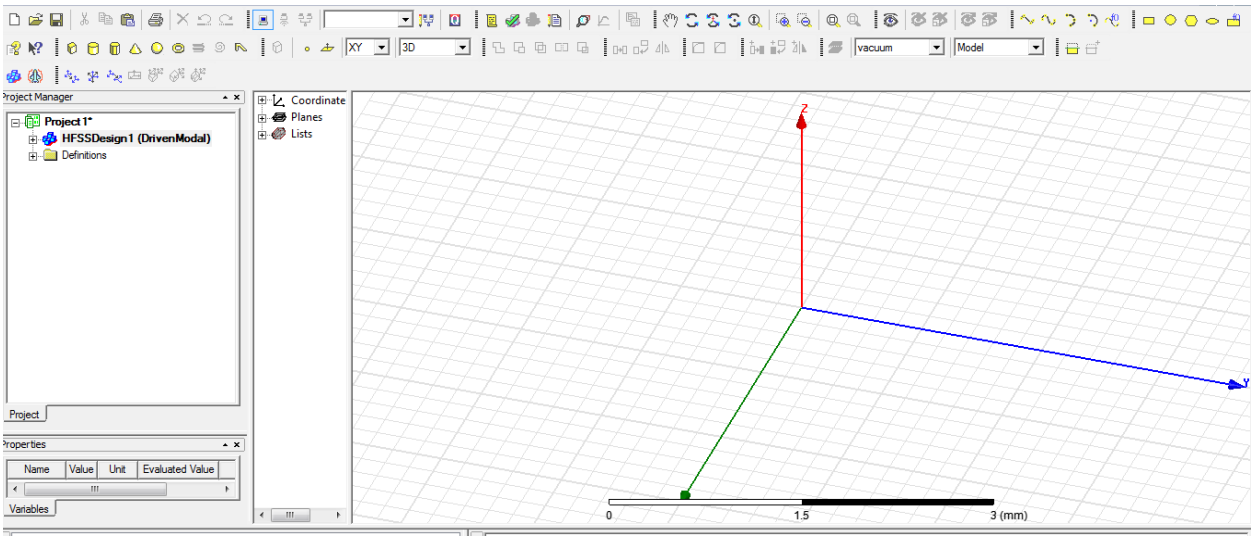
Procedure:

1. Open HFSS software and Insert new HFSS design.
2. Adjust the co-ordinates.
3. Create a ground plane(Rectangular).
4. Create a dielectric substrate with FR4_Epoxy material with same size of ground plane with z-height 1.6mm.
5. Creating the TL the ground plane & substance.
6. Create two ports (port1&port2).
7. Now give the perfect E to ground .
8. Create assign excitation-lampudport.
9. Then create radiation boundary on the designed ground and assign boundary **b**. The radiation should be given to all the faces except at ground.
10. Assign frequency and no. of passes.
11. Now add freq sweep – fast – linear count.
12. Now check validation and analyze all.
13. Then go to results – Create model solution – rectangular plot – new report. Plot both
14. Then click HFSS – click radiation – click far field – infinite sphere and give values to phi and theta.
15. Then click on results and create far field.
16. Click plot of 3D – gain-dB – new report.

Tabular Column:

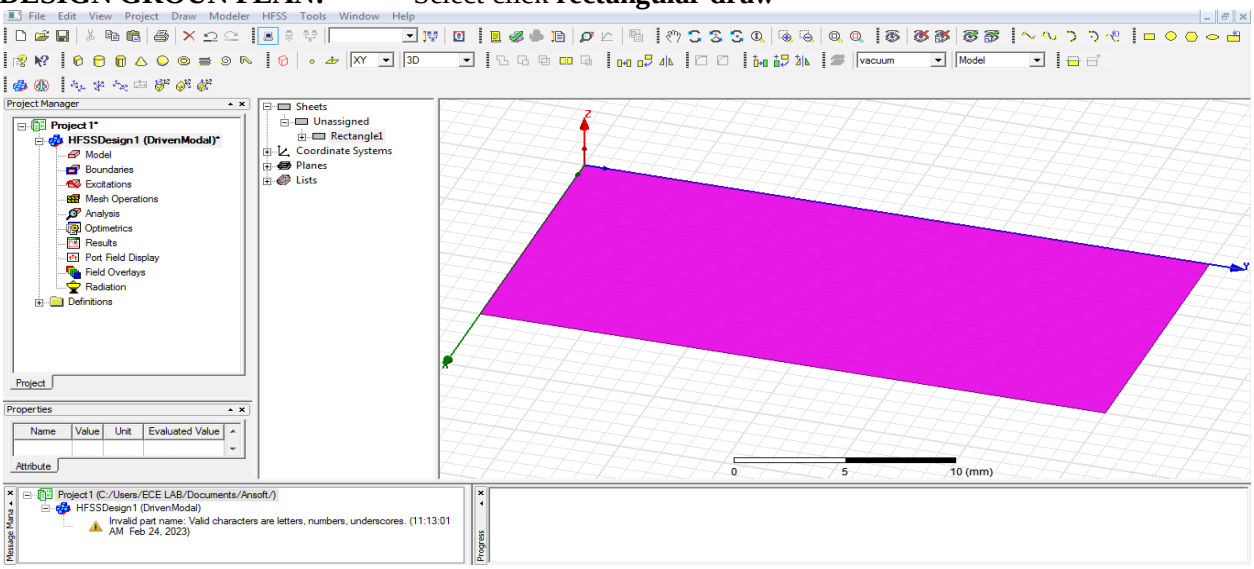
DESIGN CONSIDERATIONS Parameters	Width	Length	Height	Position
Ground plane				
substance				
TL				
Port1				
Port2				
radiation boundary				

Design: **MICROSTRIP TRANSMISSION LINE Λ/2 USING HFSS (5 GHZ)**



Open **HFSS** project-click project , open - project insert **HFSS** design.

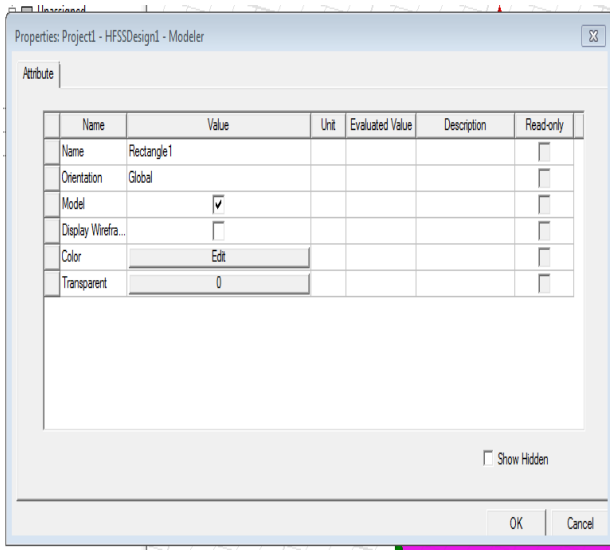
DESIGN GROUN PLAN: Select click **rectangular draw**



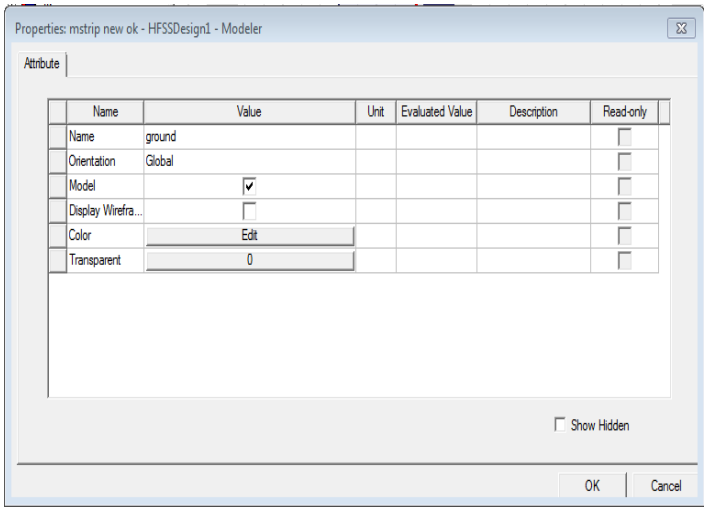
Double click **rectangle1**

Name : [ground plan] , Colour : edit [as your wish]

before name change



after name change



Double click **create rectangle:**

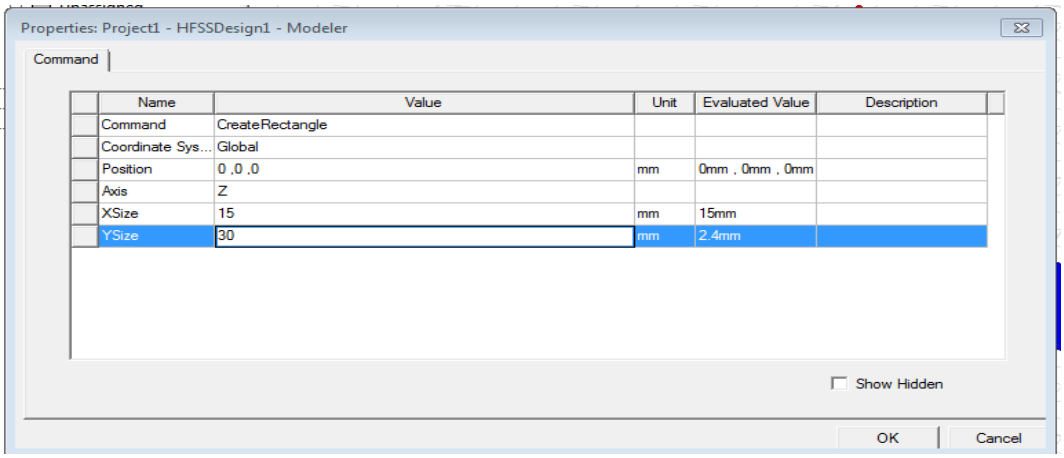
Position : 0,0,0

Axis : z

X size : 15 mm

Y size : 30 mm

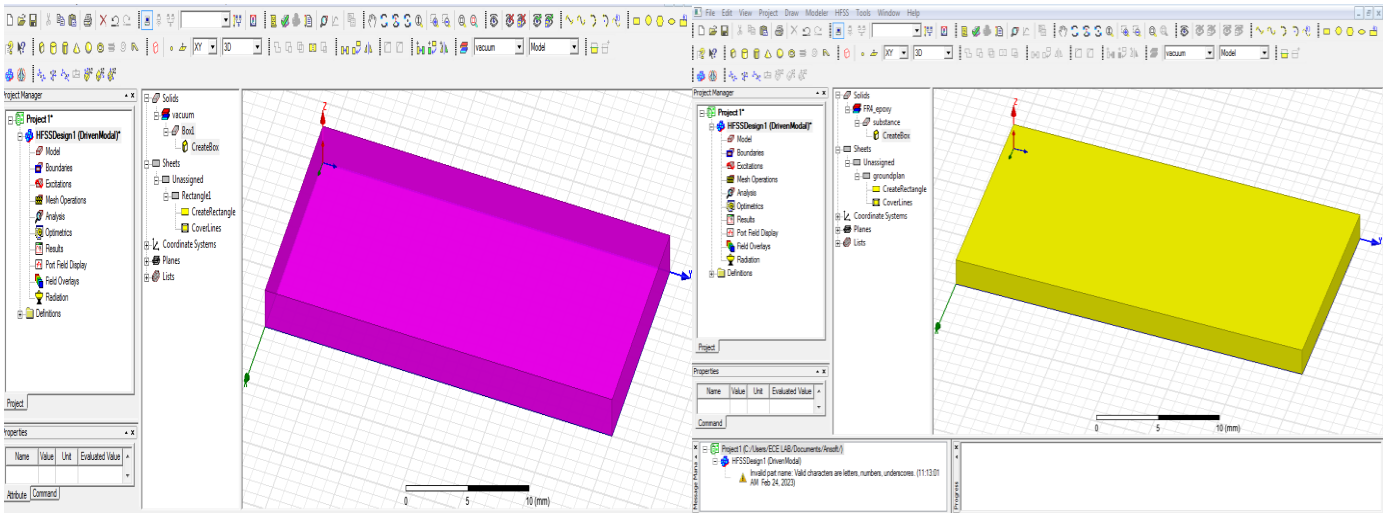
Ok



Select fit all the contents in the view

CREATE SUBSTANCE

Select draw the **BOX** and design



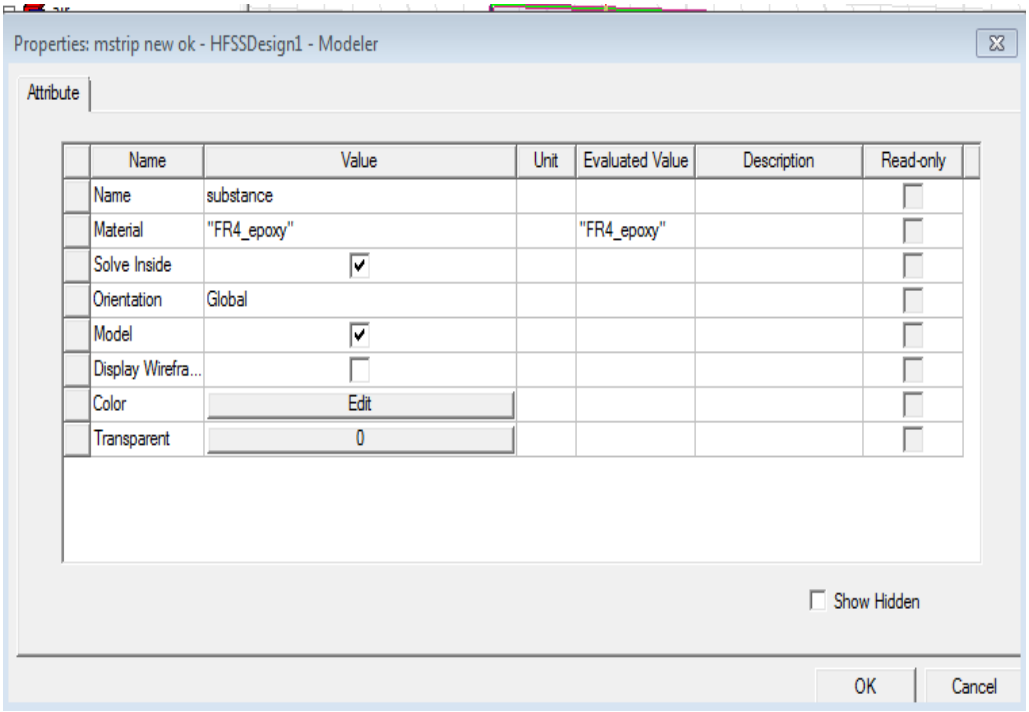
Double click **box 1** (rename **substance**)

Material –

edit fr4 - (4.4)

Select colour

Ok



Double click **create box**

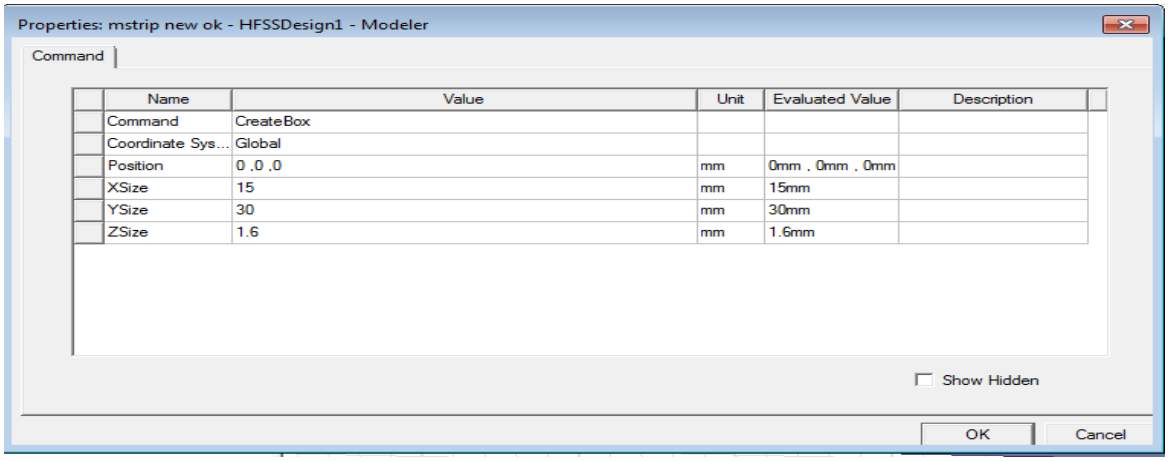
Position 0,0,0

X size- 15mm

Y size- 30mm

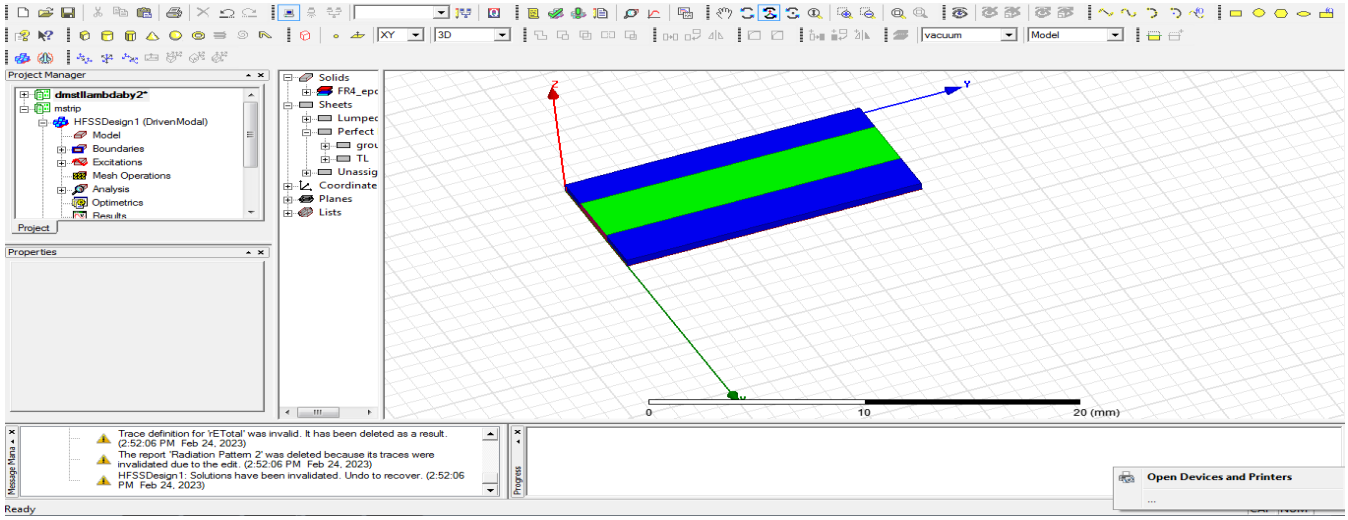
Z size- 1.6mm

Ok



DESIGN TRANSMISSION LINE

Select rectangle design center draw



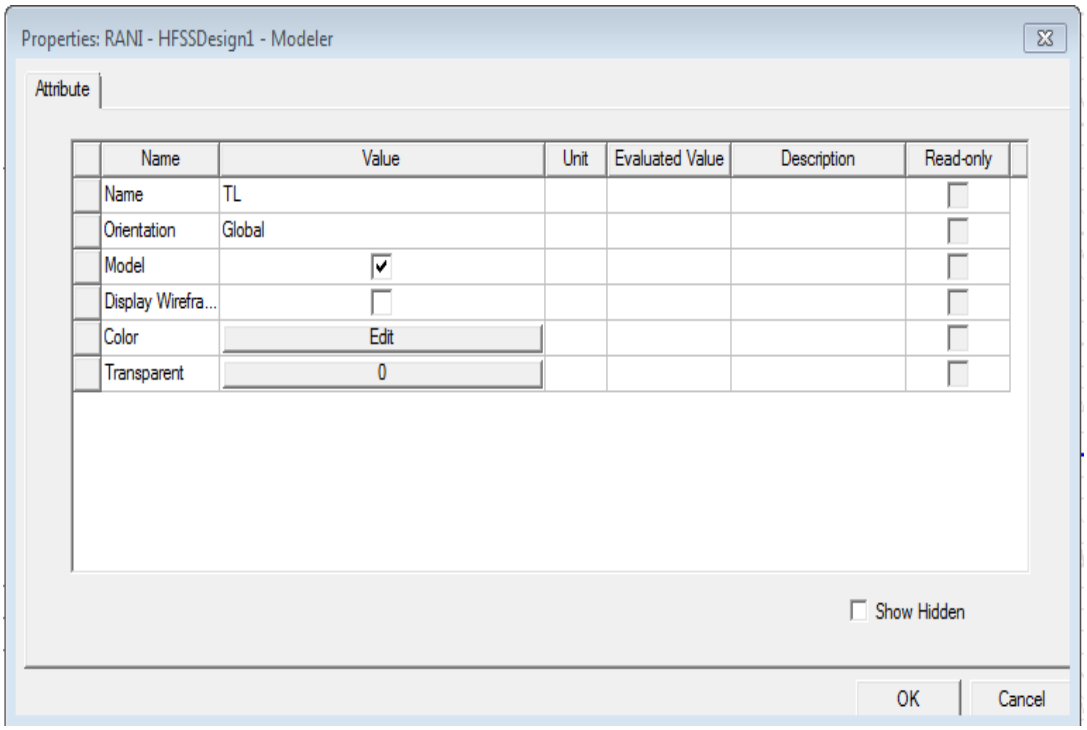
Double click

reactangle1

rename - **TL**

Colour edit

Ok



Double click **create rectangle:**

Position –

6, 0, 1.6

Axis – z

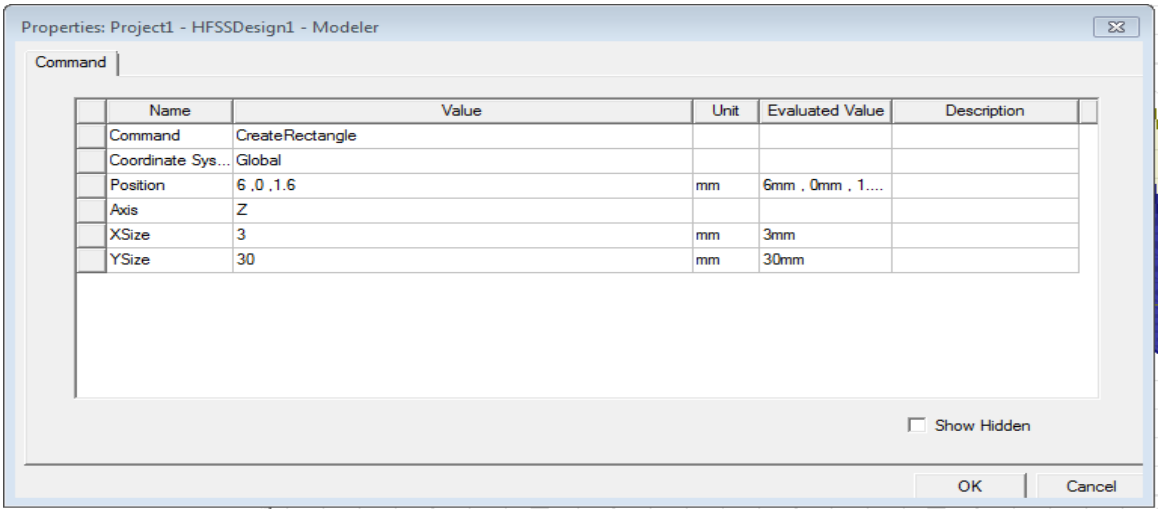
X axis –

3mm

Y size –

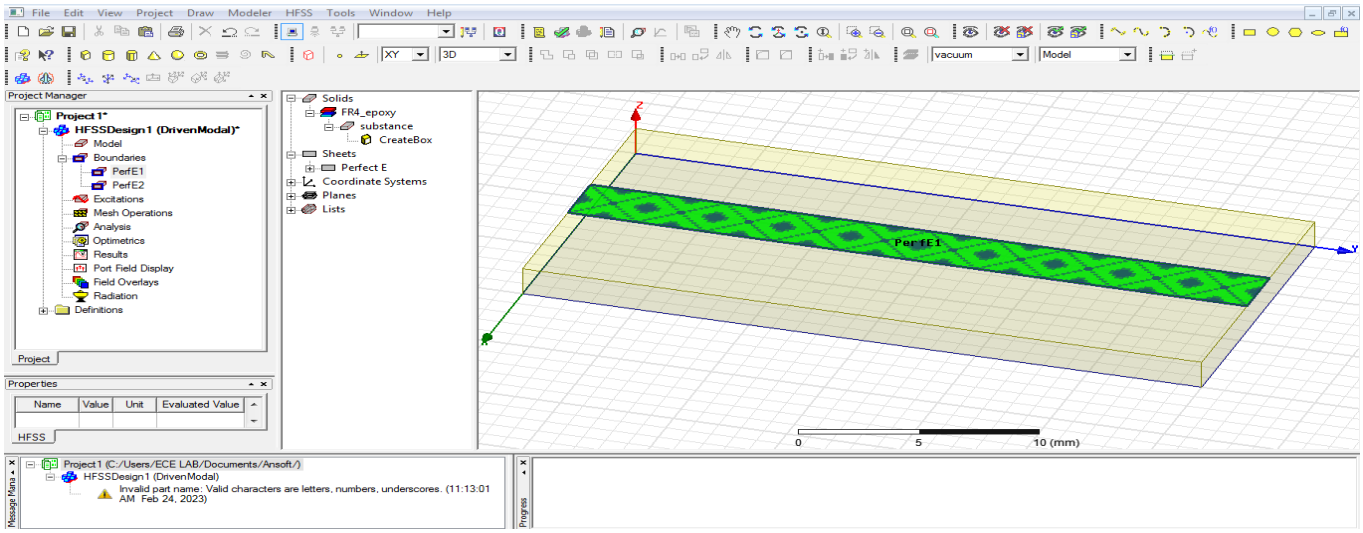
30mm

Ok

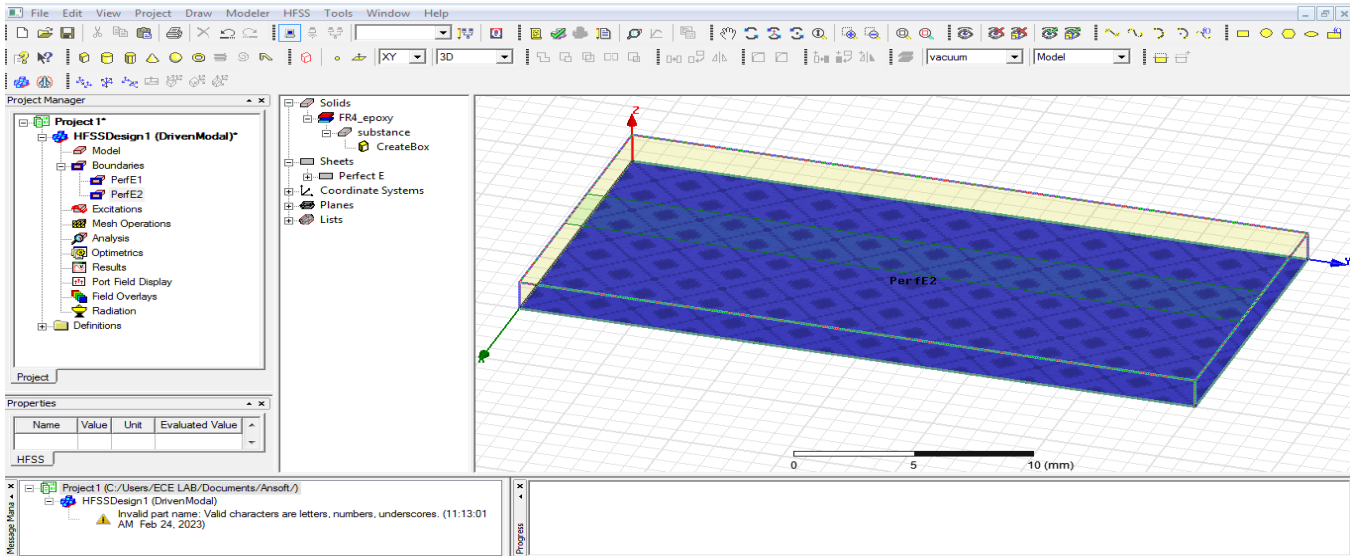


DESIGN PERFECT ELECTRIC BOUNDARY - click **TL**

Right click – assign boundary-perfect E-CLICK-name: **perfE1** - ok



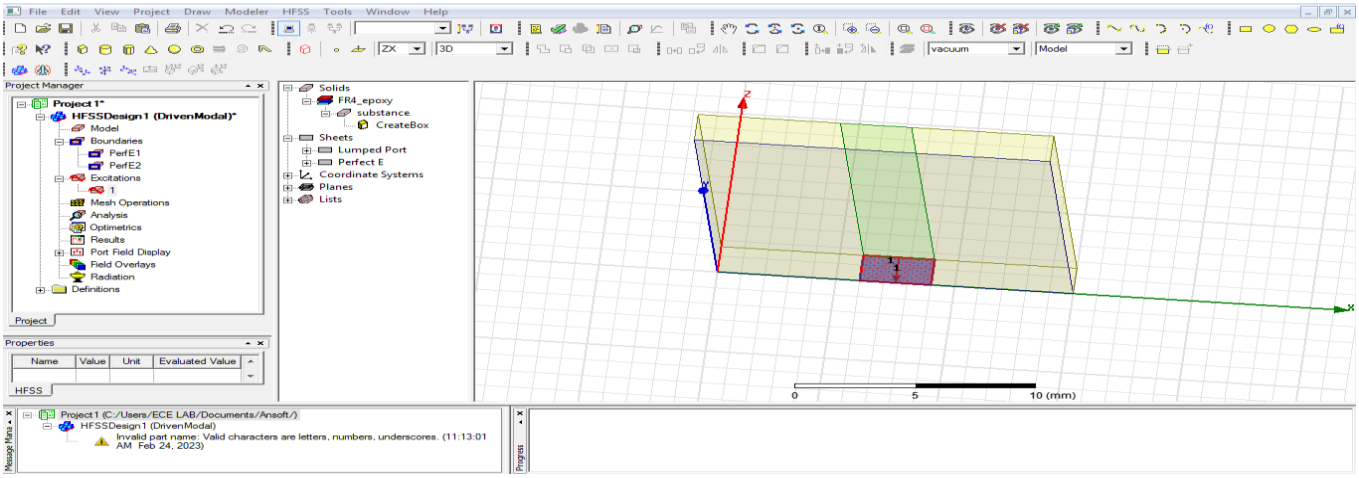
Click ground plan - right click-assign boundary-perfect E – click - name: **perfE2** ok



Design **ports** : (1&2)

Change ZX

Select draw rectangle design transmission line **port1**



Double click **rectangle1**

Rename **port1** ok

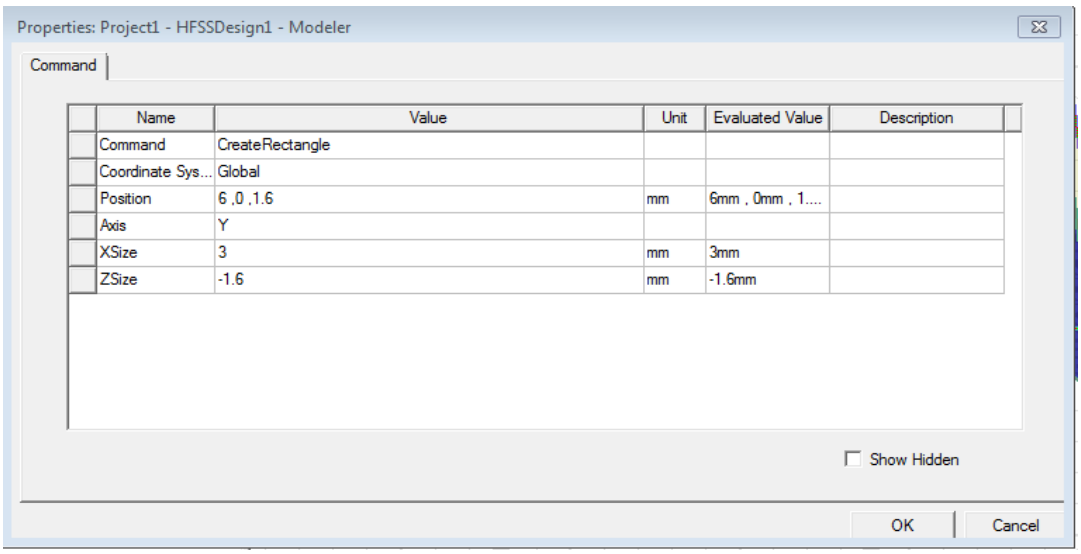
Double click **create rectangle**

Position-(6,0,1.6)

Axis-Y

X size-3mm

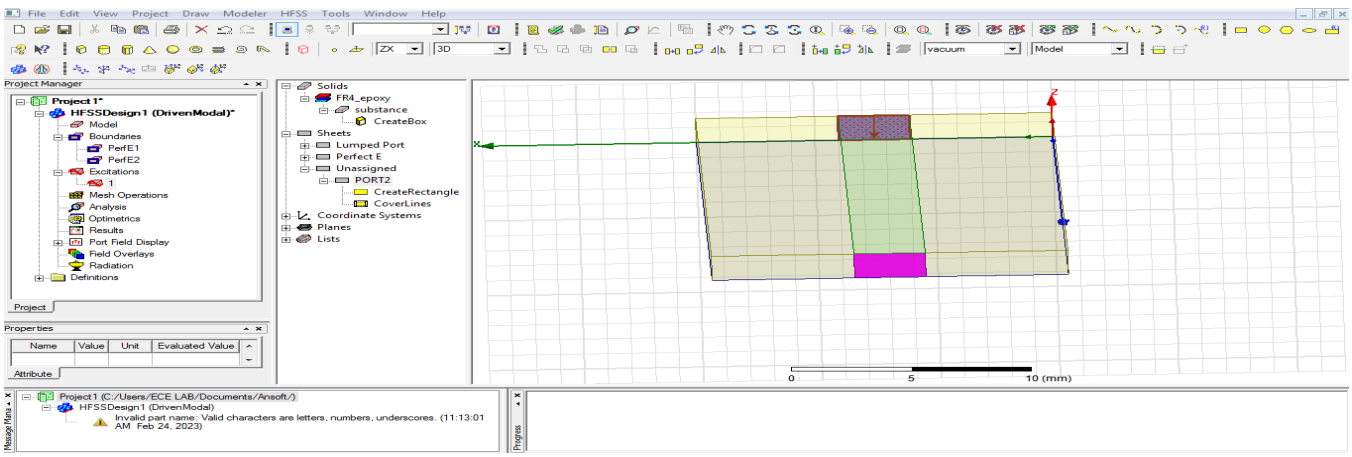
Z size-(-1.6mm) ok



Select port1-right click assign excitation-lumped port-click-name1-resistance 50 ohms-next select none new line –draw a line-defined-next-full port impedance 50 ohms finish

Design port2

Take rectangle - Select draw rectangle design transmission line **port2**



Rename rectangle2

double click port2 ok

Crate rectangle double click

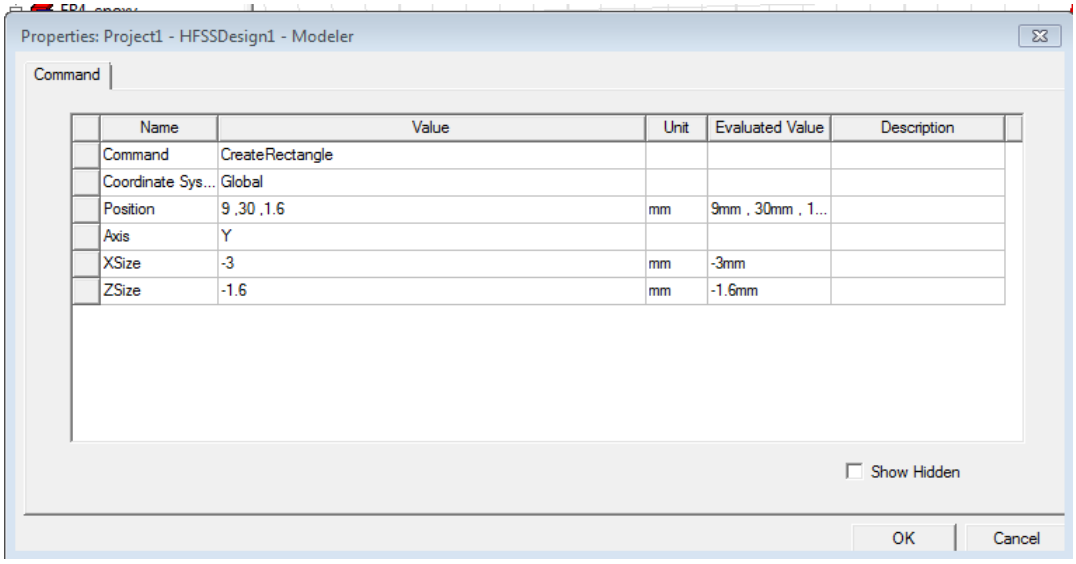
Position - (9,30,1.6)

Axis - Y

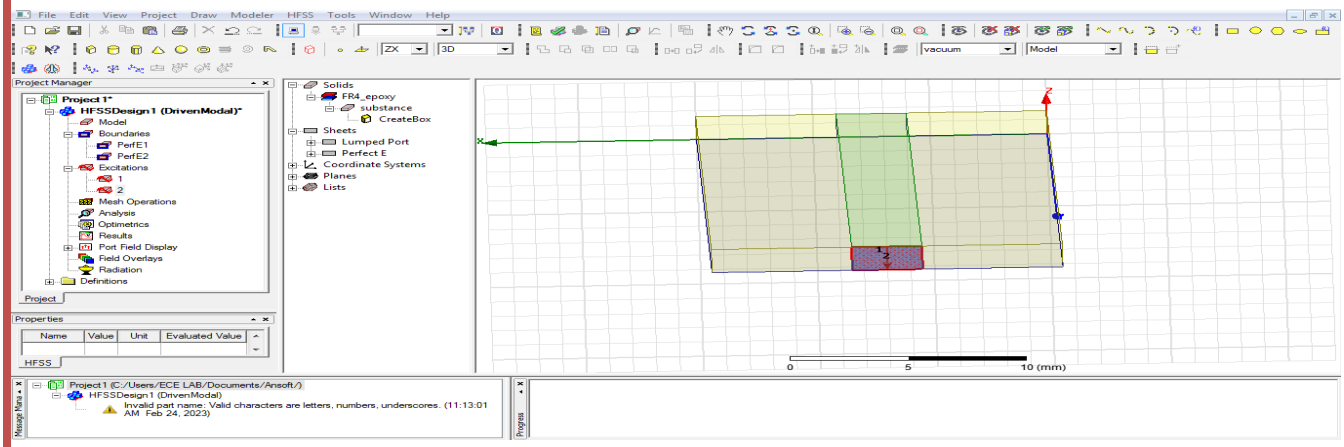
X size (3mm)

Z size (1.6mm)

ok



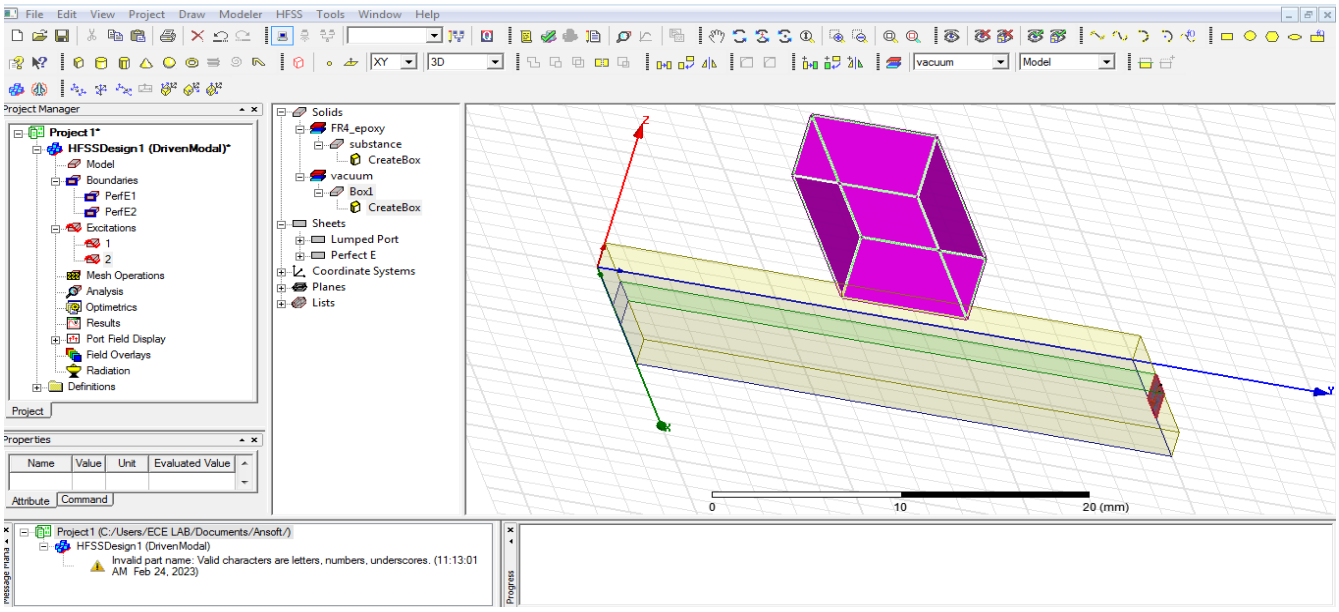
Select port2-right click assign excitation-lumped port-click-name 2-resistance - 50 ohms-next select none new line –draw a line-defined-next-full port impedance - 50 ohms – finish



Select **XY**

Design **radiation boundary**

Draw **box**



double click **box1**

rename **radiation boundary**

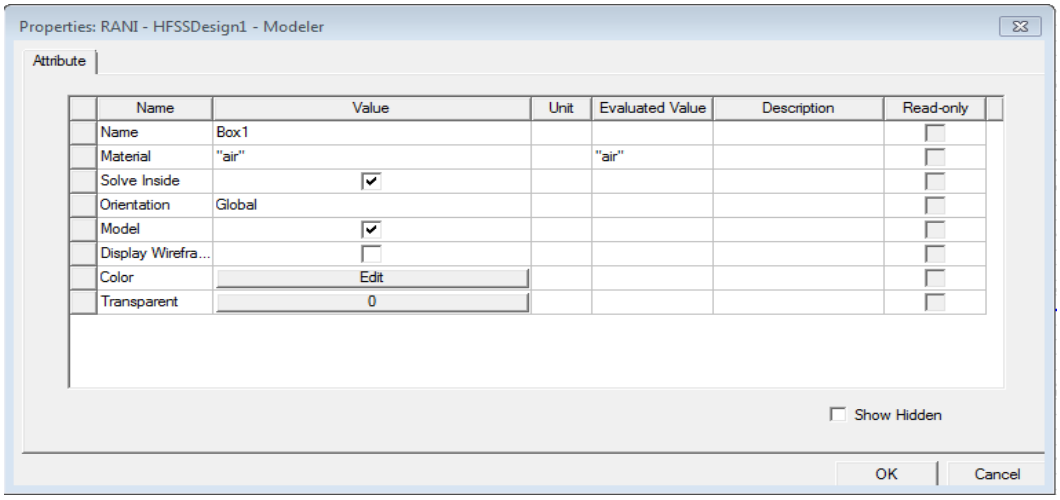
Material - edit **air**

ok

Colour edit

ok

Double click **create box**



Position-

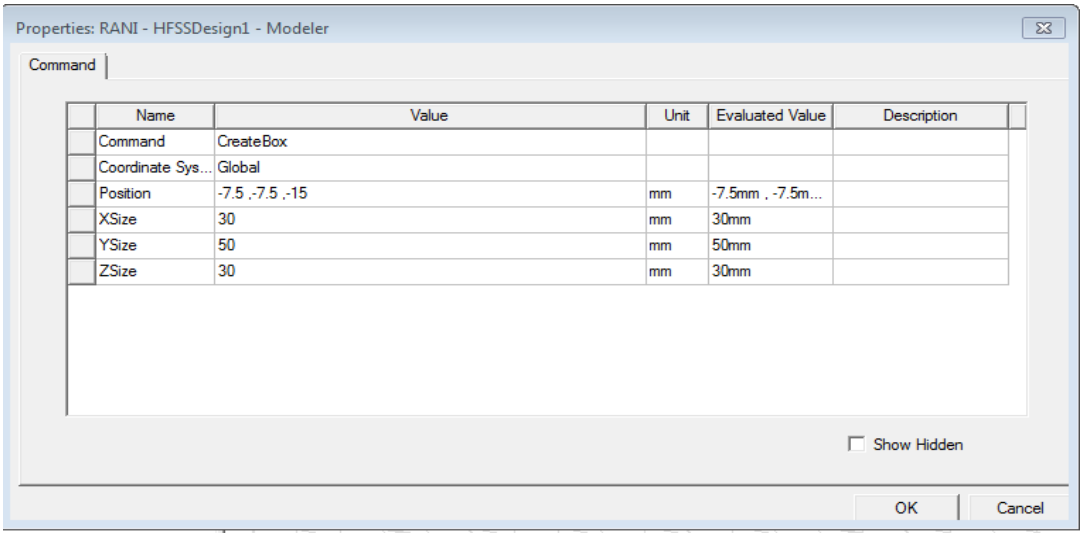
(-7.5,-7.5,-15)

X size-30mm

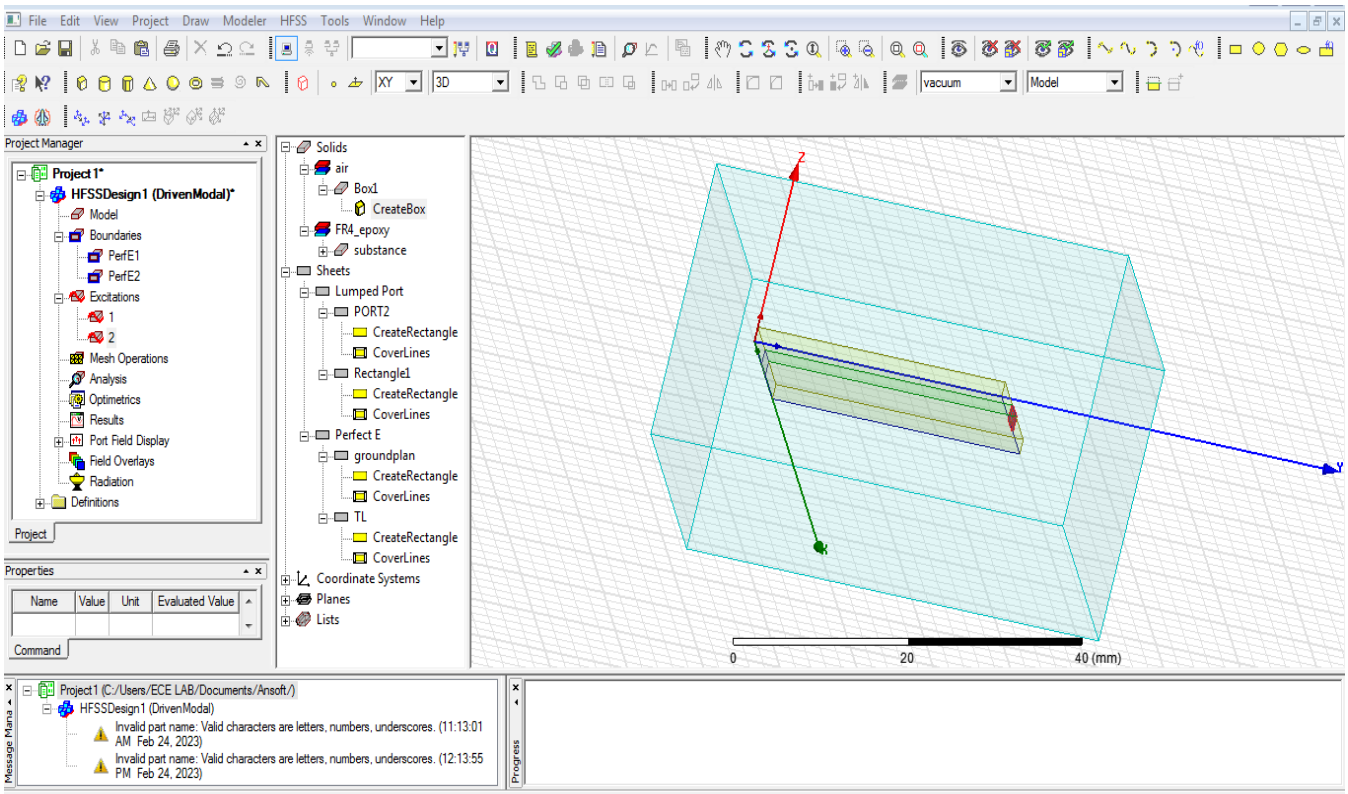
Y size-50mm

Z size-30mm

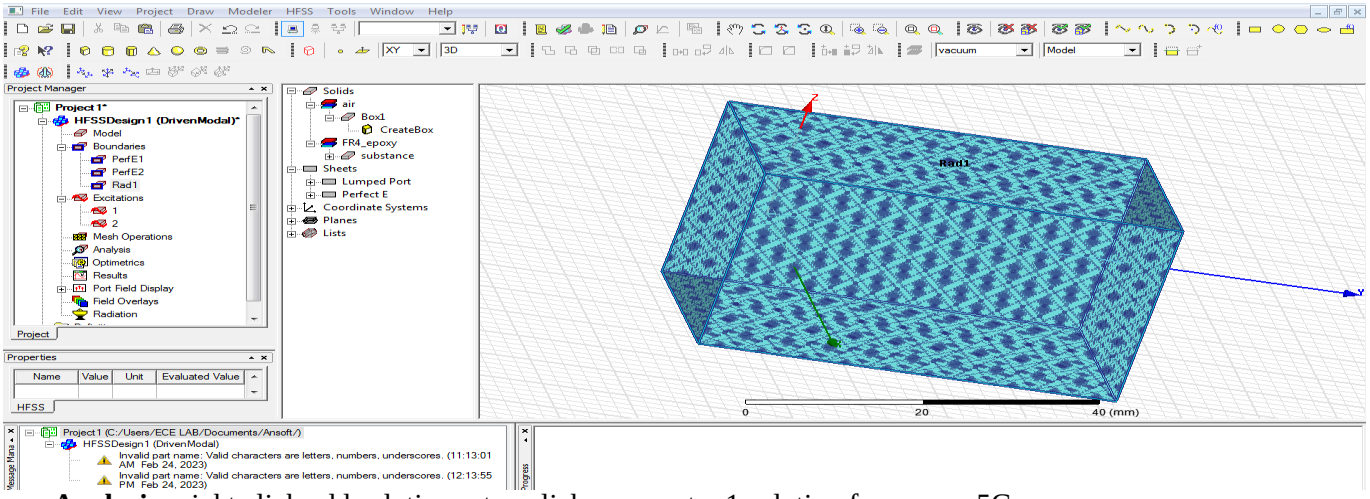
ok



Fit all



Click radiation boundary-right click-assign boundary –radiation click name **rad1** ok



Analysis –right click add solution setup click name setup1 solution frequency-5GHZ

Maximum number of passes -12

Maximum Delta S-0.02 ok

Analysis right click- **setup1** right click - add frequency sweep

Sweep type : fast

Type : linear count

Start freq:1GHZ

Stop freq:10GHZ

Count:101GHZ

Click display-see all frequencies ok

Result analysis(error checking)

Click double click **validity**

HFSS Design- design setting

3D model

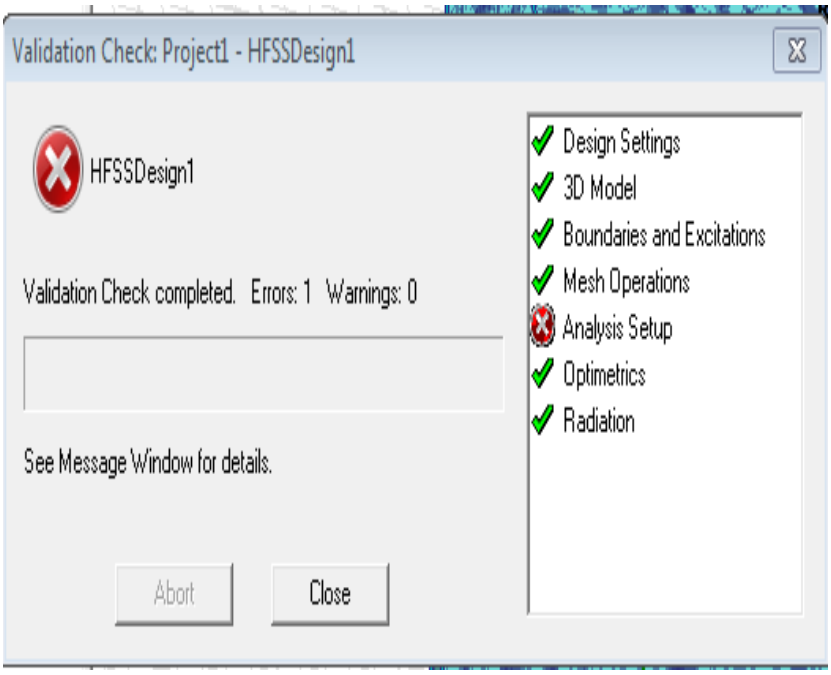
Boundaries and excitations

Mesh operation

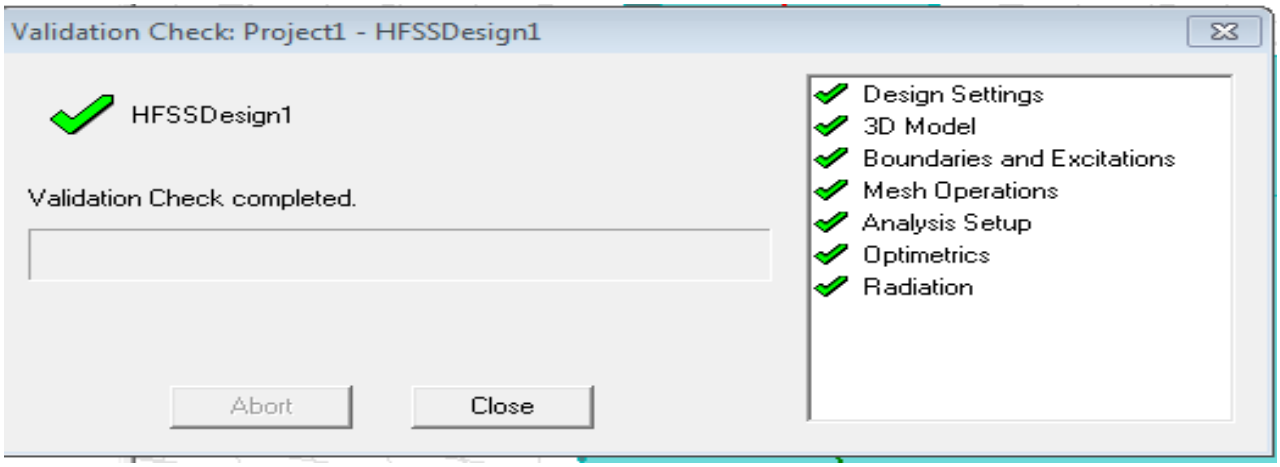
Analysis setup

Optimetrics

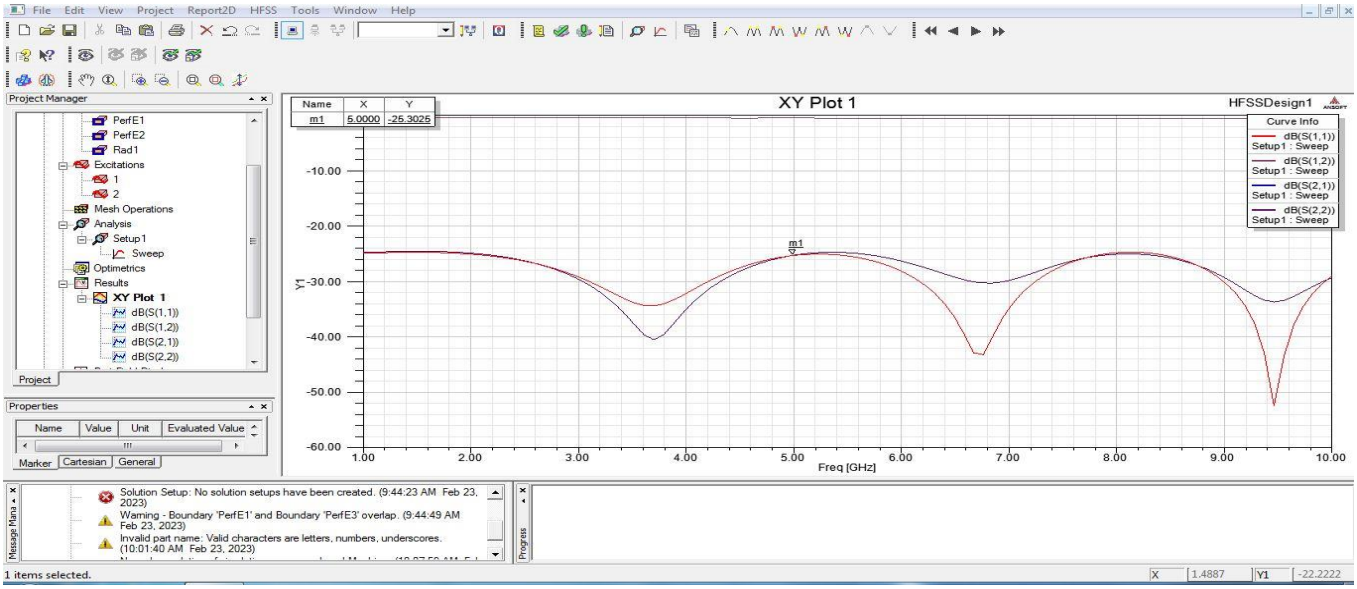
Radiation



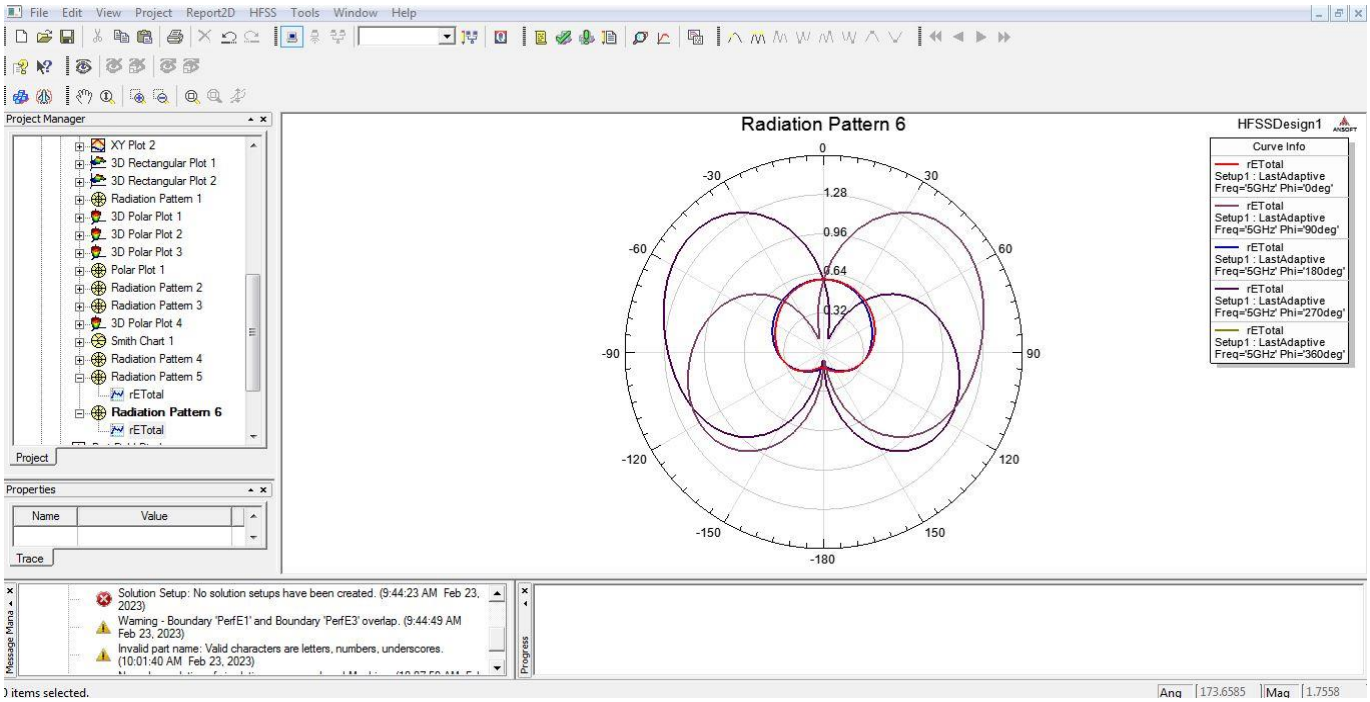
Next **analysis all**-any error rectify-ok



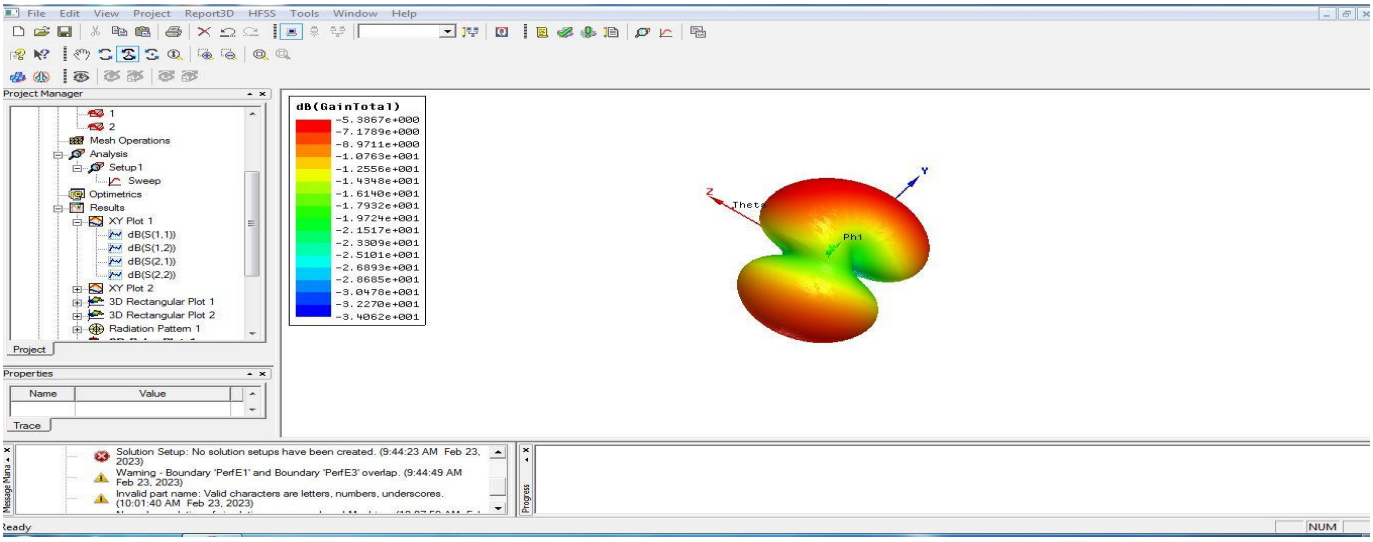
<u>Model Waveforms</u>: 5 GHZ (S Parameter)



radiation pattern graphs



3d graphs



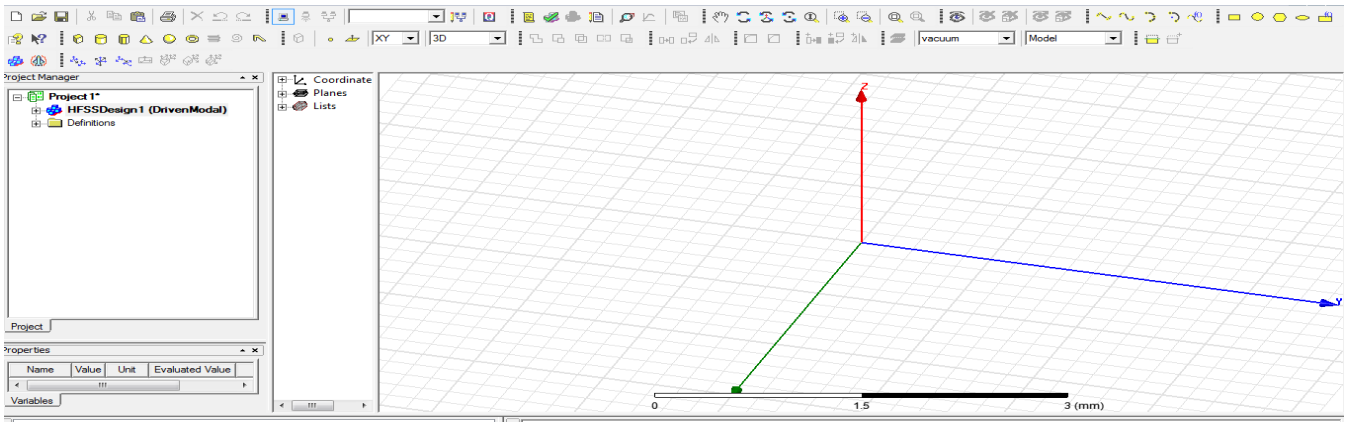
vswr graph

Calculations:

b). Aim: Design of λ/4 micro strip transmission line.

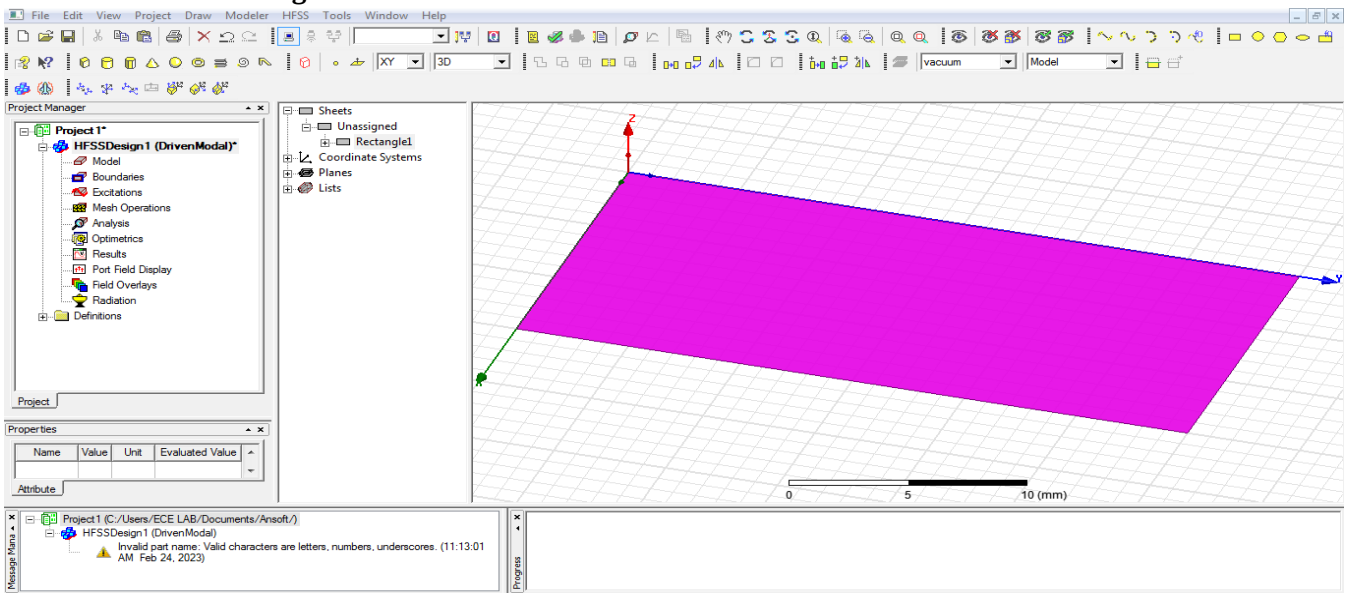
<u>Design</u>: MICROSTRIP TRANSMISSION LINE Λ/4 USING HFSS (5 GHZ)

Open **HFSS** project-click project , open - project insert **HFSS** design.



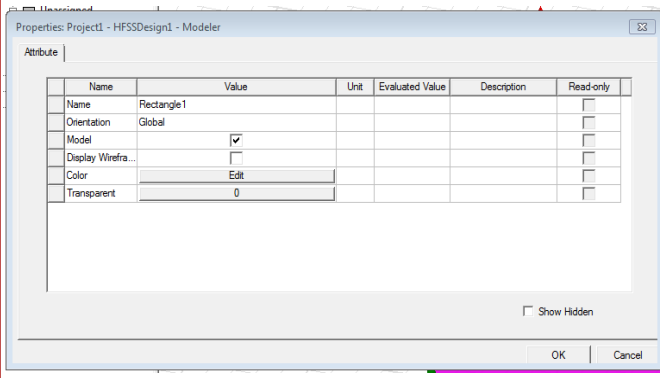
DESIGN GROUN PLAN:

Select click **rectangular draw**

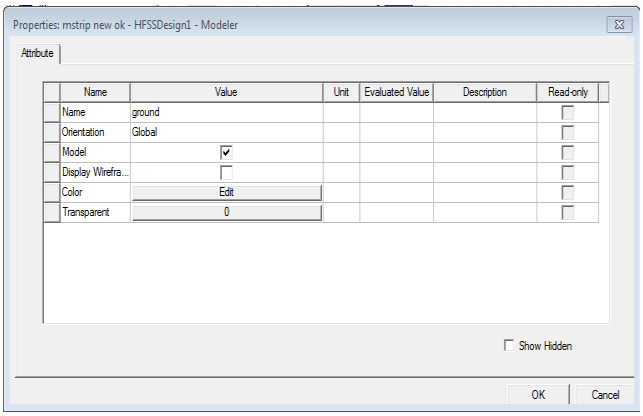


Double click **rectangle1**

Name : [ground plan] , Colour : edit [as your wish]



before name change



after name change

Double click

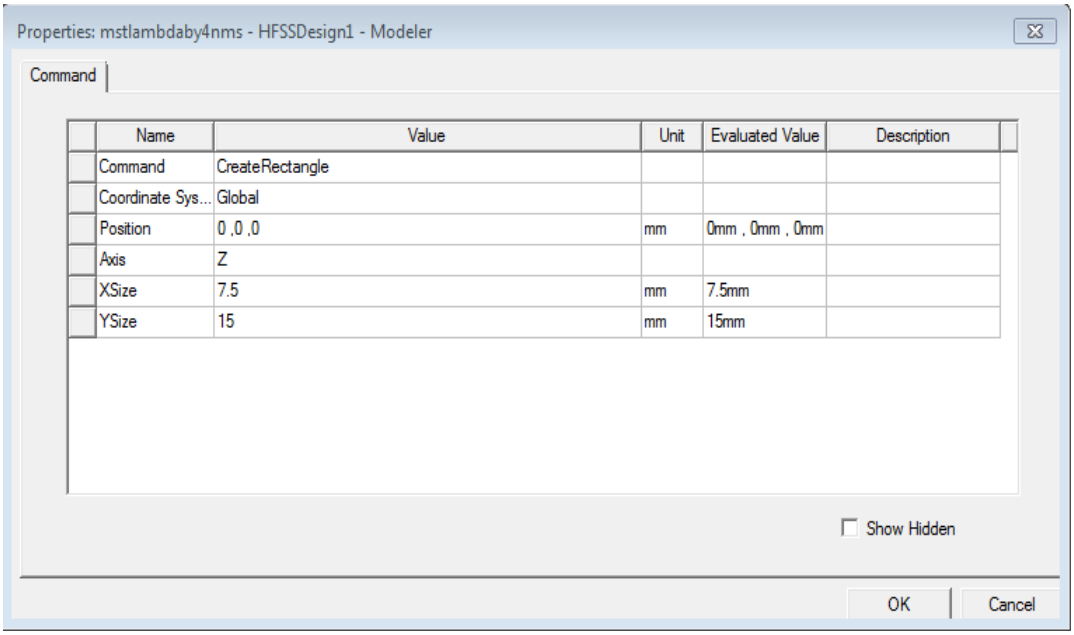
Create rectangle:

Position : 0,0,0

Axis : z

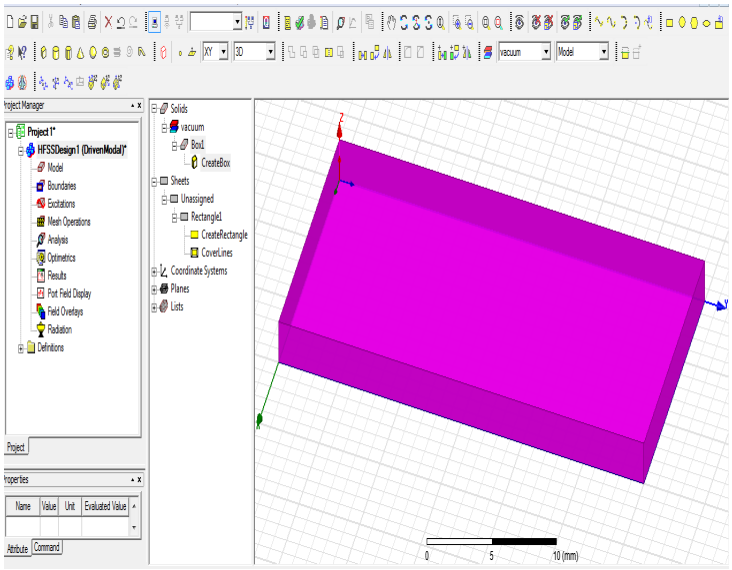
X size : 7.5 mm

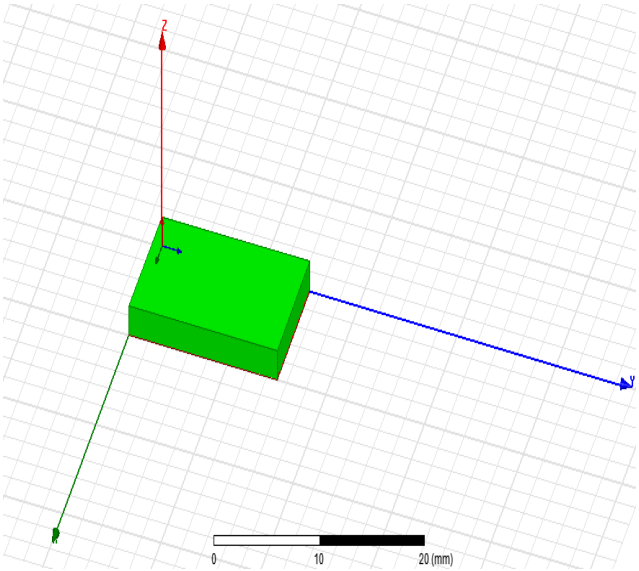
Y size : 15 mm



Ok **Select fit all the contents in the view**

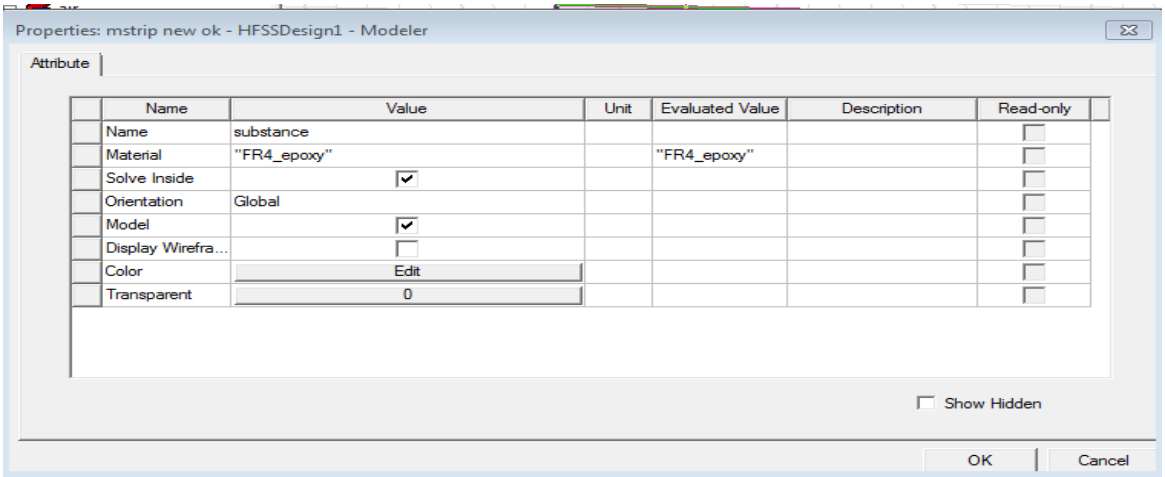
CREATE SUBSTANCE : Select draw the **BOX** and design





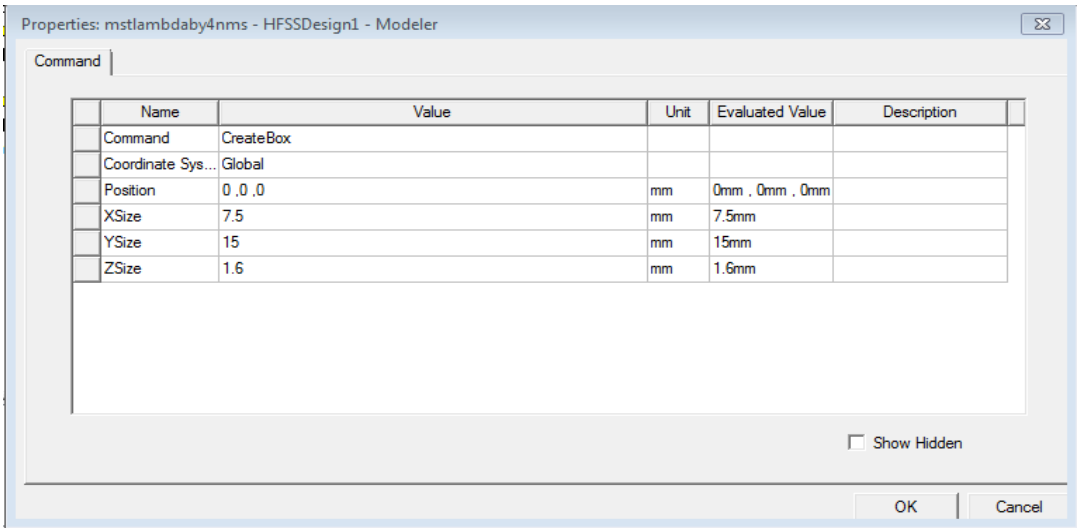
Double click **box 1** (rename **substance**)

Select-colour, ok



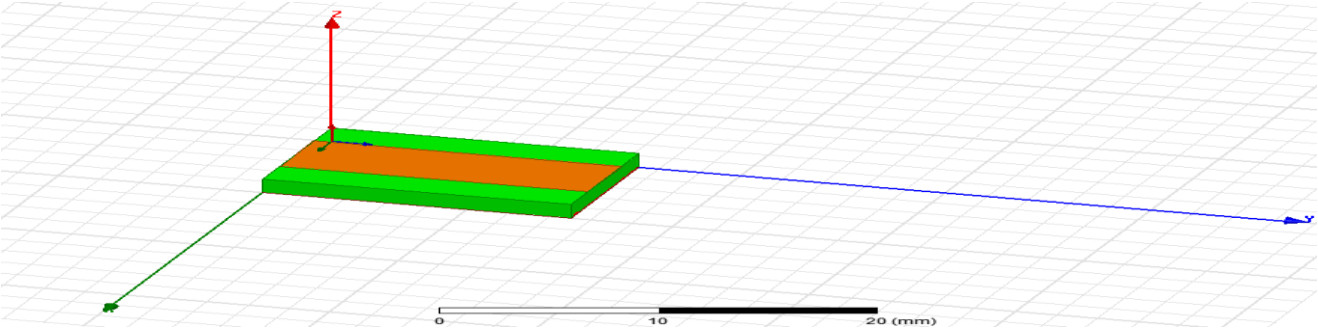
Double click **create box**
Material – edit fr4 - (4.4)
Position- 0,0,0
X size- 7.5mm
Y size- 15mm

Z size- 1.6mm
Ok

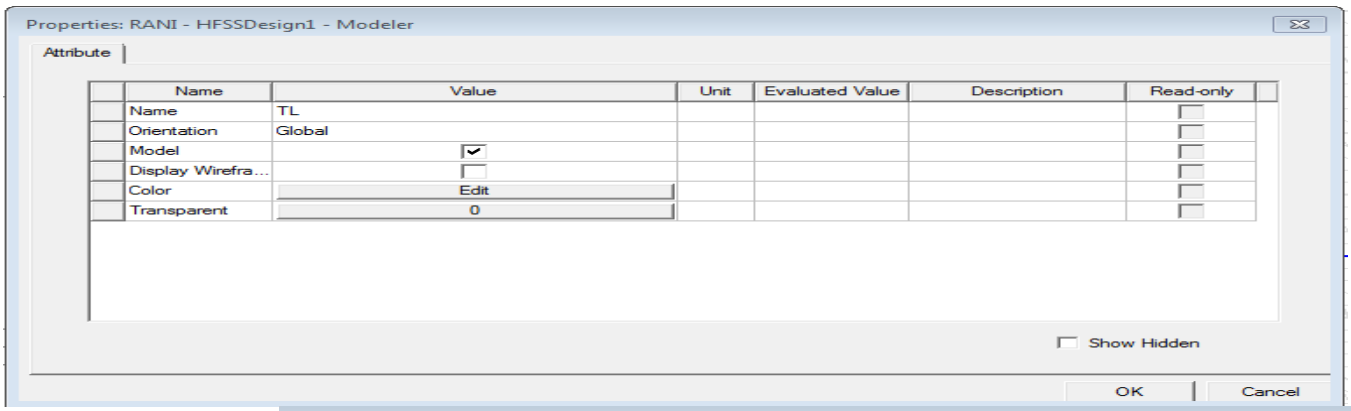


DESIGN TRANSMISSION LINE

Select rectangle design center draw



Double click **reactangle1**
rename – **TL** Colour edit Ok



Double click

Create rectangle:
Position:

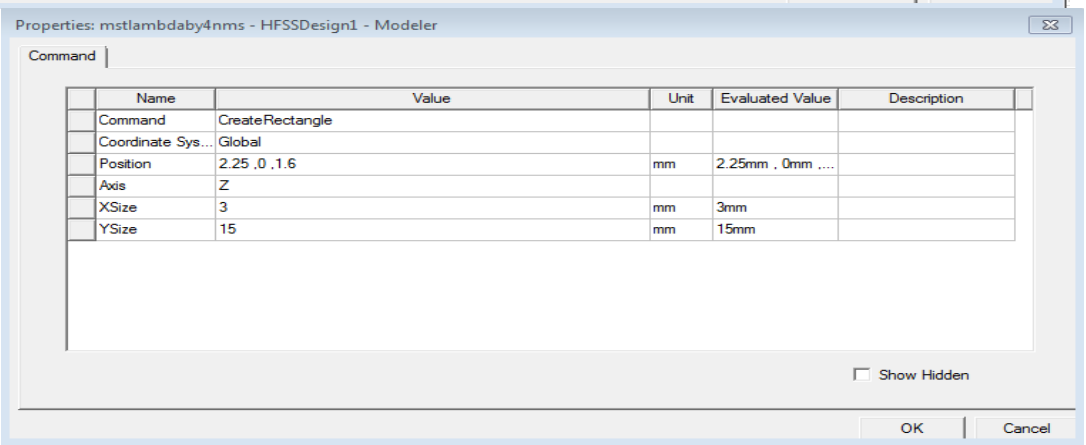
2.25, 0, 1.6

Axis : z

X axis: 3mm

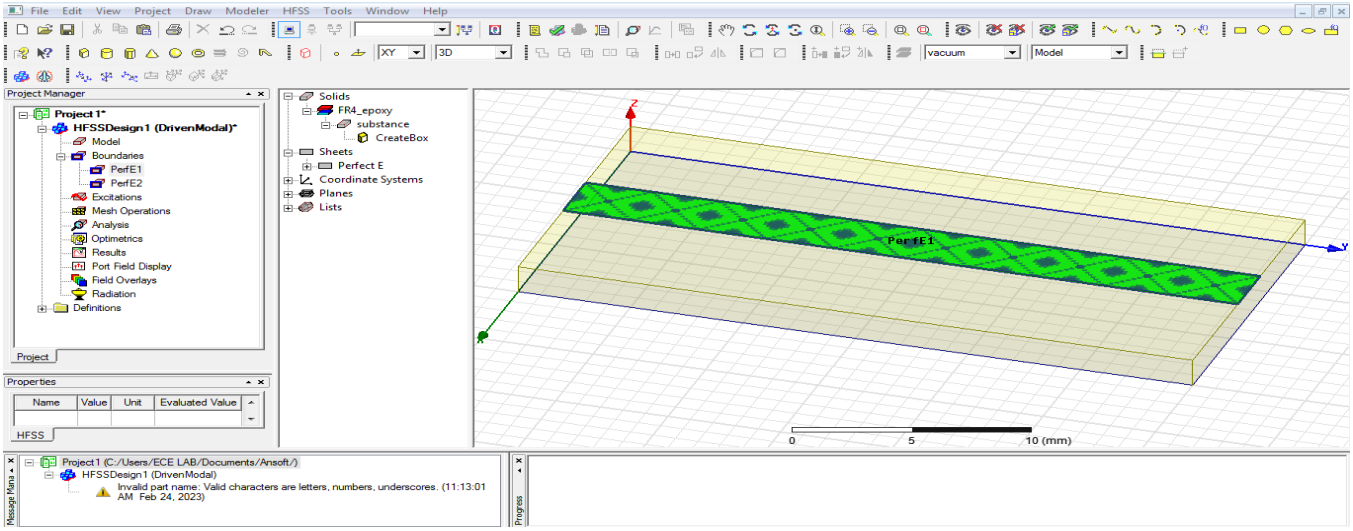
Y size :

15mm Ok

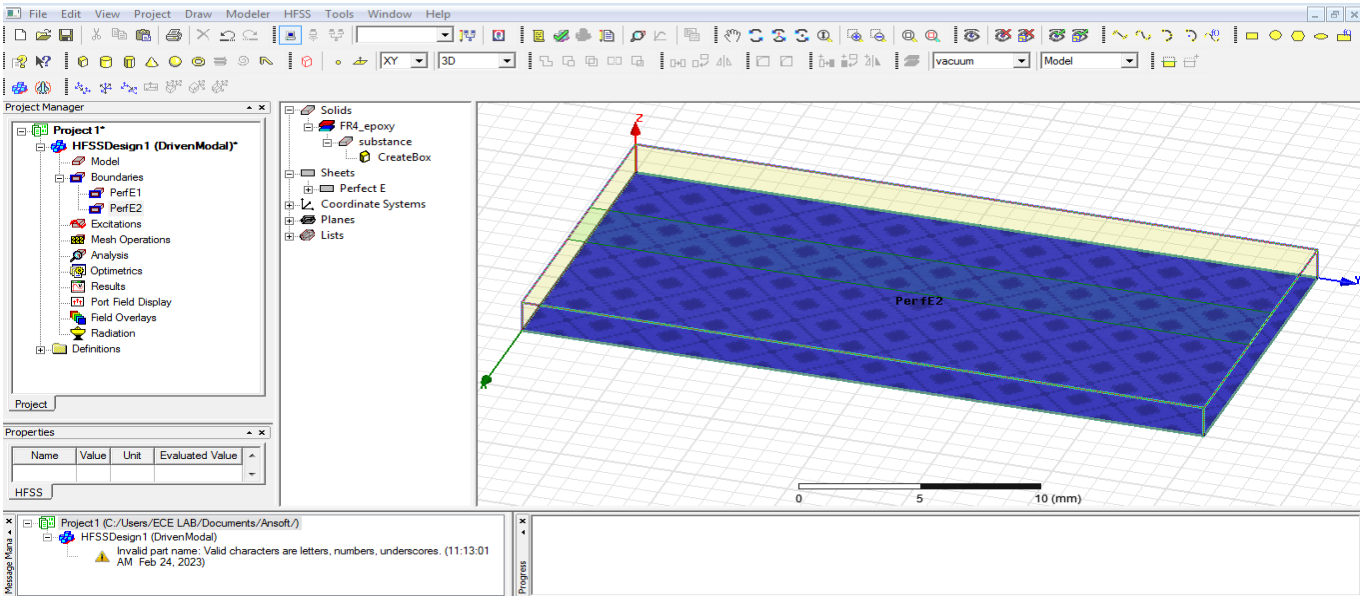


DESIGN PERFECT ELECTRIC BOUNDARY - click **TL**

Right click – assign boundary-perfect E-CLICK-name: **perfE1** - ok

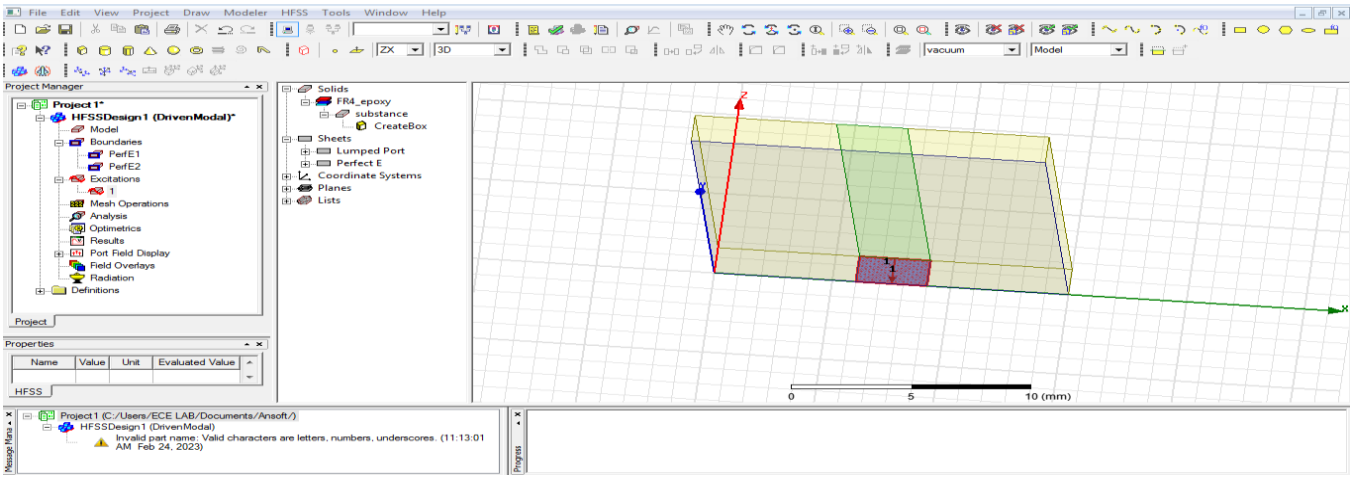


Click ground plan - right click-assign boundary-perfect E – click - name: **perfE2** ok



Design **ports** : (1&2) Change ZX

Select draw rectangle design transmission line **port1**



Double click **rectangle1**
Rename **port1** ok
Double click **create rectangle**
Position-

(2.25, 0, 1.6)

Axis-Y

X size-

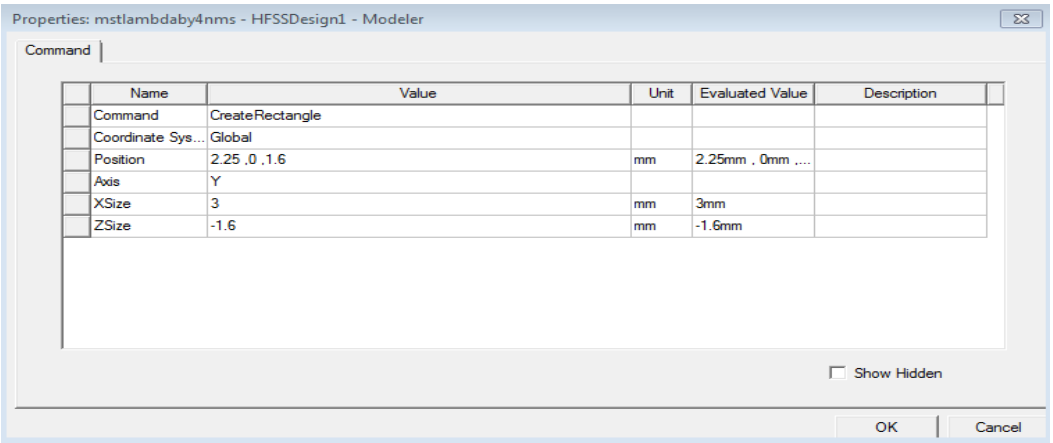
3mm

Z size-

(-1.6mm)

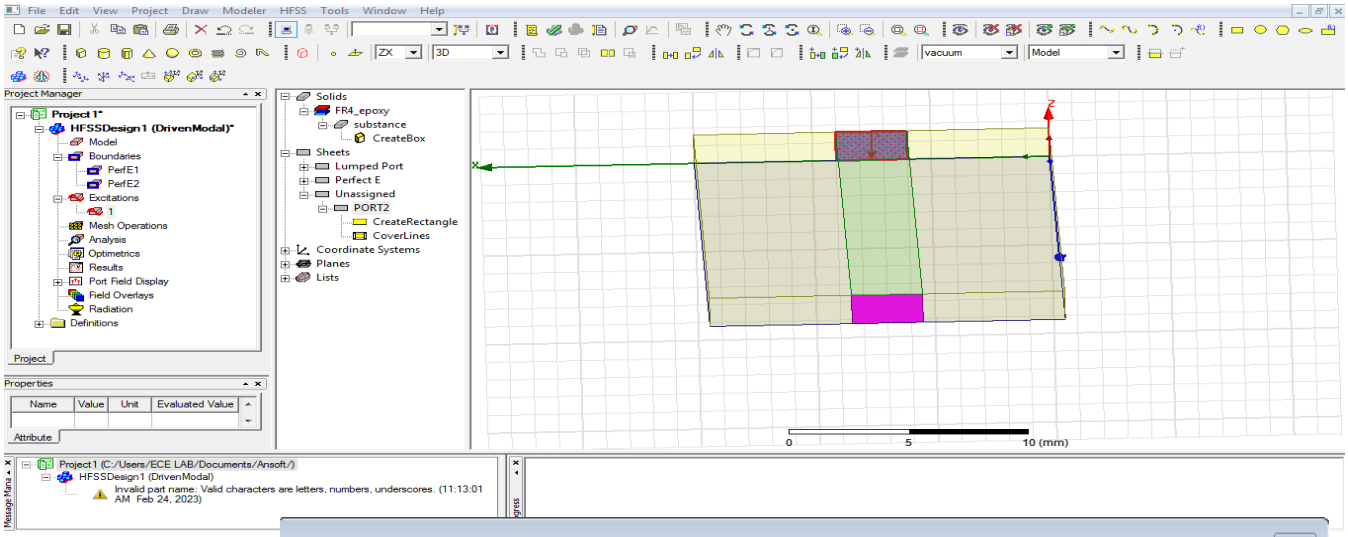
ok

Select port1-right click



assign excitation-lumped port-click-name1-resistance 50 ohms-next, select none new line –draw a line-defined-next-full port impedance 50 ohms finish

Design port2 Take rectangle - Select draw rectangle design transmission line **port2**



Rename **rectangle2**

Double click port2 ok

Crate rectangle double click

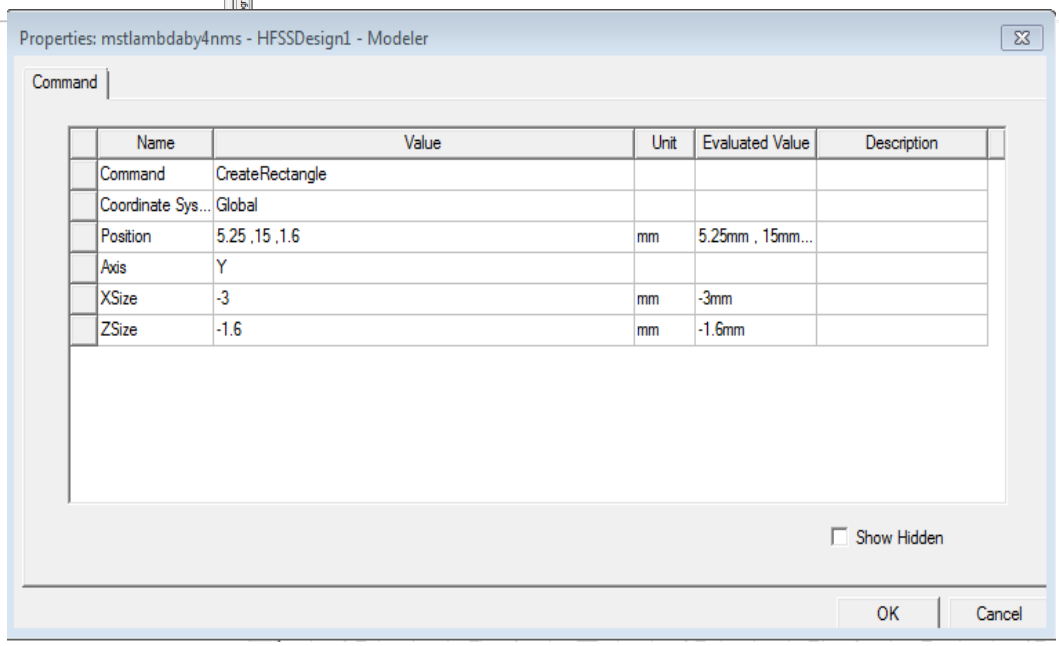
Position - (5.25, 15, -1.6)

Axis - Y

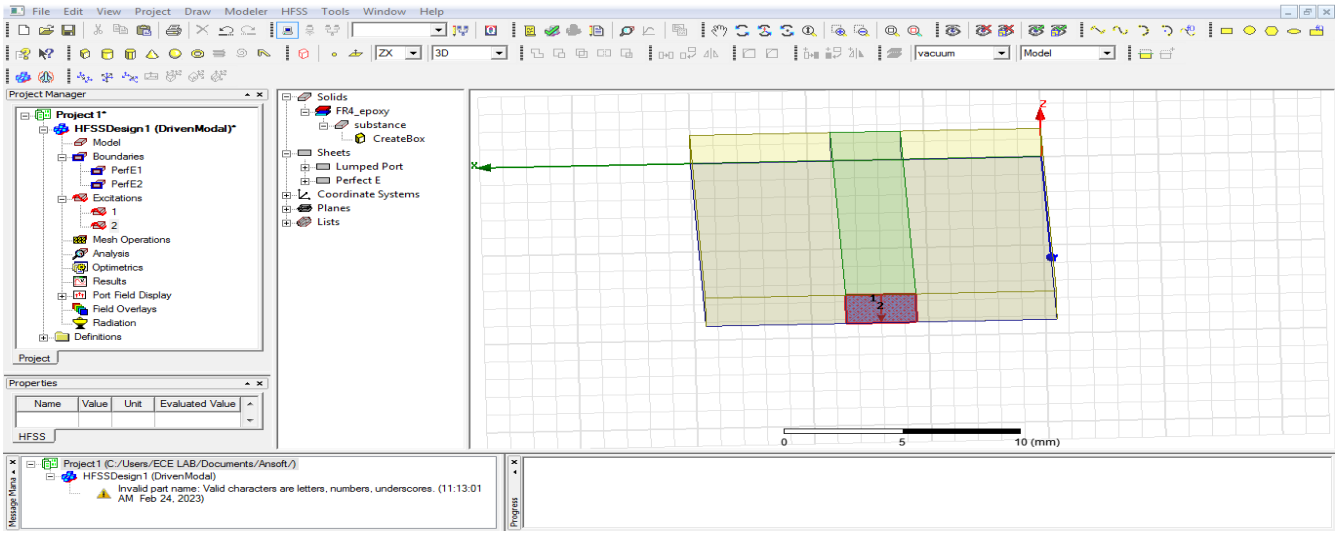
X size (-3mm)

Y size (-1.6mm)

ok



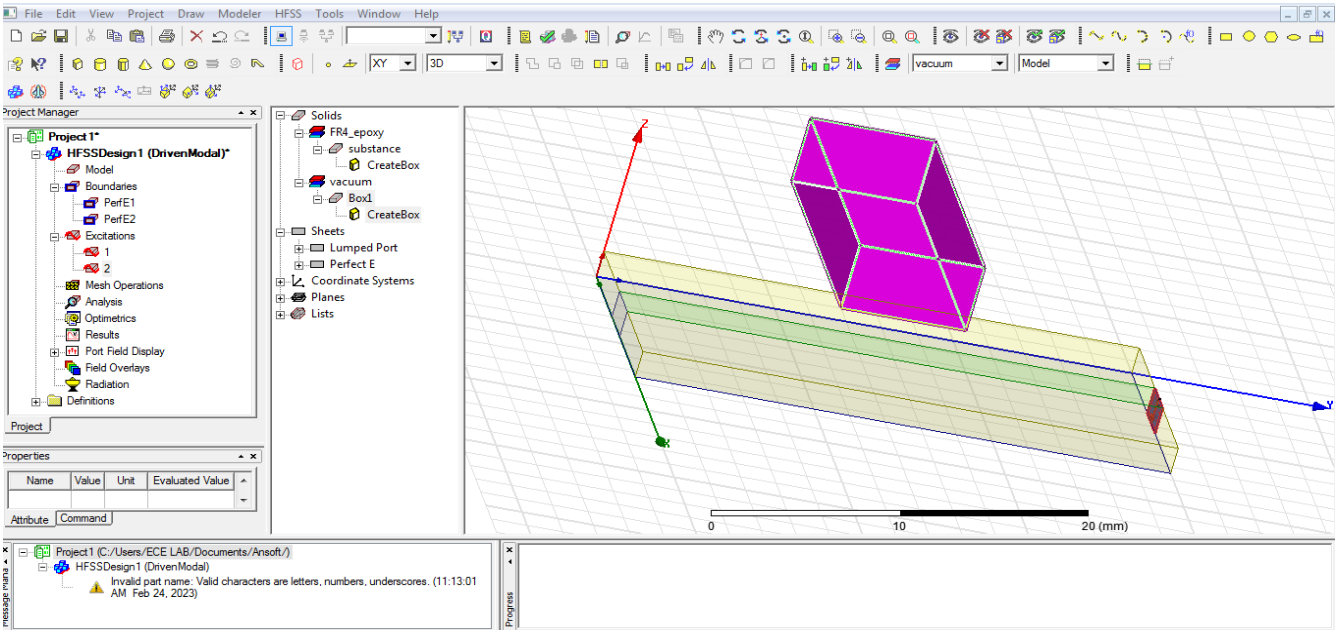
Select port2-right click assign excitation-lumped port-click-name1-resistance - 50 ohms-next select none new line –draw a line-defined-next-full port impedance - 50 ohms - finish



Select **XY**

Design **radiation boundary**

Draw **box**



double click **box1**

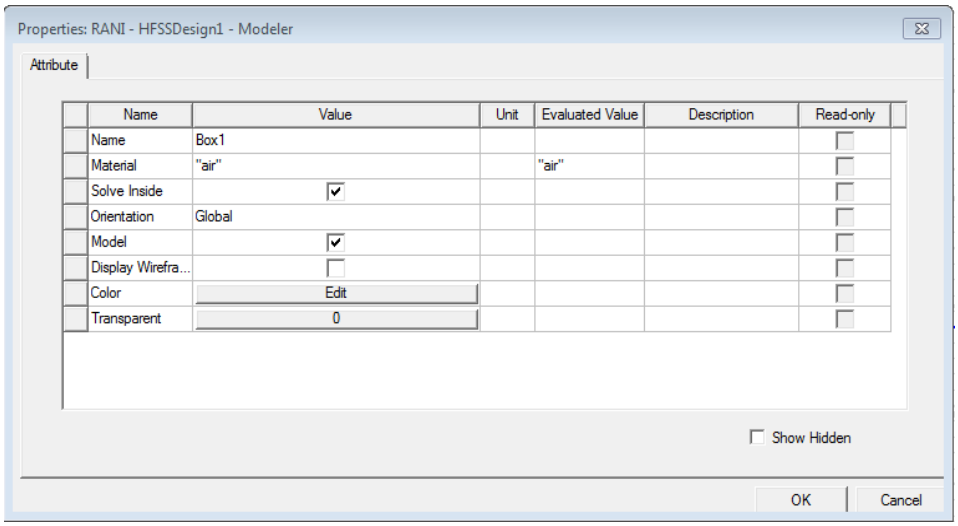
rename **radiation boundary**

Material - edit - **air**

ok

Colour edit

ok



Double click

create box

Position- (-3,-3,-3)

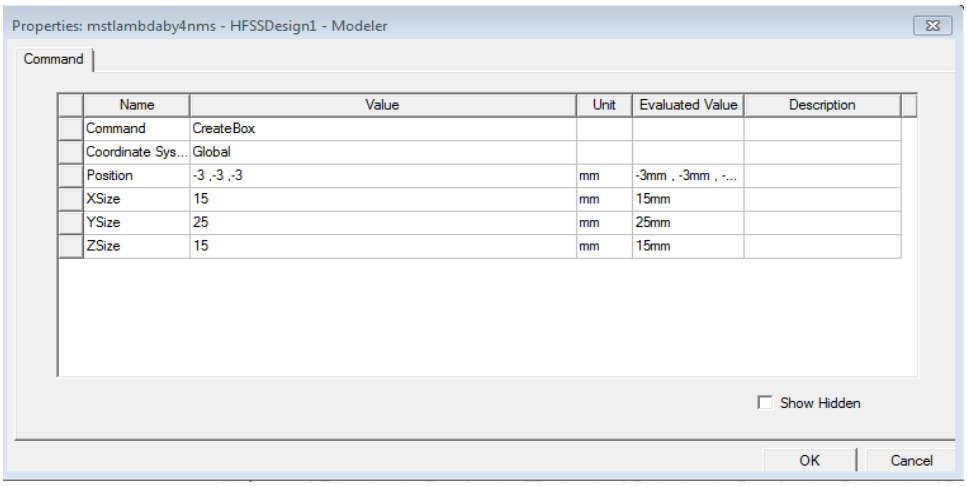
X size-15mm

Y size-25mm

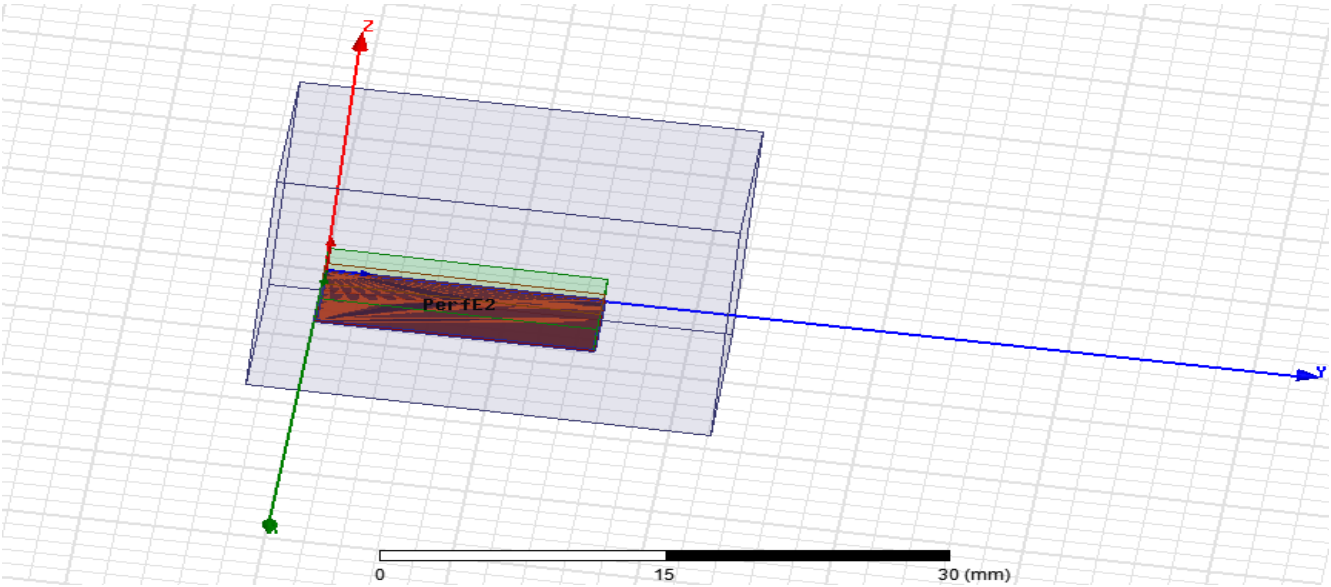
Z size-15mm

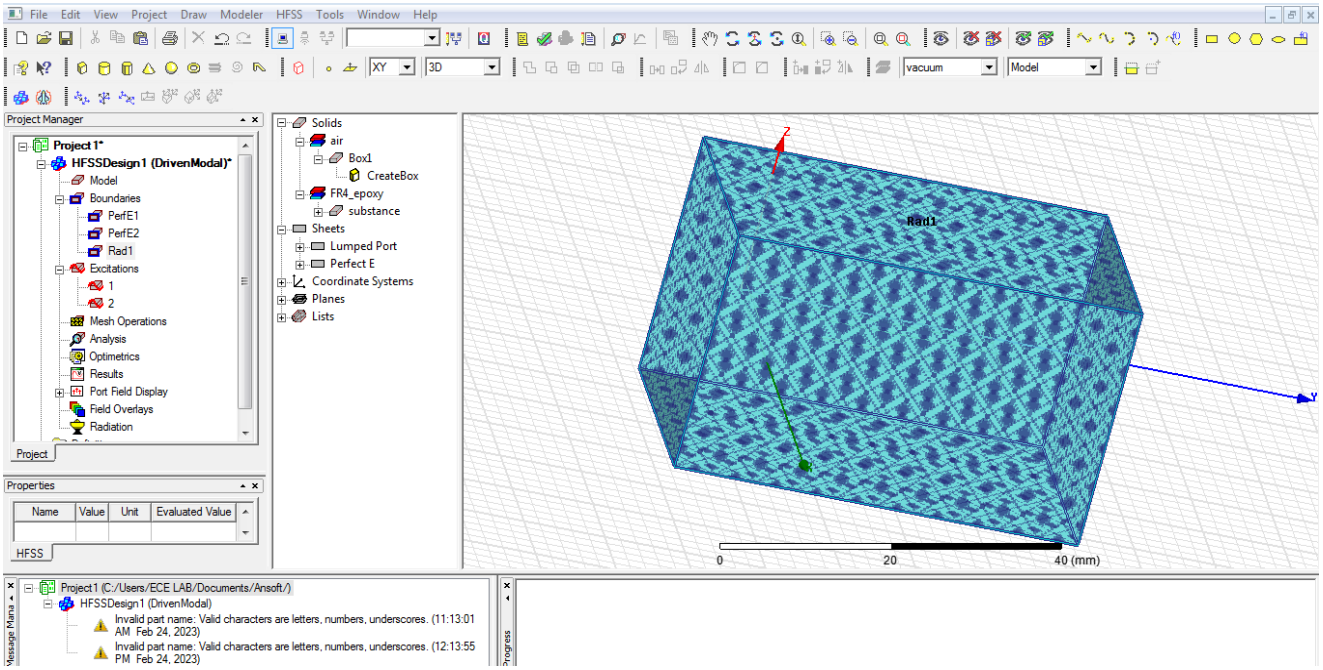
ok

Fit all



Click radiation boundary-right click-assign boundary –radiation click name **rad1** ok





Analysis –right click add solution setup click name setup1 solution frequency-5GHZ

Maximum number of passes -12

Maximum Delta S - 0.02 ok

Analysis right click- **setup1** right click - add frequency sweep

Sweep type : fast

Type : linear count

Start freq:1GHZ

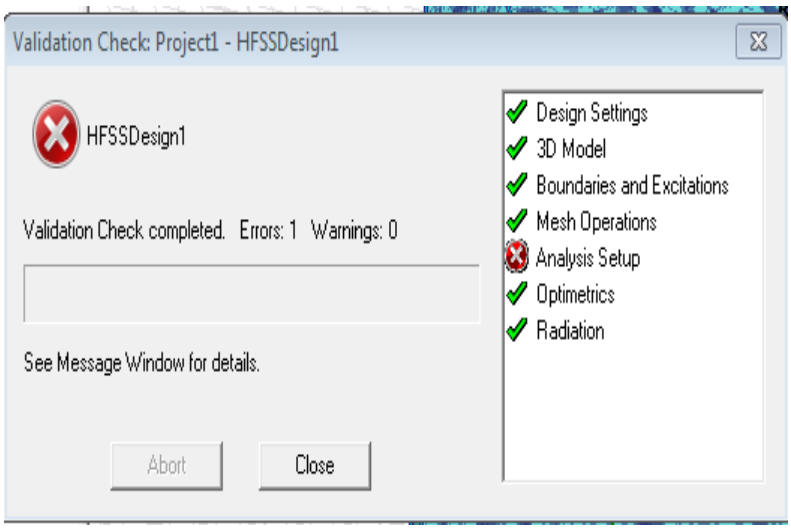
Stop freq:10GHZ

Count:101GHZ

Click display-see all frequencies ok

Result analysis(error checking)

Click double click **validity**



HFSS Design- design setting

3D model

Boundaries and excitations

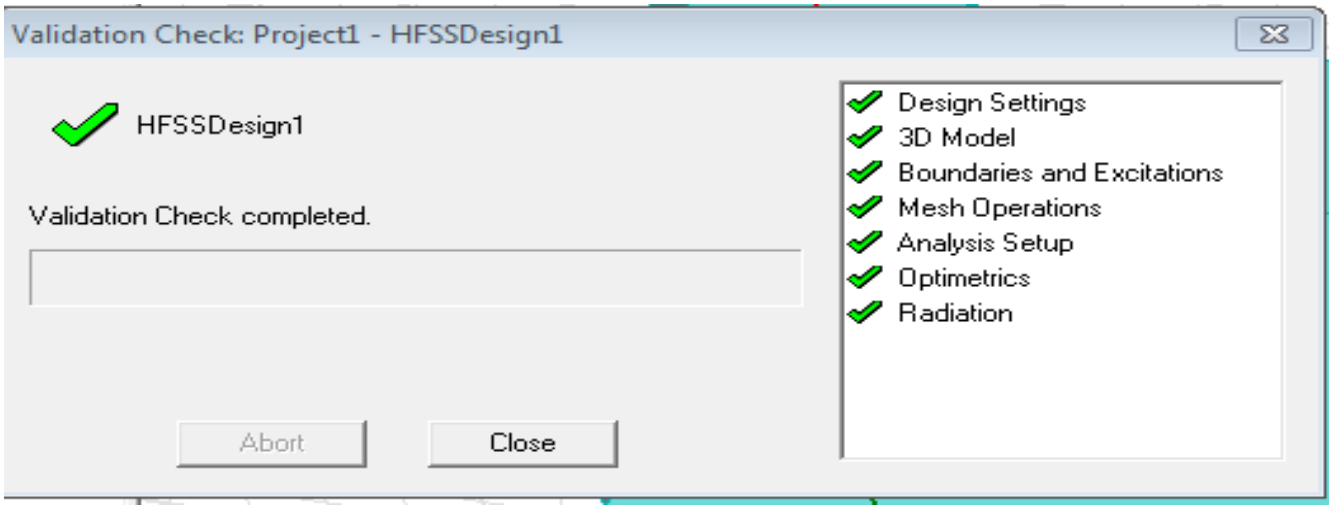
Mesh operation

Analysis setup

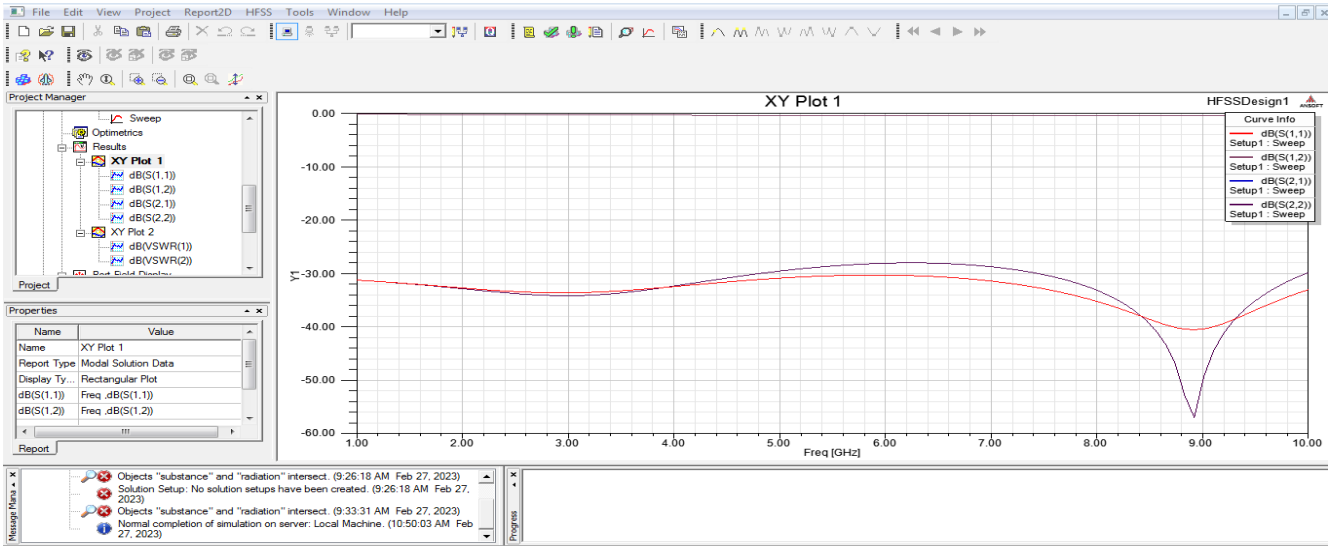
Optimetrics

Radiation

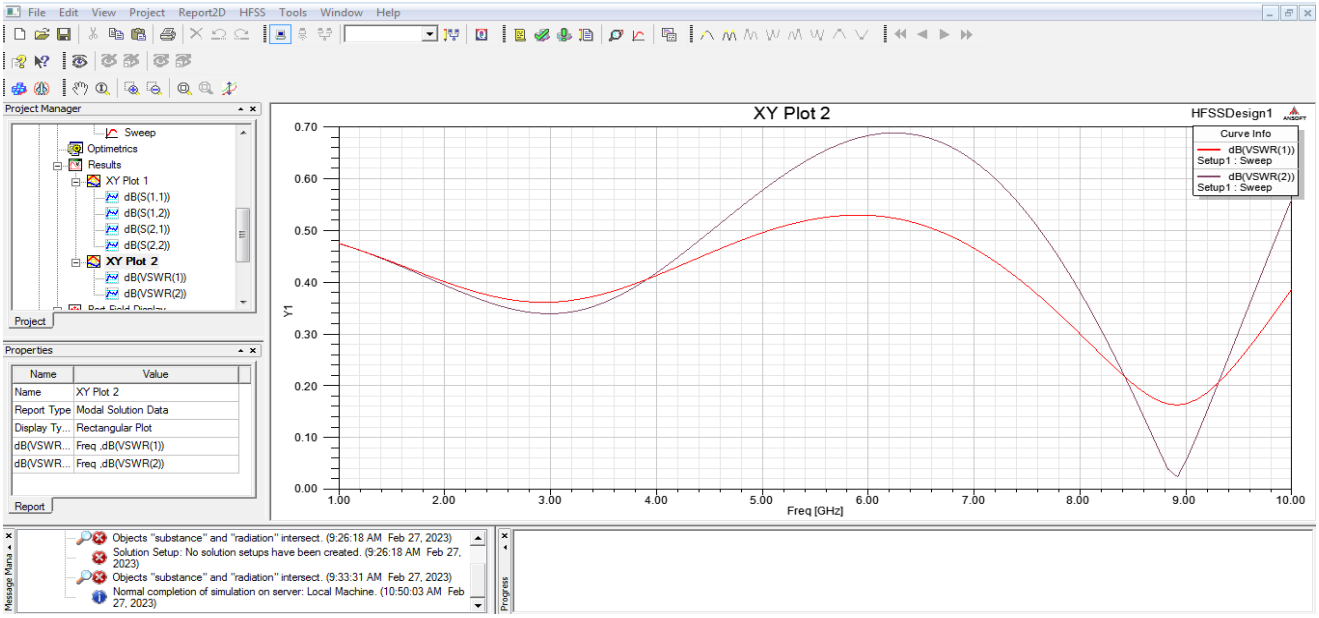
Next **analysis all**-any error rectify-ok



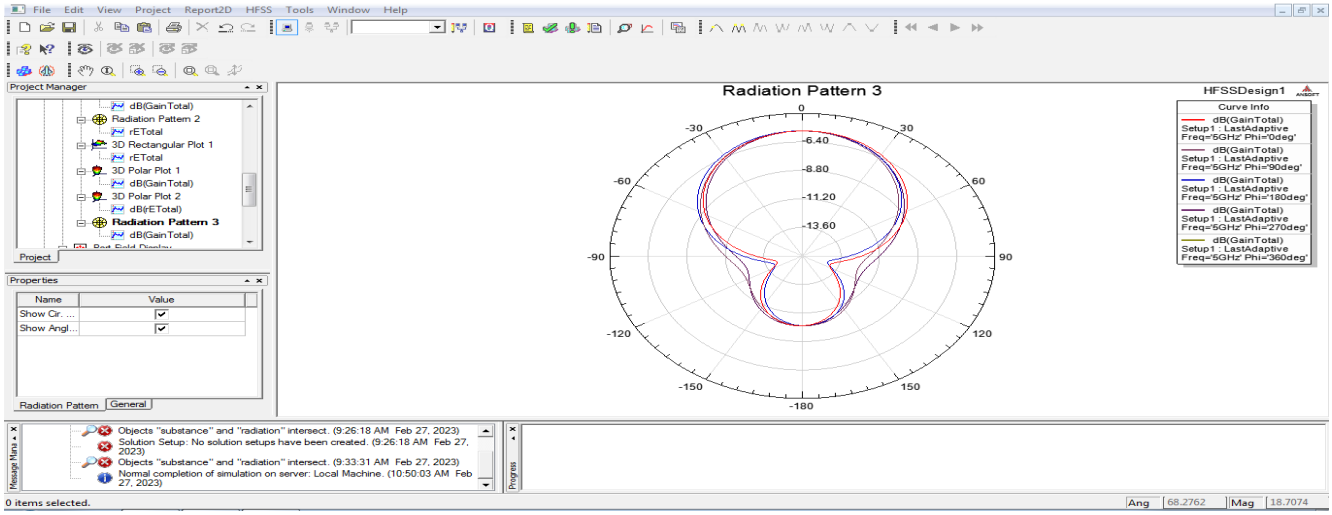
Model Waveforms: 5 GHZ (S Parameter)



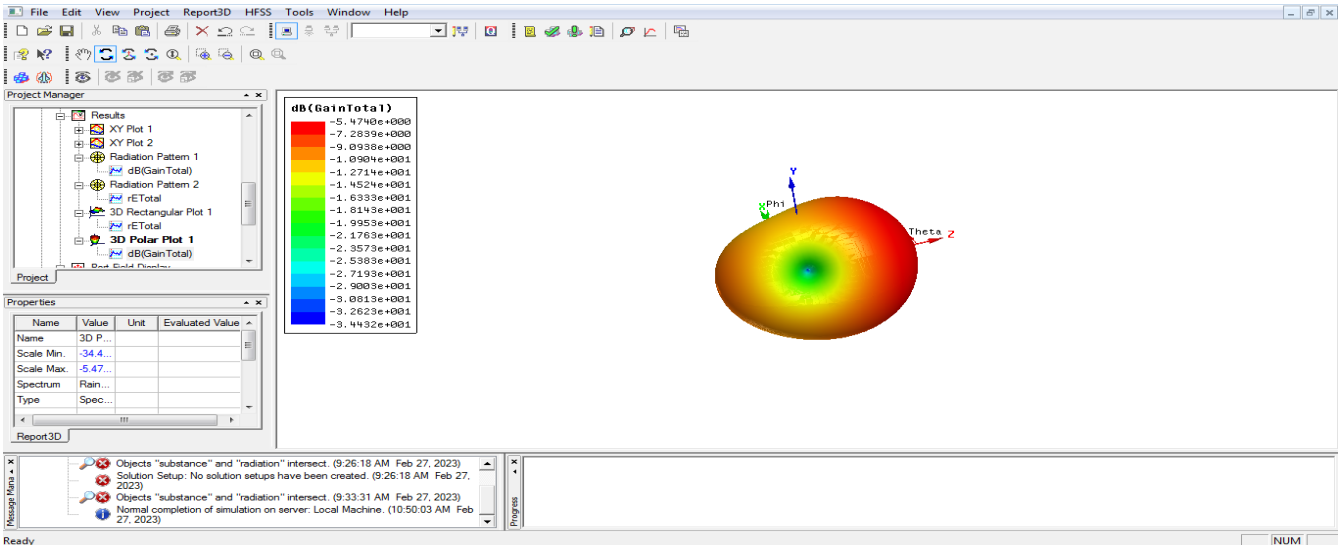
vswr graph



radiation pattern graphs



3d graphs



Tabular Column:

DESIGN CONSIDERATIONS Parameters	Width	Length	Height	Position
Ground plane				
substance				
TL				
Port1				
Port2				
radiation boundary				

Result:

Conclusion:

Viva questions:

1. Define microstrip antenna?

2. What are the types of microstrip antenna?

3. Give applications of microstrip antenna?

4. Write advantages of microstrip line?

5. Write the characteristics of microstrip transmission line?

Exp: 02 **MICRO STRIP PATCH ANTENNA** **Date:**

Aim: Design and characterization of Micro strip patch antenna.

Apparatus Required:

1. **Computer**
2. **Hfss software.**

Procedure:

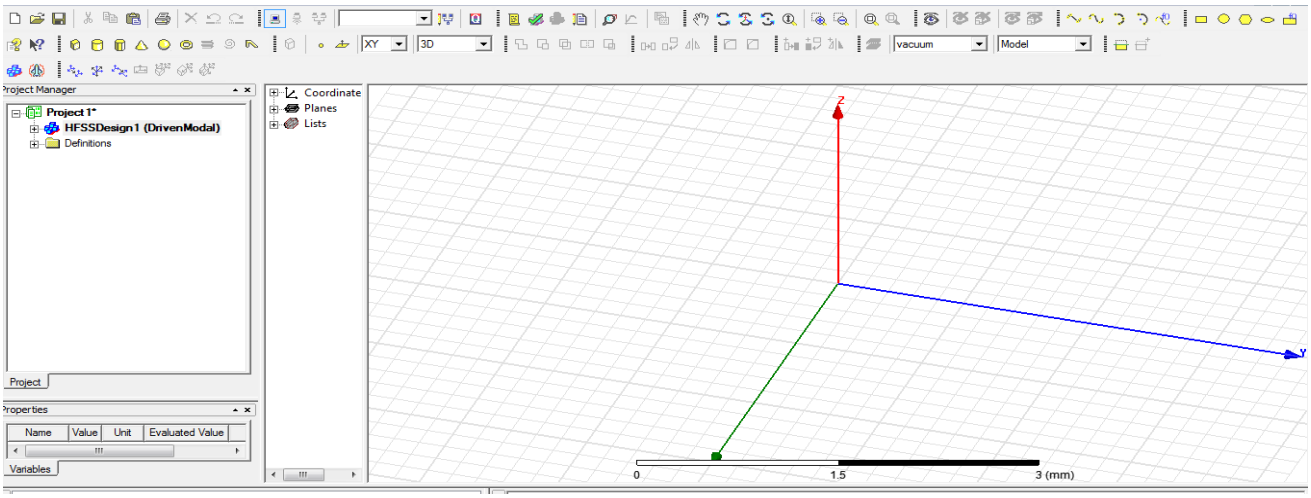
1. Open HFSS software and Insert new HFSS design.
2. Adjust the co-ordinates.
3. Create a ground plane(Rectangular 2D).
4. Create a dielectric substrate with FR4_Epoxy material with same size of ground plane with z-height 1.6mm.
5. Create patch antenna
6. Creating the feedline on the ground plane
7. Create two ports (port1).
8. Now give the perfect E to ground .
9. Create assign excitation-lampud port.
10. Then create radiation boundary on the designed ground and assign boundary . The radiation should be given to all the faces except at ground.
11. Assign frequency and no. of passes.
12. Now add freq sweep – fast – linear count.
13. Now check validation and analyze all.
14. Then go to results – Create model solution – rectangular plot – new report. Plot both
15. Then click HFSS – click radiation – click far field – infinite sphere and give values to phi and theta.
16. Then click on results and create far field.
17. Click plot of 3D – gain-dB – new report.

.

Tabular Column:

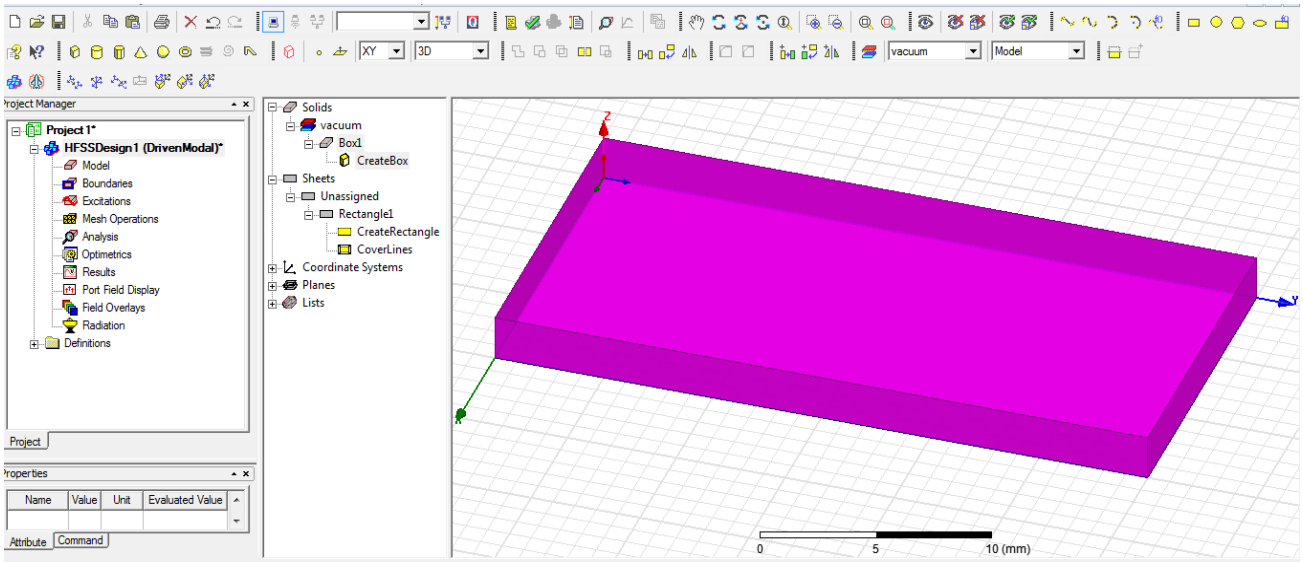
DESIGN CONSIDERATIONS Parameters	**Width**	**Length**	**Height**	**Position**
Ground plane				
Substance				
Patch				
feed				
Port1				
Radiation				

DESIGN: MICROSTRIP MICROSTRIP PATCH ANTENNA USING HFSS (2.4 GHZ)



Open **HFSS** project-click project , open - project insert **HFSS** design.

CREATE SUBSTANCE Select draw the **BOX** and design



Select fit all the contents in the view

Double click **box 1** (rename **substance**)

Material – edit fr4 - (4.4)

Select colour,

ok

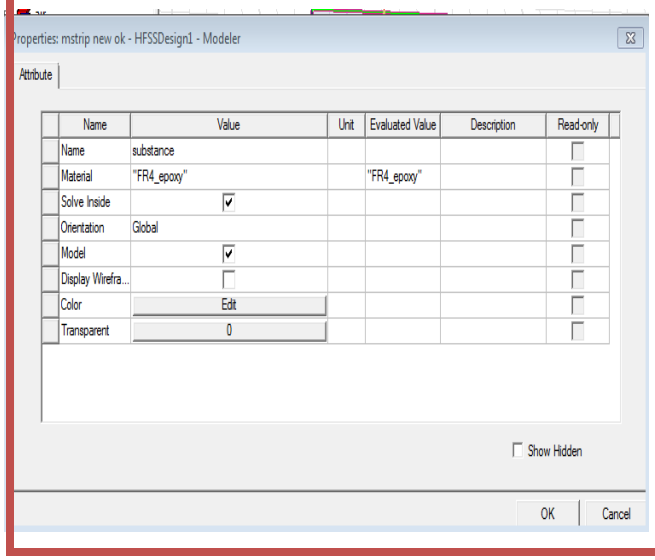
Double click **create box**

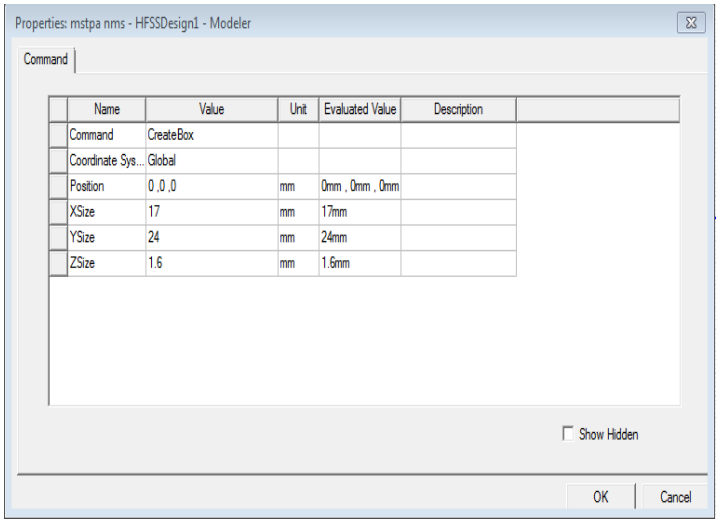
Position 0,0,0

X size-7.5mm

Y size-15mm

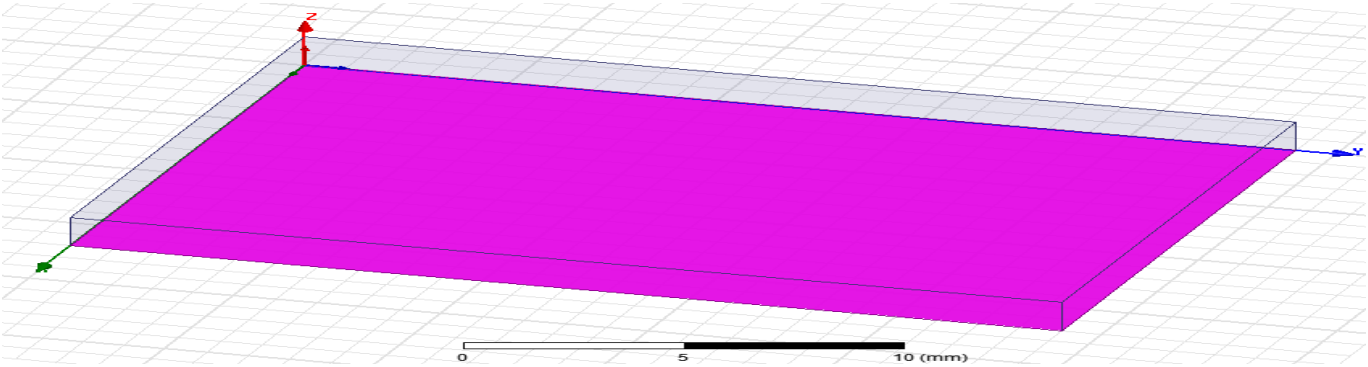
Z size-1.6mm Ok





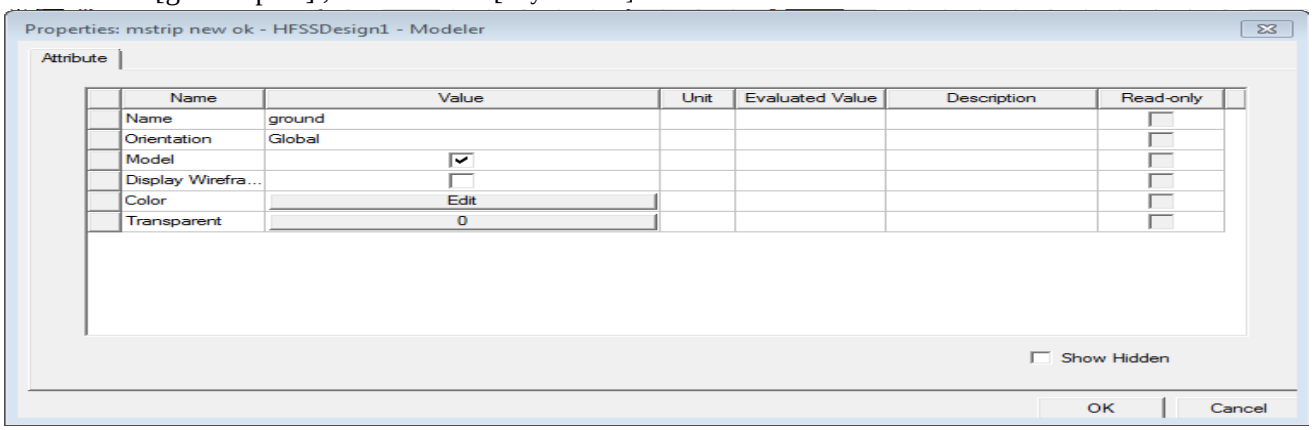
DESIGN GROUN PLAN:

Select click **rectangular draw**



Double click **rectangle1**

Name: [ground plan] , Colour : edit [as your wish]



Double click **create rectangle:**

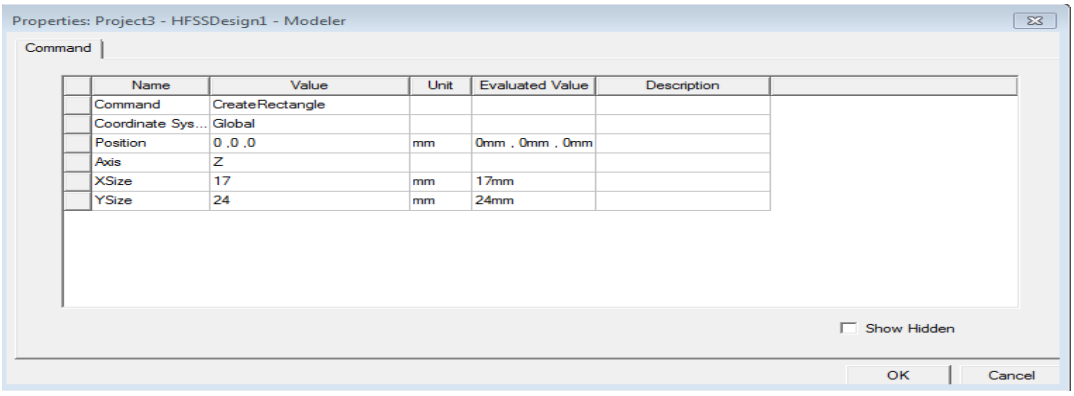
Position : 0,0,0

Axis : z

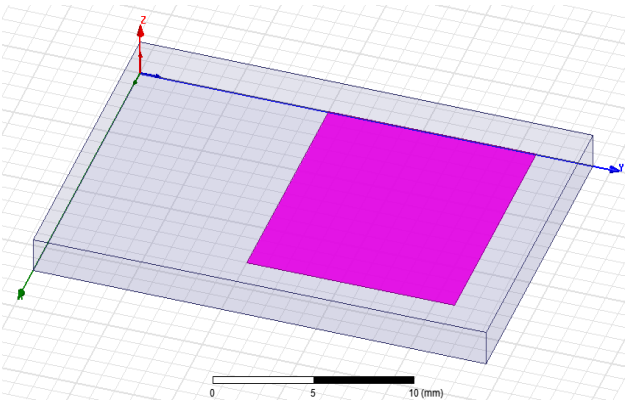
X size : 17mm

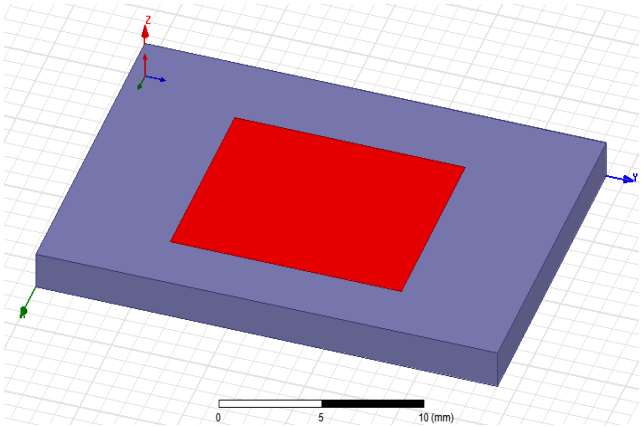
Y size : 24 mm

Ok



DESIGN PATCH ANTENNA





Select rectangle design center draw

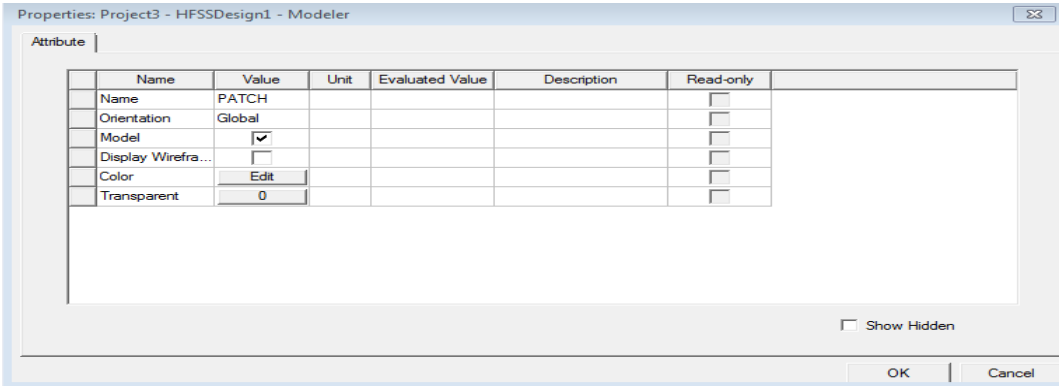
Double click **reactangle1,**

rename – **PATCH,**

Colour - edit

Ok

Double click



Create rectangle:

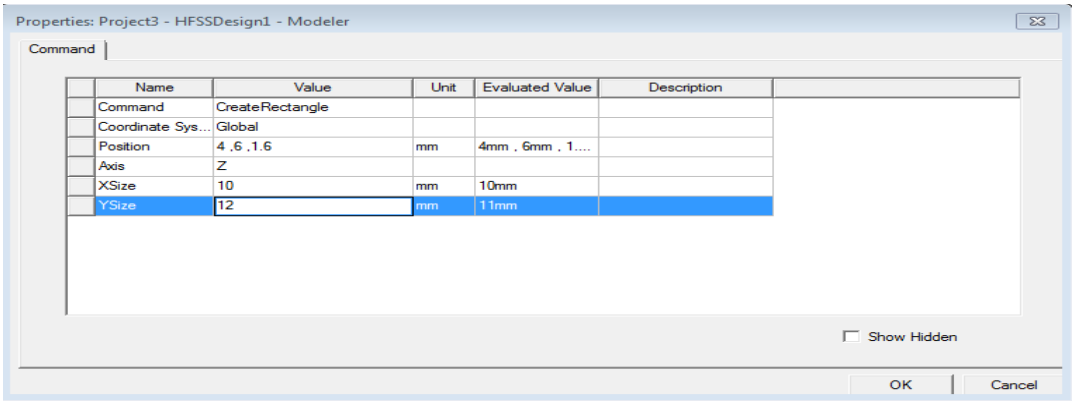
Position: 4,6, 1.6

Axis: z

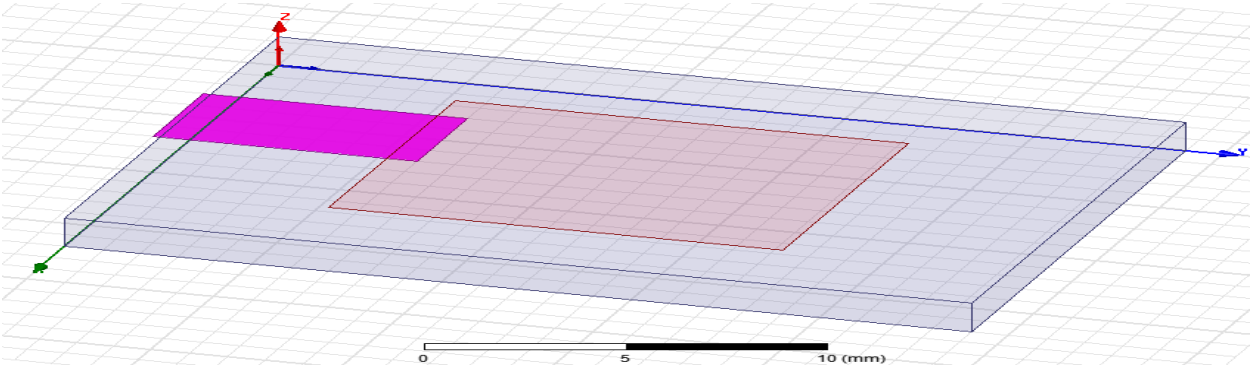
X axis: 10mm

Y size: 12mm

Ok

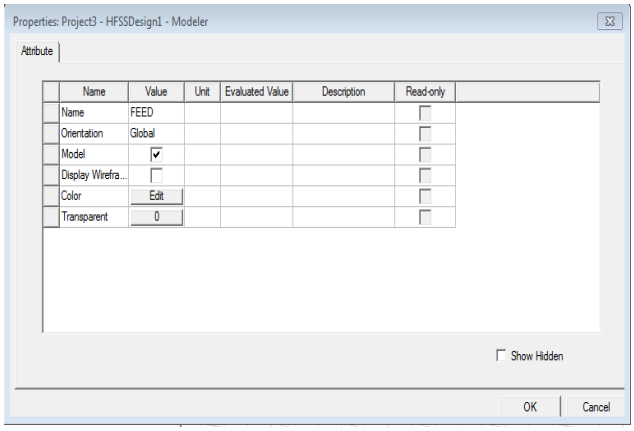


DESIGN FEED: Select click **rectangular draw**



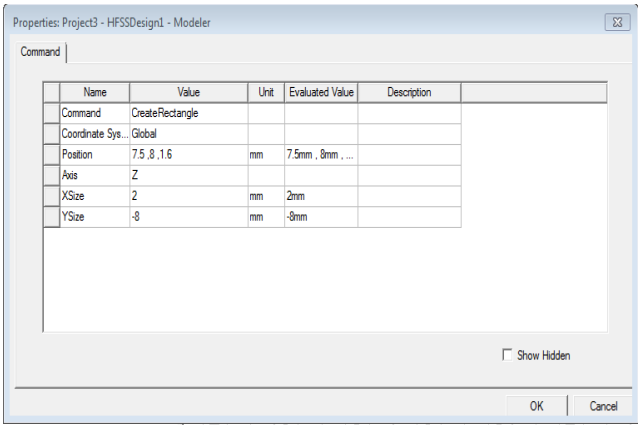
Double click **reactangle1,**

rename – **FEED,**
Colour - edit
Ok

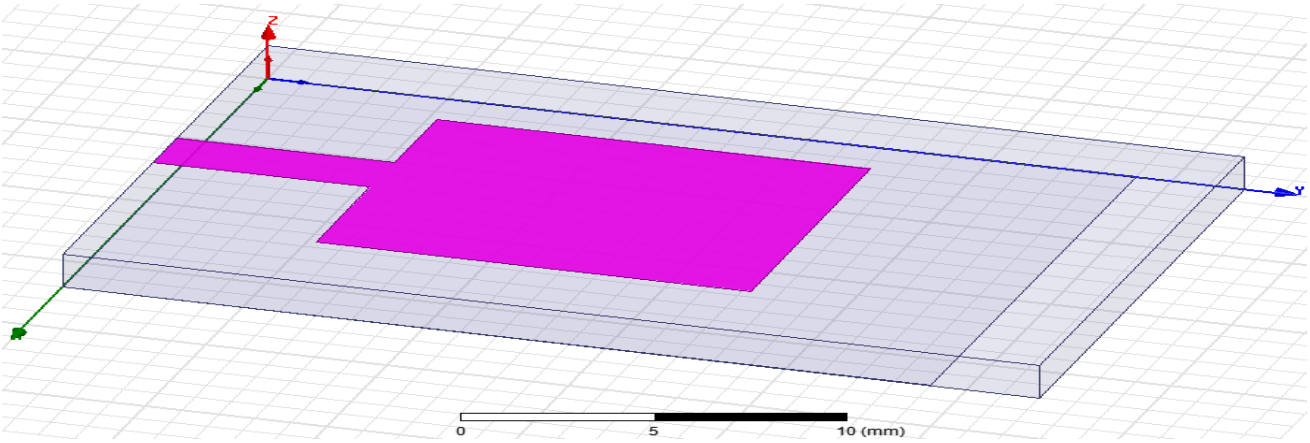


Double click **Create rectangle:**

Position: 7.5,8, 1.6
Axis: z
X axis: 2 mm
Y size: -8 mm

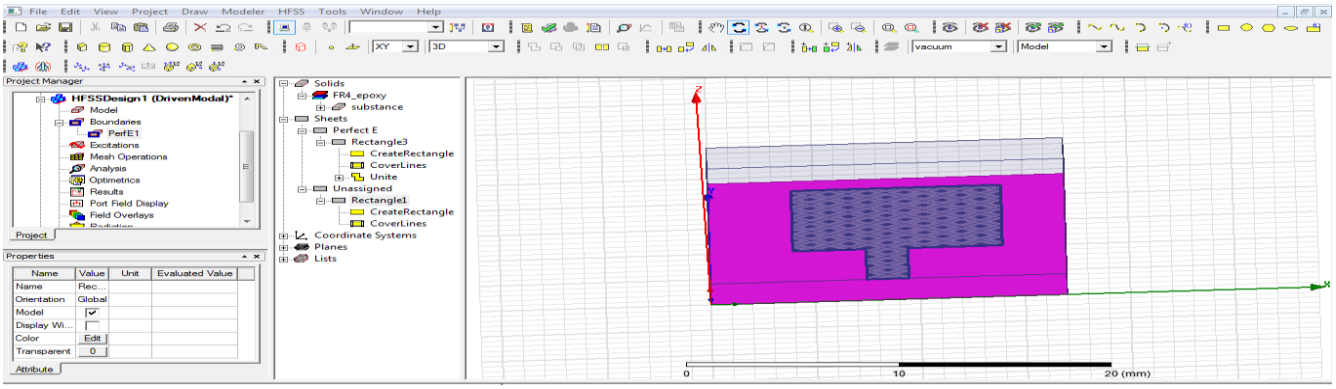


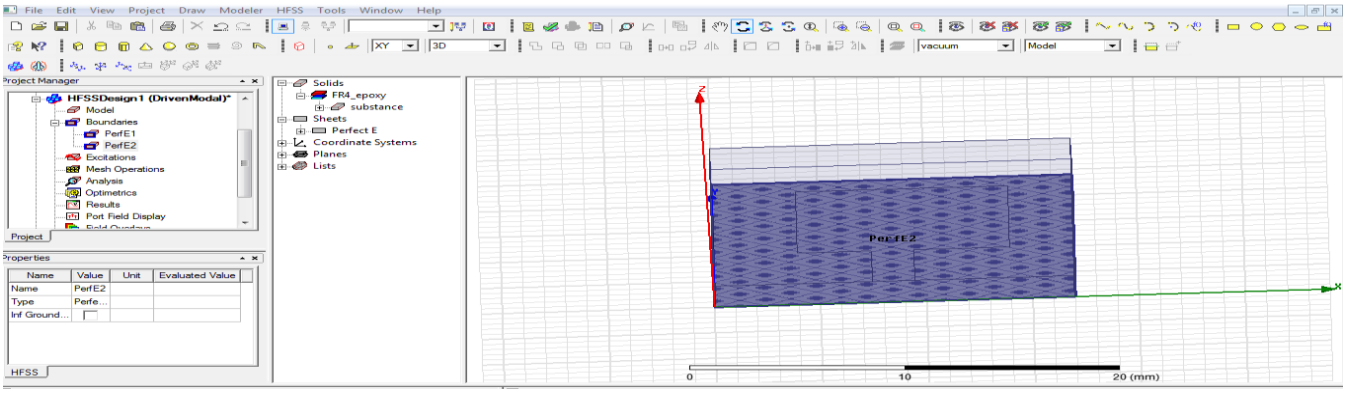
click **PATCH and FEED both unite**



DESIGN PERFECT ELECTRIC BOUNDARY - click **feed & patch**

Right click – assign boundary-perfect E-CLICK-name: **perfE1** - ok

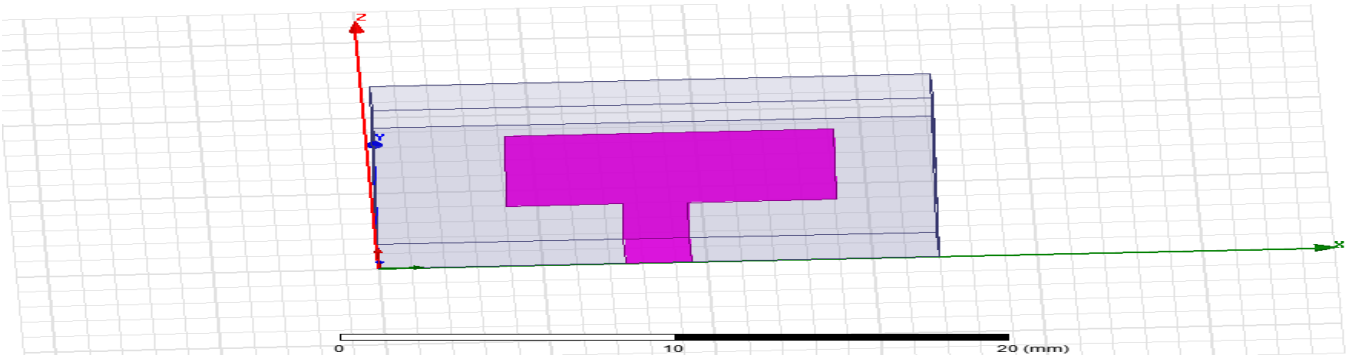




Click ground plan - right click-assign boundary-perfect E – click - name: **perfE2** ok

Design **ports** : (1) Change ZX

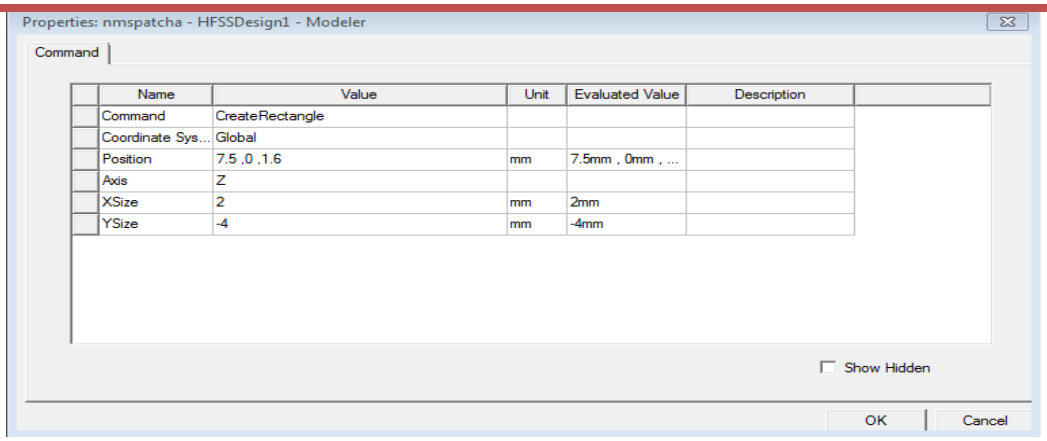
Select draw rectangle design **port1**



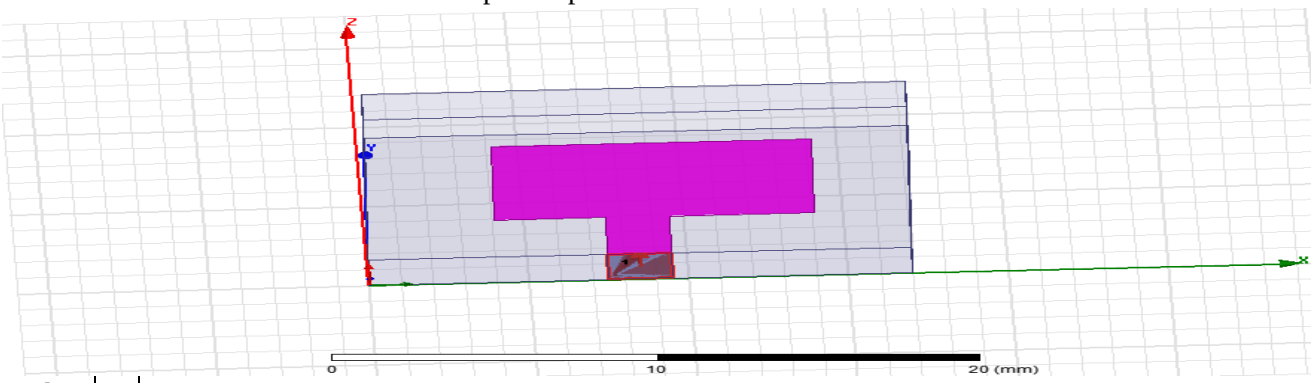
Double click

rectangle1

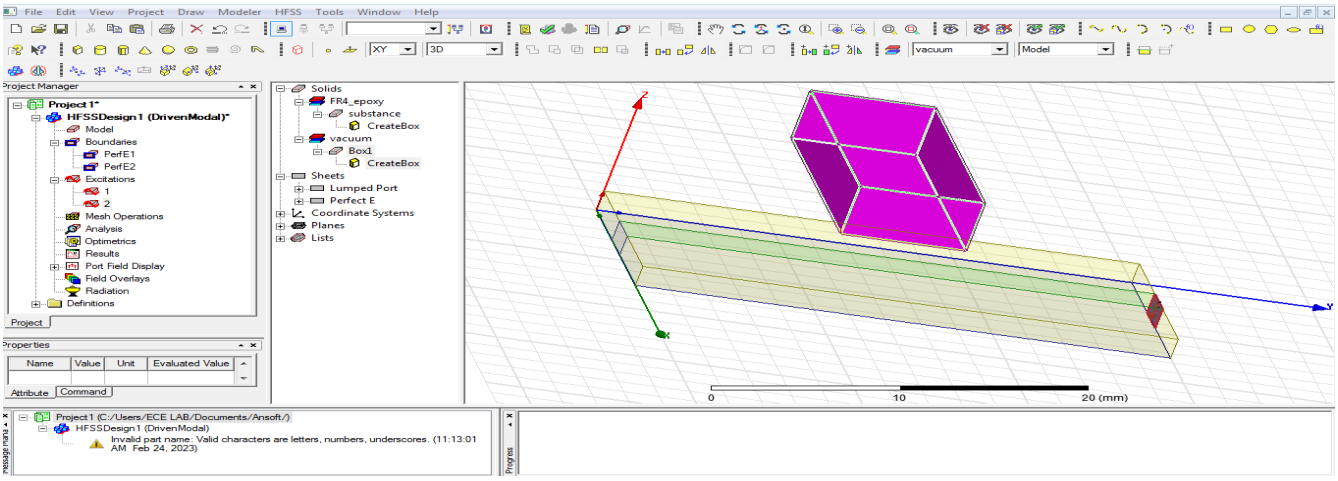
Rename **port1**
ok
Double click
create rectangle
Position-
(7.5, 0, 1.6)
Axis-Y
X size- 2mm
Z size-(-4mm)



Select port1-right click assign excitation-lumped port-click-name1-resistance 50 ohms-next, select none new line –draw a line-defined-next-full port impedance 50 ohms finish



Select XY
Design **radiation boundary**



Draw **box**
double click

box1

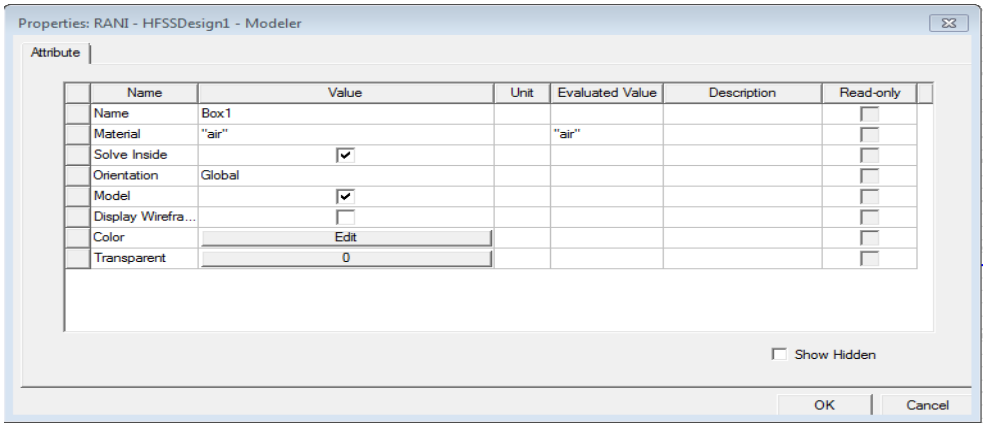
rename

radiation boundary

Material –

edit **air**

Colour



edit

ok

Double click

create box

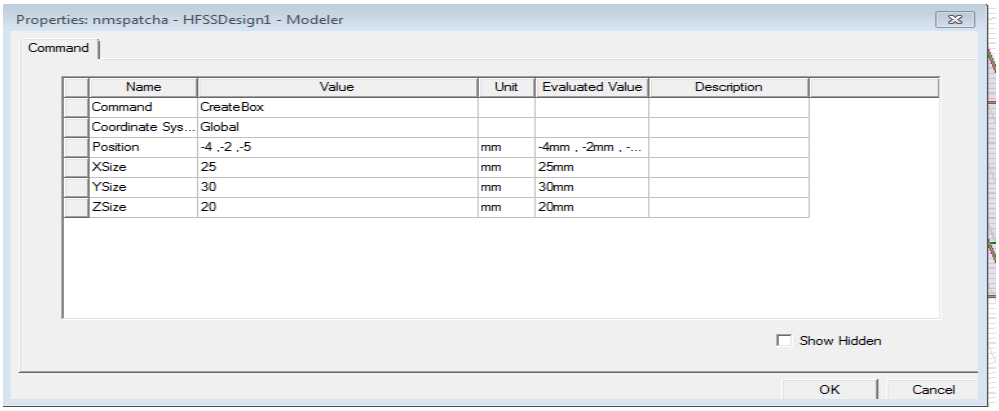
Position- (-4,-2,-5)

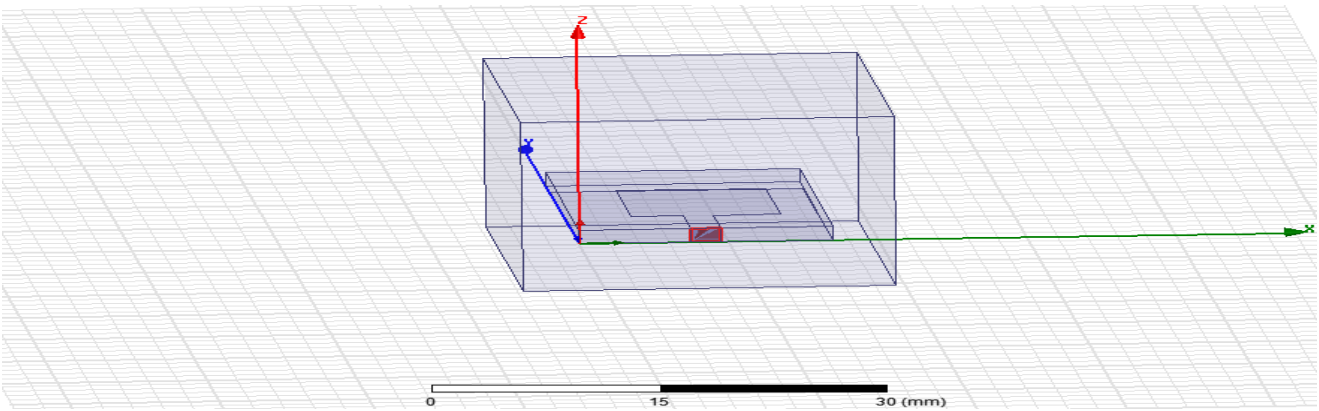
X size- 25mm

Y size- 30mm

Z size- 25mm

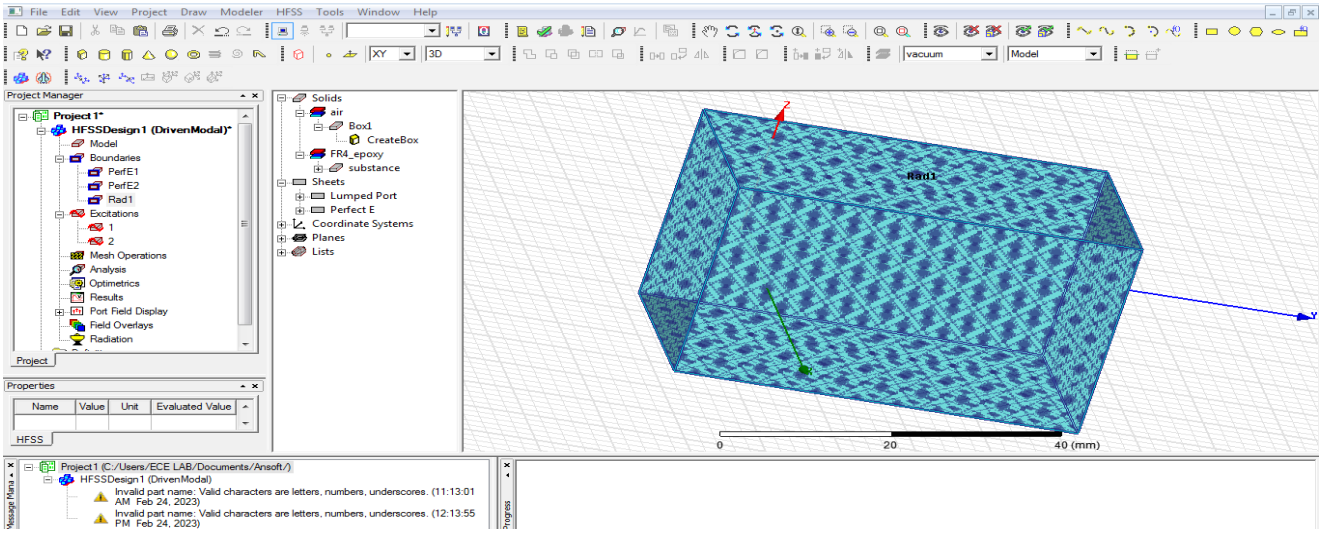
ok





Fit all

Click radiation boundary-right click-assign boundary –radiation click name **rad1** ok



Analysis –right click add solution setup click name setup1 solution frequency-2.4GHZ

Maximum number of passes -12

Maximum Delta S - 0.02 ok

Analysis right click- **setup1** right click - add frequency sweep

Sweep type : fast

Type : linear count

Start freq:1GHZ

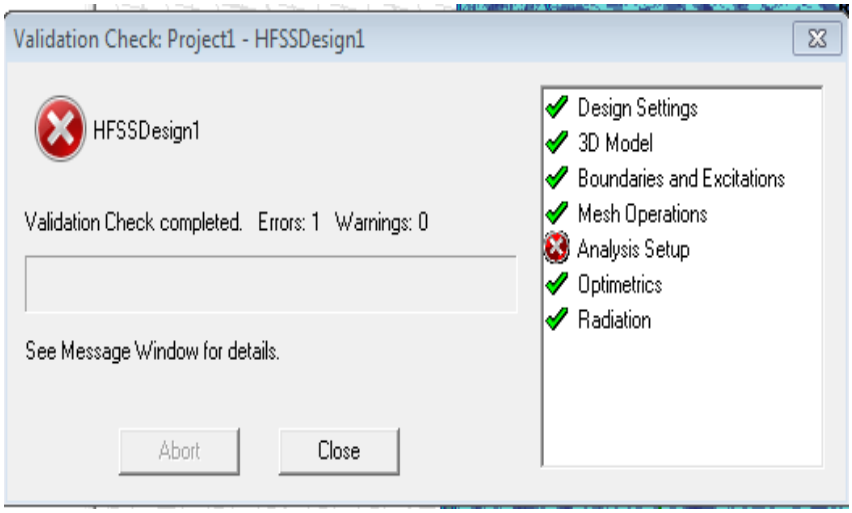
Stop freq:10GHZ

Count:101GHZ

Click display-see all frequencies ok

Result analysis(error checking)

Click double click **validity**



HFSS Design- design setting

3D model

Boundaries and excitations

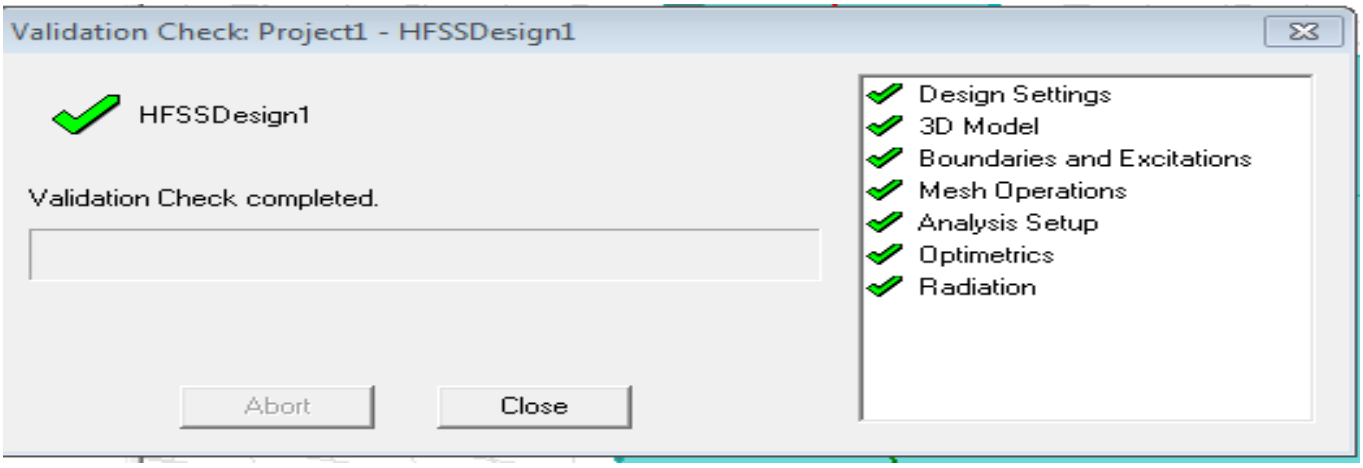
Mesh operation

Analysis setup

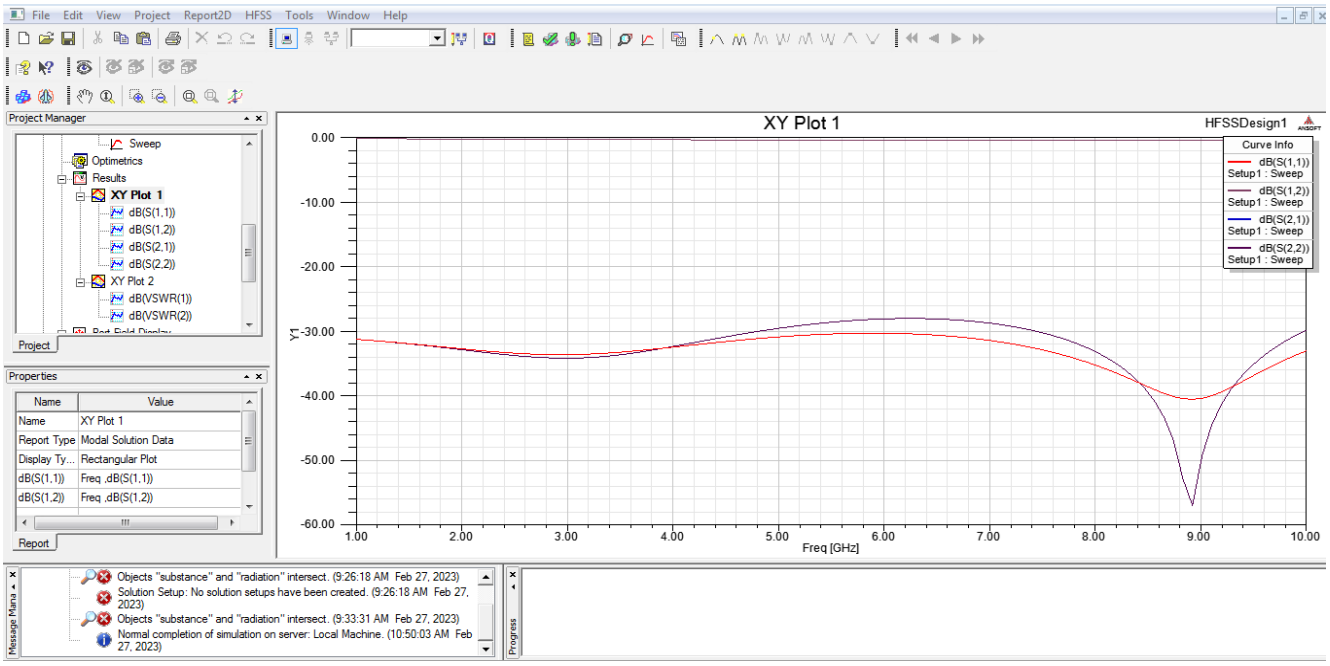
Optimetrics

Radiation

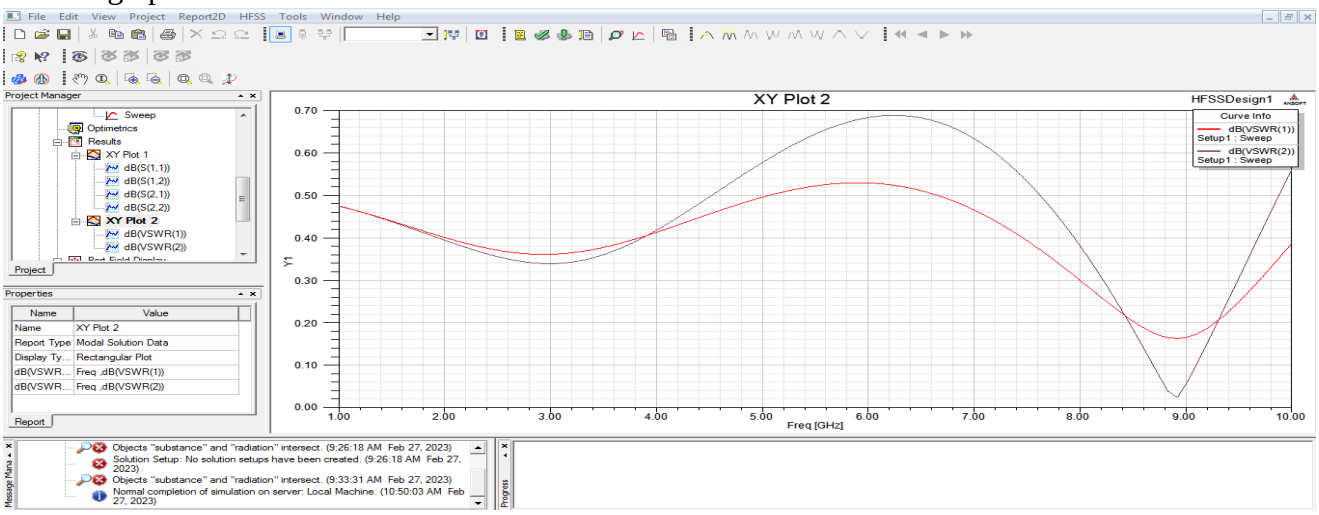
Next **analysis all**-any error rectify-ok



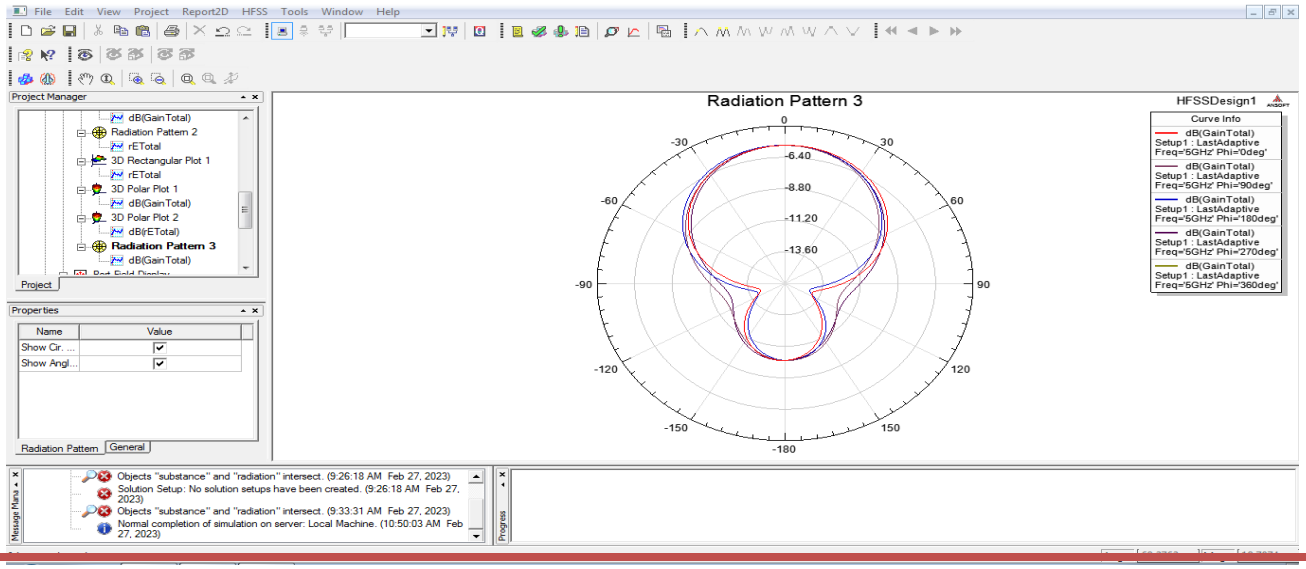
Model Waveforms: 2.4 GHZ (S Parameter)



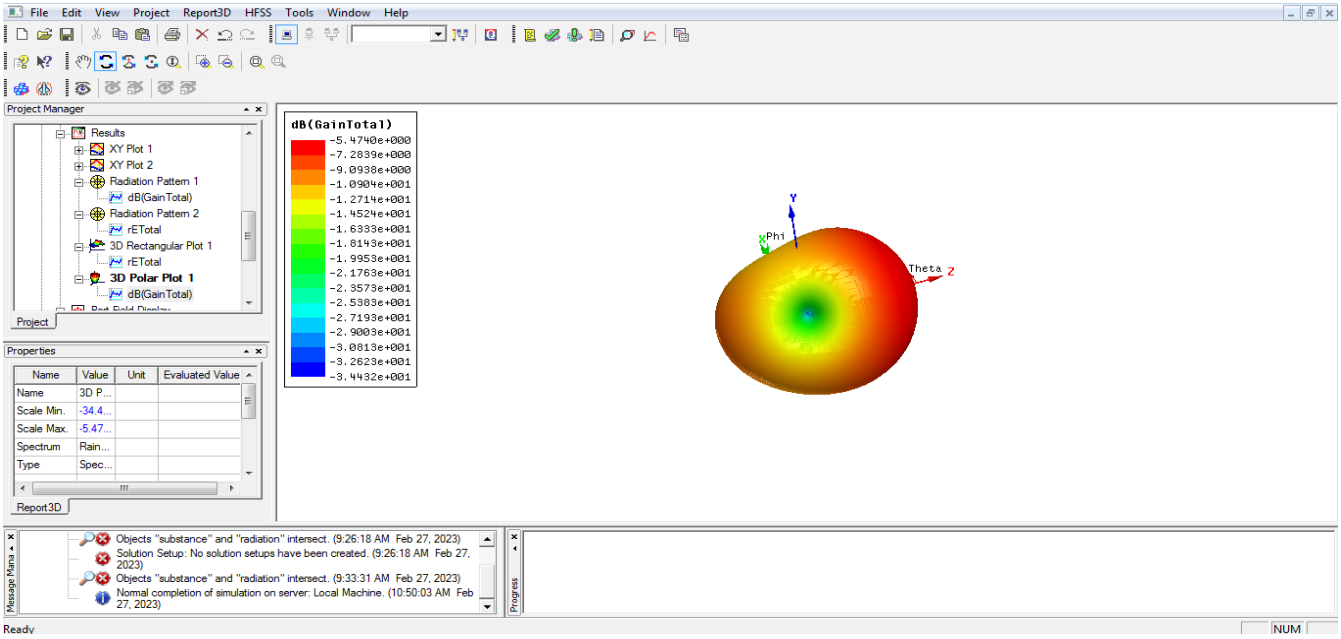
vswr graph



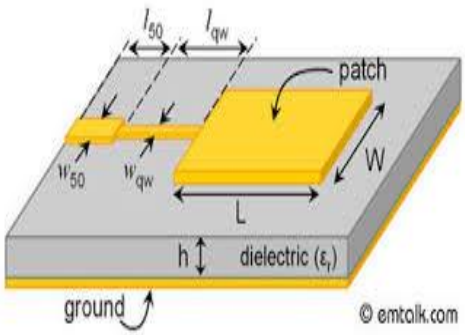
radiation pattern graphs

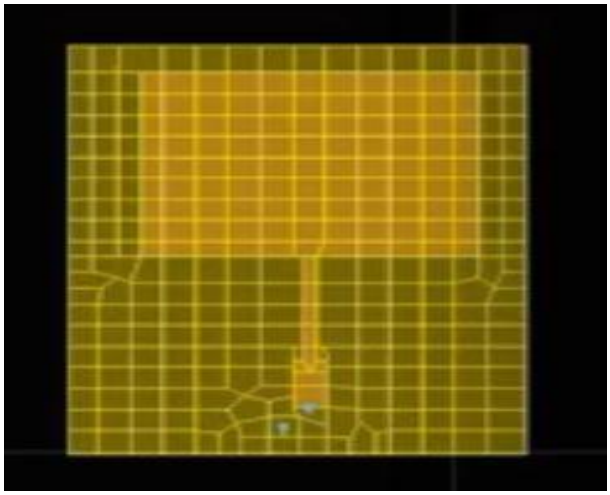


3d graphs



Model wave





Formulas:

Result:

Conclusion:

Viva questions:

1.What is micro strip patch antenna?

2.What is micro strip antenna used for?

3.What are the 3 types of micro strip antenna?

4.What is the difference between micro strip antenna and patch antenna?

5.What is micro strip antenna and its characteristics?

Exp.No: 03 **Date:**

MICROSTRIP TRANSMISSION LINE AND STANDING WAVE PATTERN

Aim: Analyse of a Microstrip Transmission Line and standing wave pattern at various frequencies

Apparatus Required:

1. **Computer**
2. **Hfss software.**

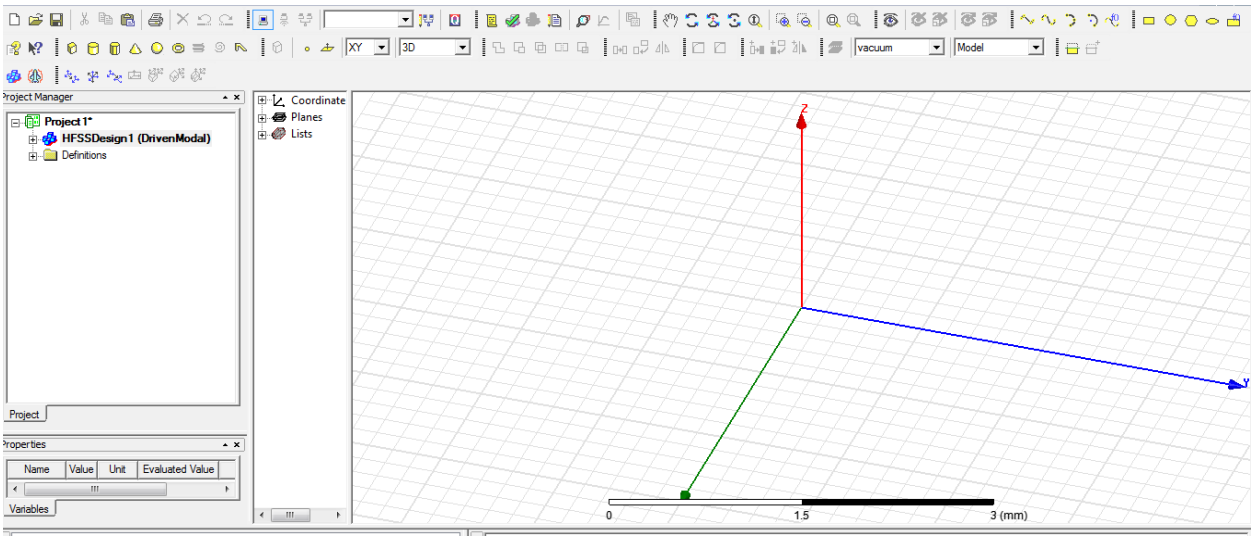
Procedure:

1. Open HFSS software and Insert new HFSS design.
2. Adjust the co-ordinates.
3. Create a ground plane(Rectangular).
4. Create a dielectric substrate with FR4_Epoxy material with same size of ground plane with z-height 1.6mm.
5. Creating the TL the ground plane & substance.
6. Create two ports (port1&port2).
7. Now give the perfect E to ground .
8. Create assign excitation-lampudport.
9. Then create radiation boundary on the designed ground and assign boundary **b**. The radiation should be given to all the faces except at ground.
10. Assign frequency and no. of passes.
11. Now add freq sweep – fast – linear count.
12. Now check validation and analyze all.
13. Then go to results – Create model solution – rectangular plot – new report. Plot both
14. Then click HFSS – click radiation – click far field – infinite sphere and give values to phi and theta.
15. Then click on results and create far field.
16. Click plot of 3D – gain-dB – new report.

Tabular Column:

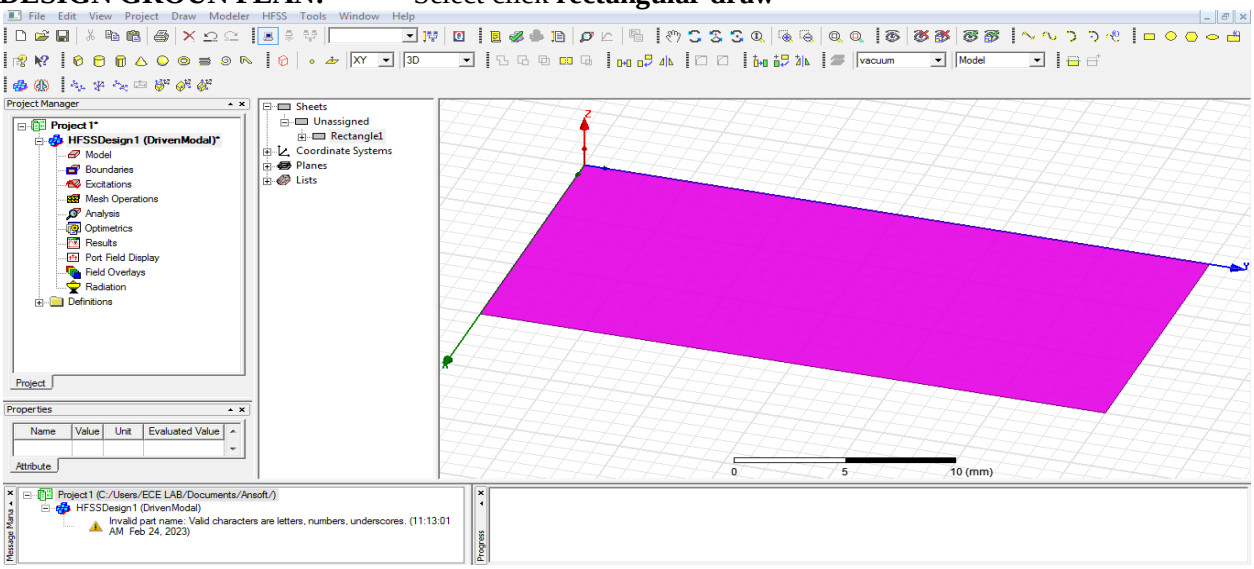
DESIGN CONSIDERATIONS Parameters	Width	Length	Height	Position
Ground plane				
substance				
TL				
Port1				
Port2				
radiation boundary				

Design: **MICROSTRIP TRANSMISSION LINE USING HFSS (2.4GHZ &3.8GHz)**



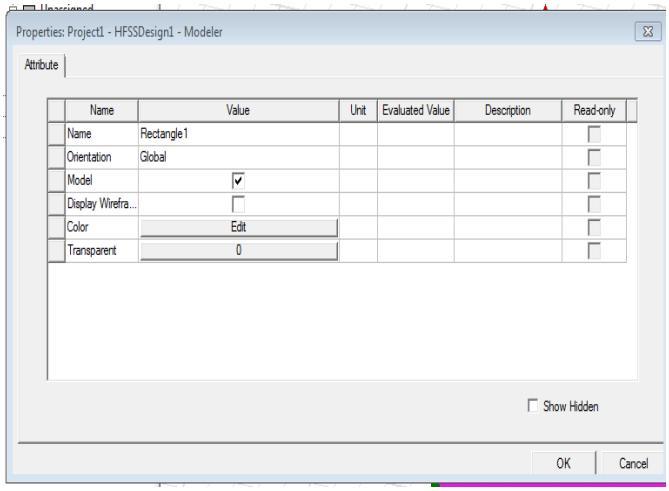
Open **HFSS** project-click project , open - project insert **HFSS** design.

DESIGN GROUN PLAN: Select click **rectangular draw**

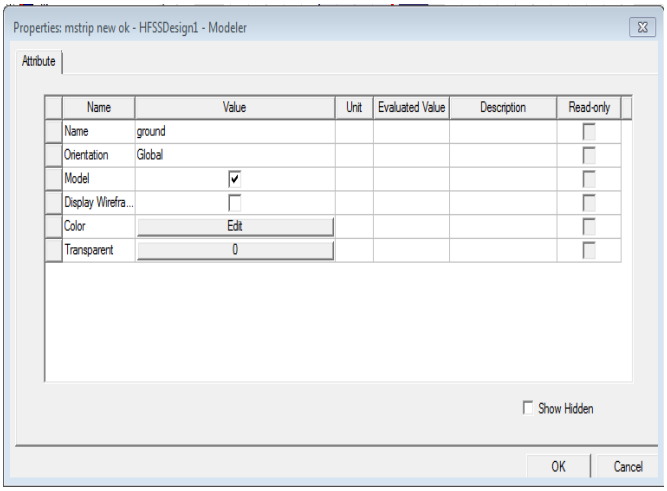


Double click **rectangle1**

Name : [ground plan] , Colour : edit [as your wish]



before name change



after name change

Double click

create rectangle:

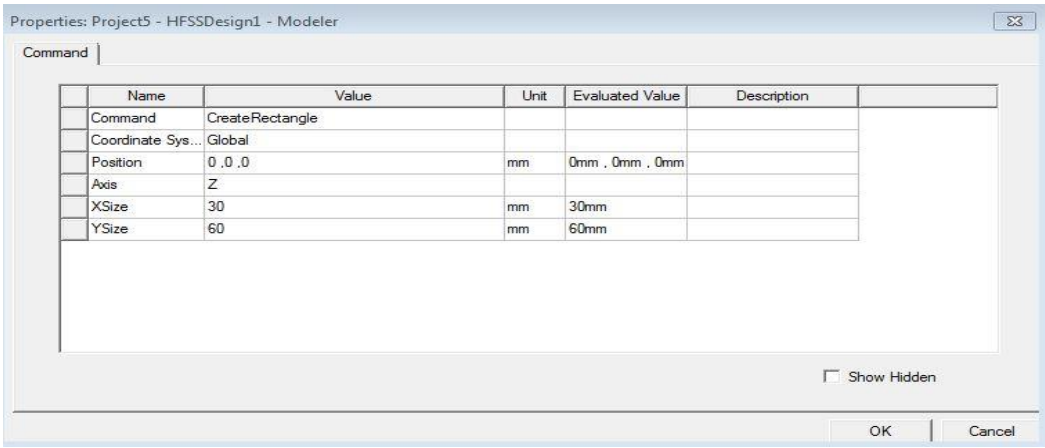
Position : 0,0,0

Axis : z

X size : 30 mm

Y size : 60 mm

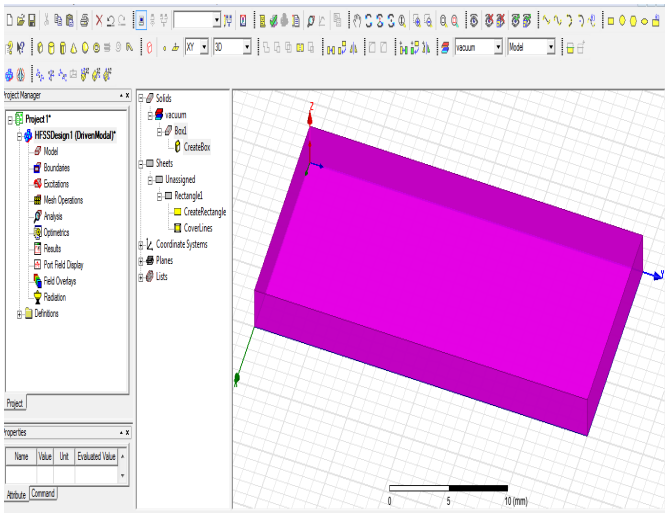
Ok

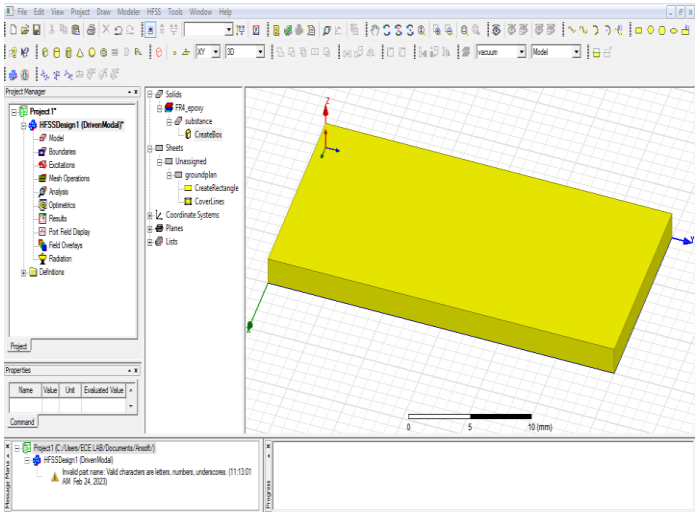


Select fit all the contents in the view

CREATE SUBSTANCE

Select draw the **BOX** and design



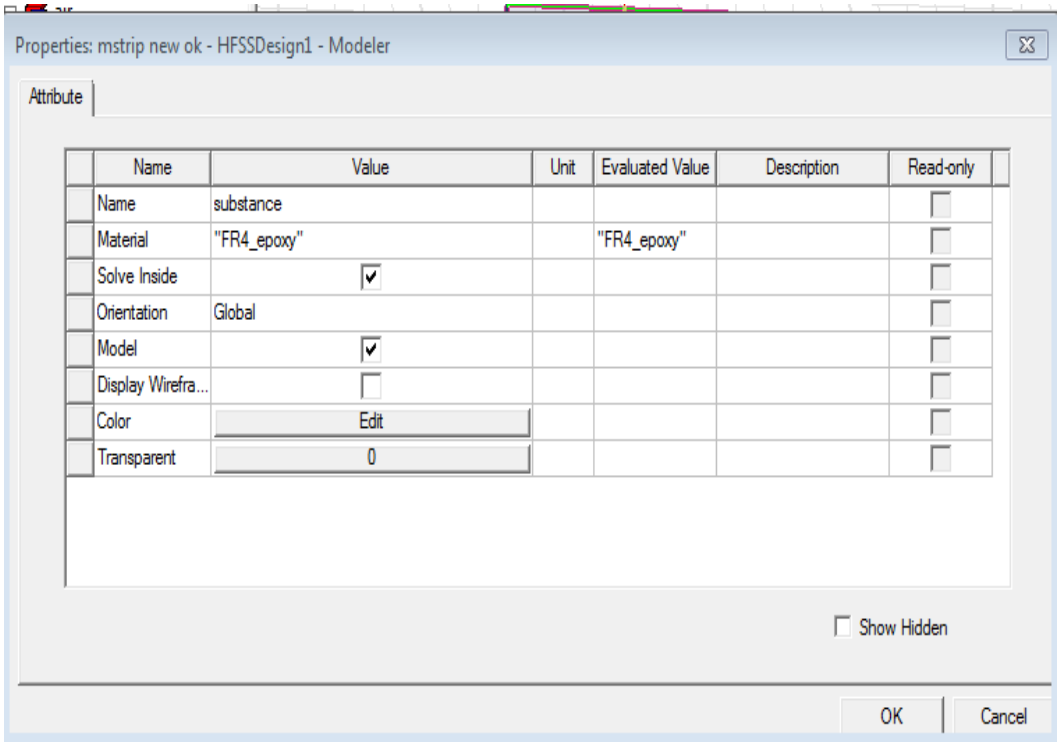


Double click **box 1** (rename **substance**)

Material – edit fr4 - (4.4)

Select colour

Ok



Double click **create box**

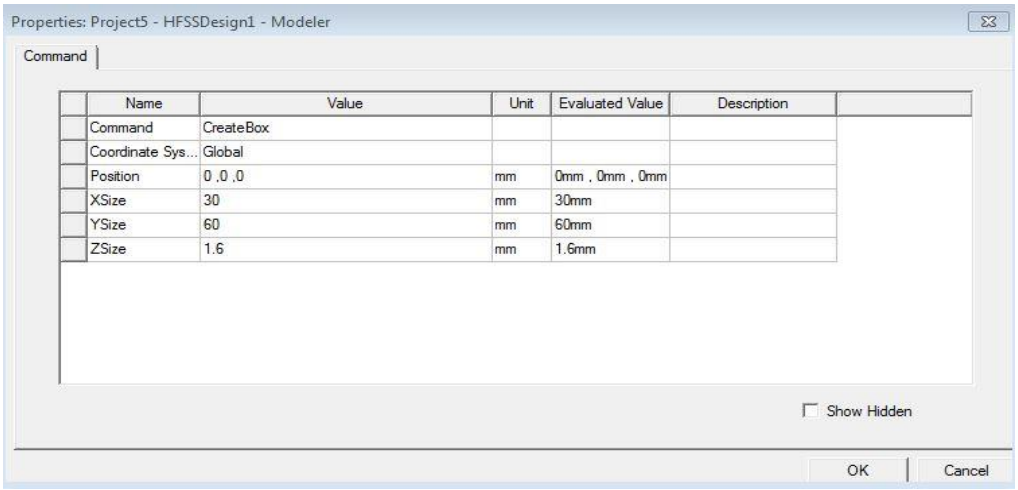
Position 0,0,0

X size-30mm

Y size-60mm

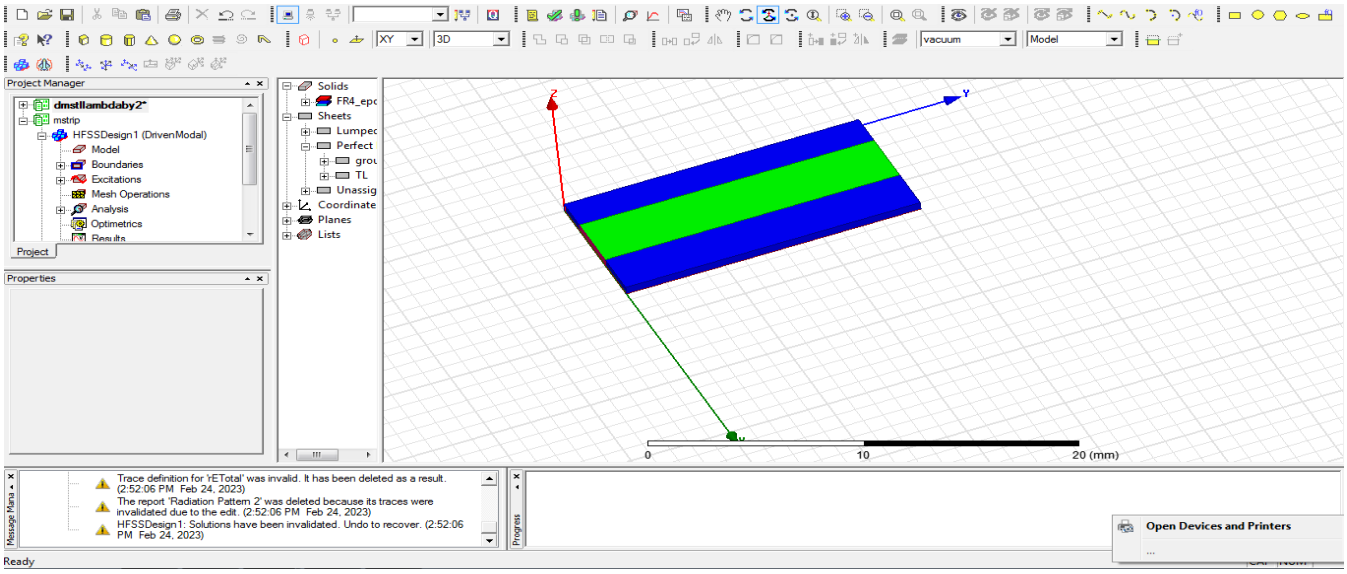
Z size-1.6mm

Ok



DESIGN TRANSMISSION LINE

Select rectangle design center draw

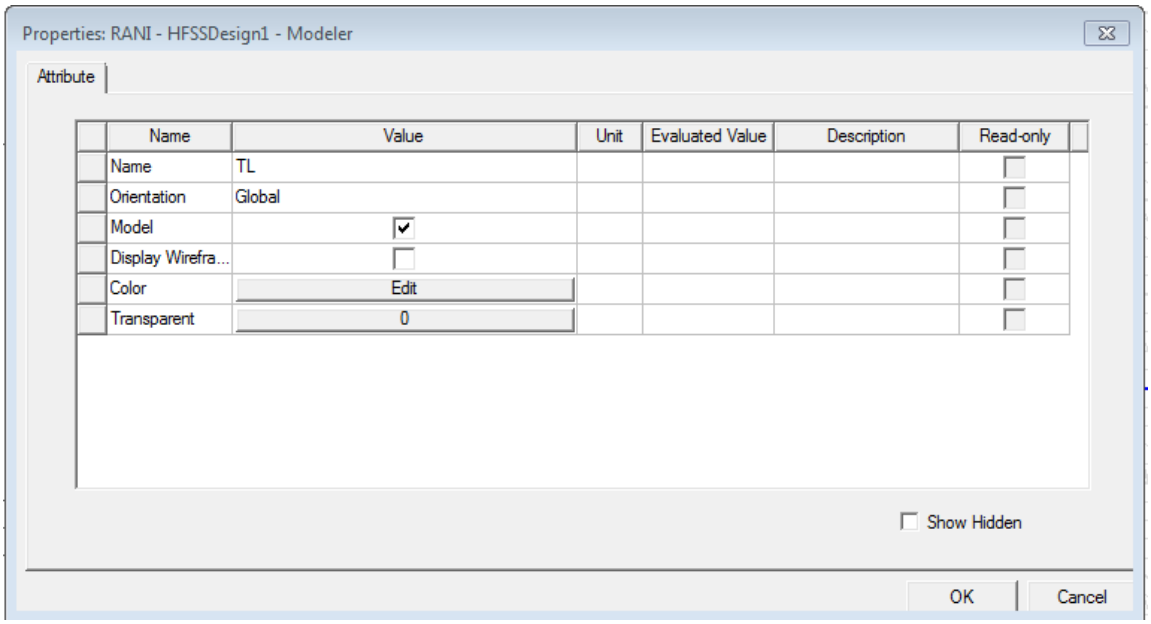


Double click **reactangle1**

rename - **TL**

Colour edit

Ok



Double click **create rectangle:**

Position –

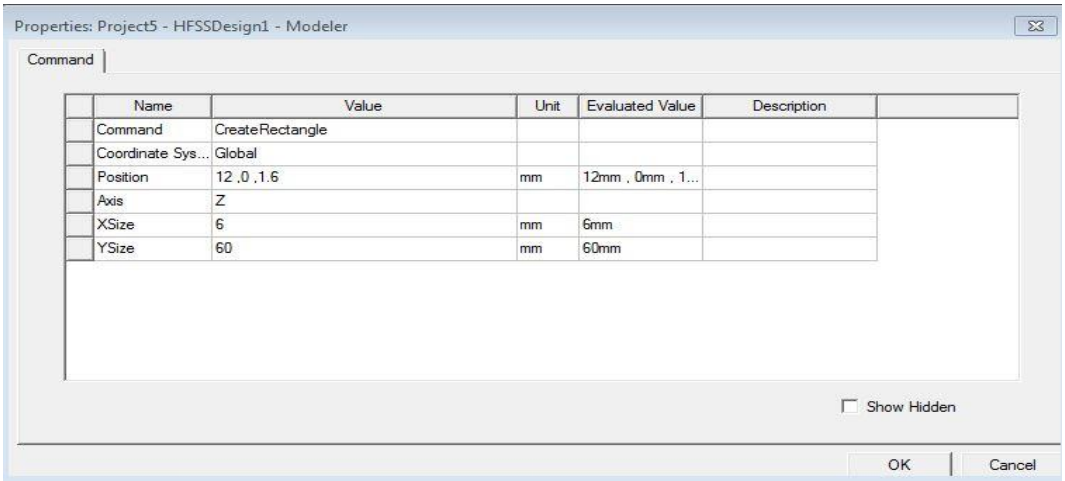
12, 0, 1.6

Axis – z

X axis - 6mm

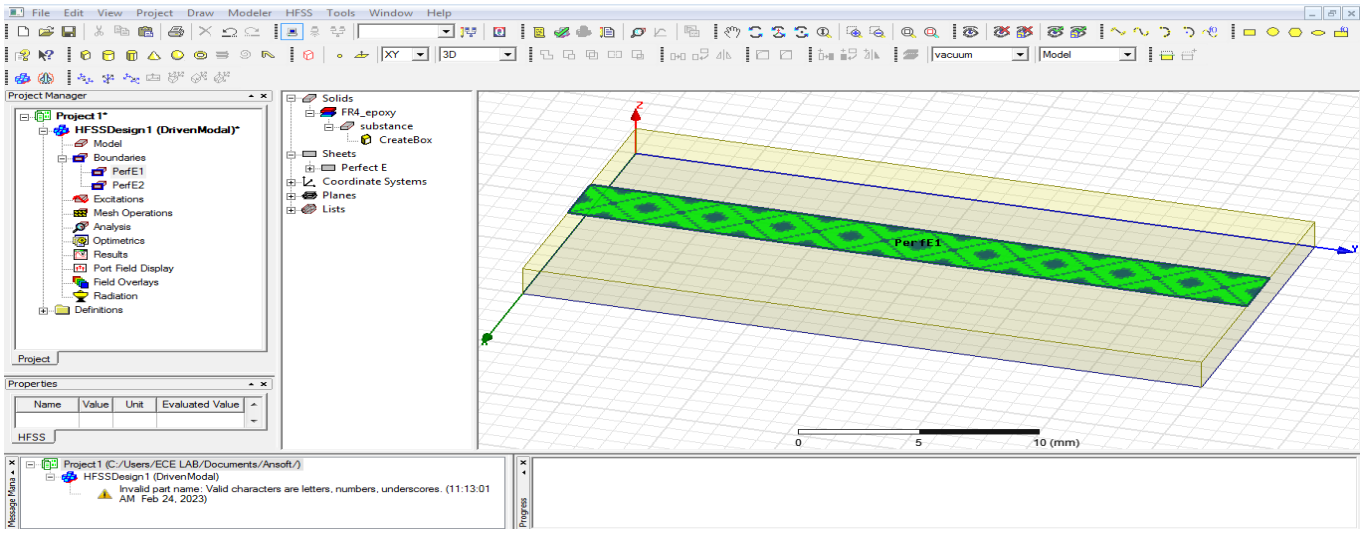
Y size - 60mm

Ok

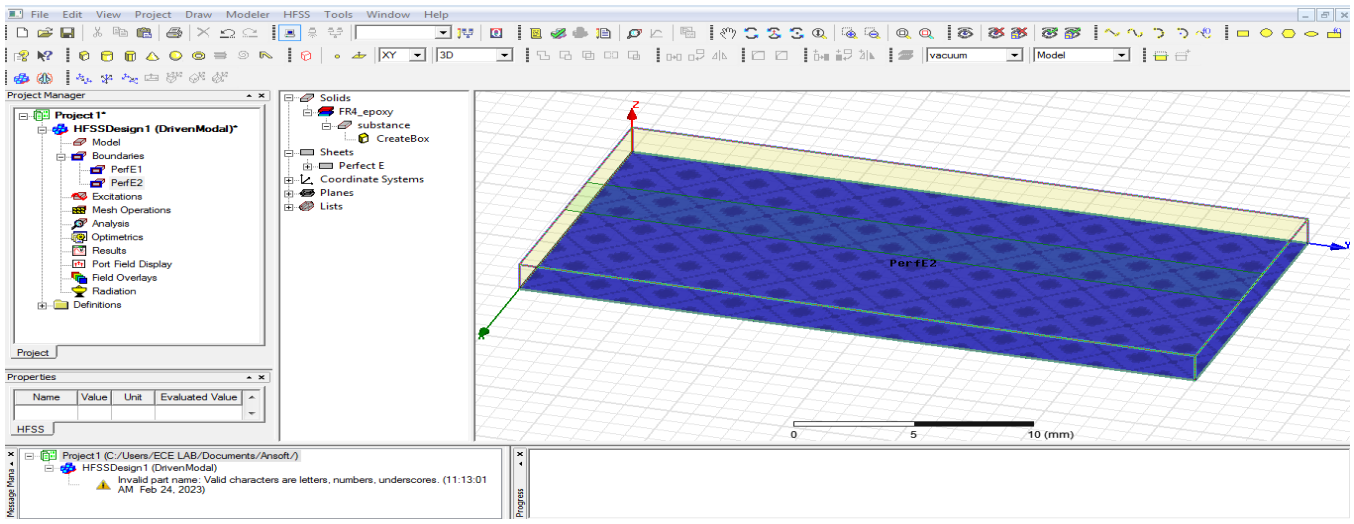


DESIGN PERFECT ELECTRIC BOUNDARY - click **TL**

Right click – assign boundary-perfect E-CLICK-name: **perfE1** - ok



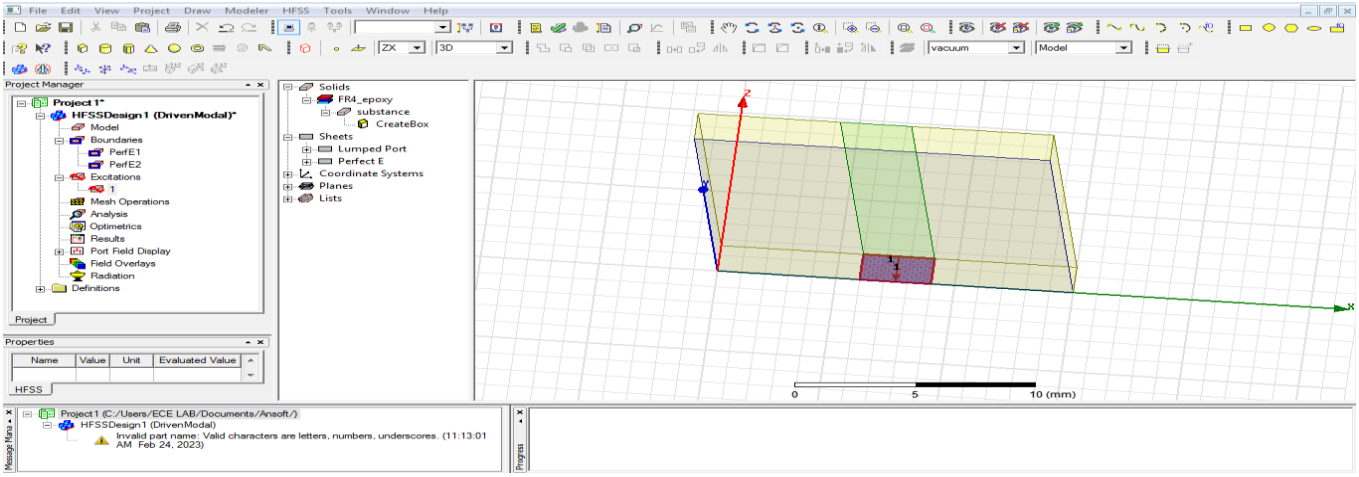
Click ground plan - right click-assign boundary-perfect E – click - name: **perfE2** ok



Design **ports** : (1&2)

Change ZX

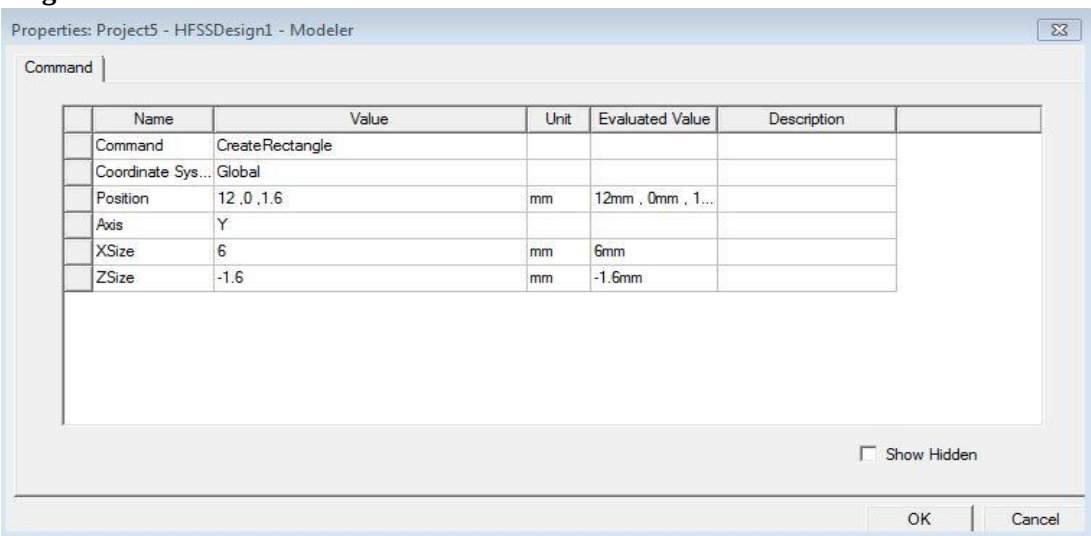
Select draw rectangle design transmission line **port1**



Double click **rectangle1**

Rename **port1** ok

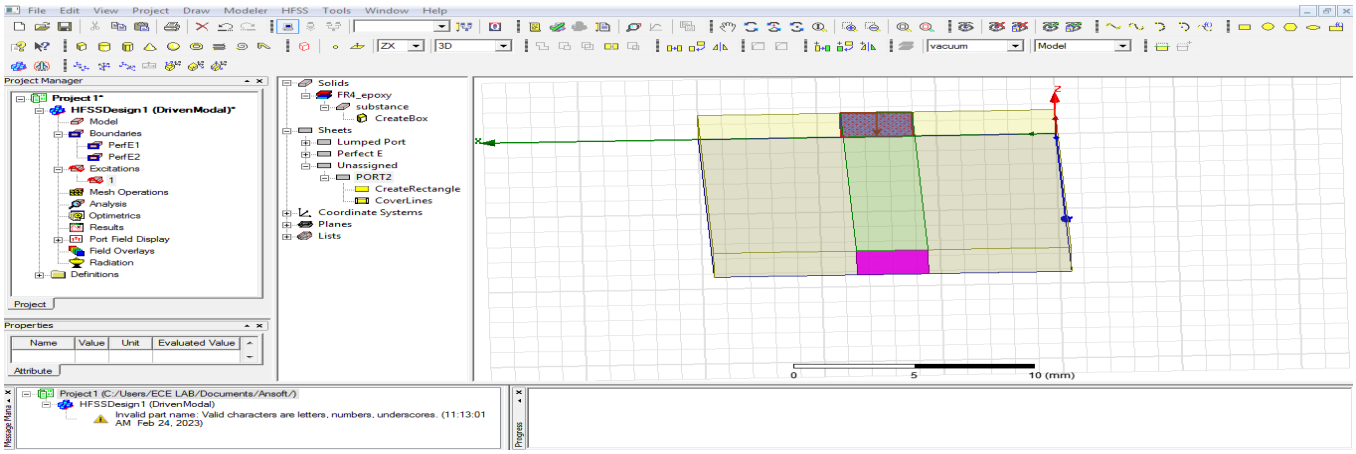
Double click **create rectangle**
Position-(12,0,1.6)
Axis-Y
X size-6mm
Z size-(-1.6mm)
ok



Select port1-right click assign excitation-lumped port-click-name1-resistance 50 ohms-next select none new line –draw a line-defined-next-full port impedance 50 ohms finish

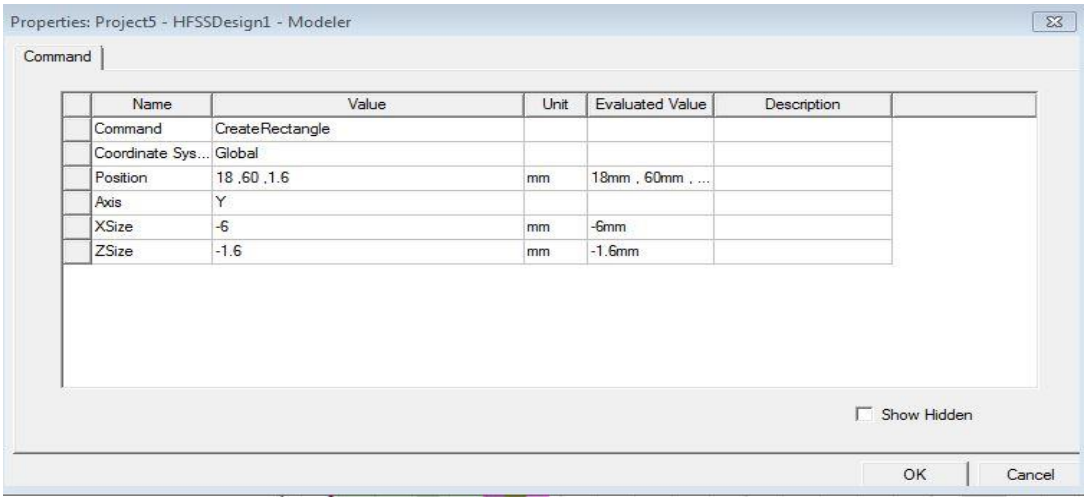
Design port2

Take rectangle - Select draw rectangle design transmission line **port2**

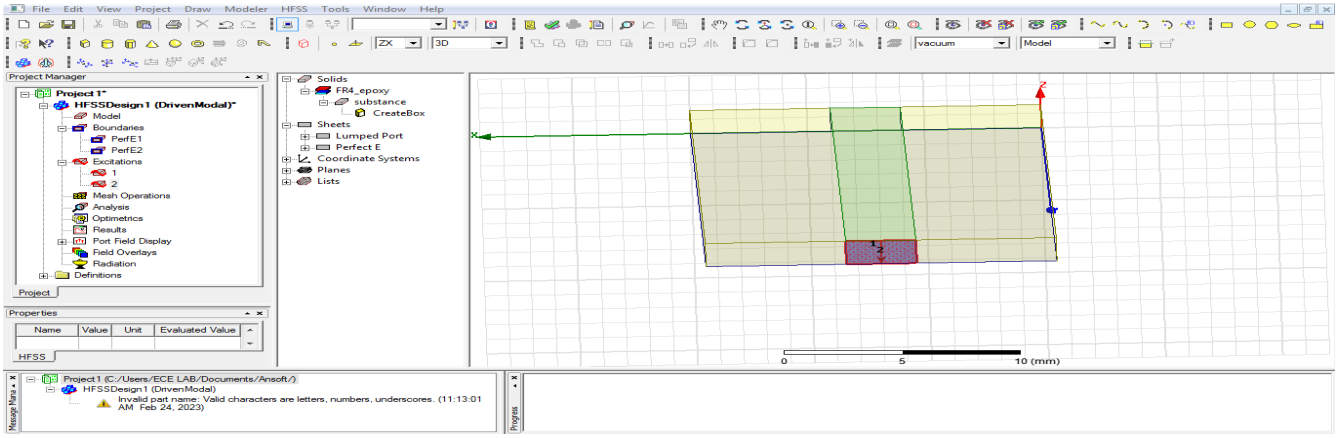


Rename rectangle2 double click port2 ok

Crate rectangle double click
Position - (18,60,1.6)
Axis - Y
X size- (-6mm)
Z size (-1.6mm)
ok



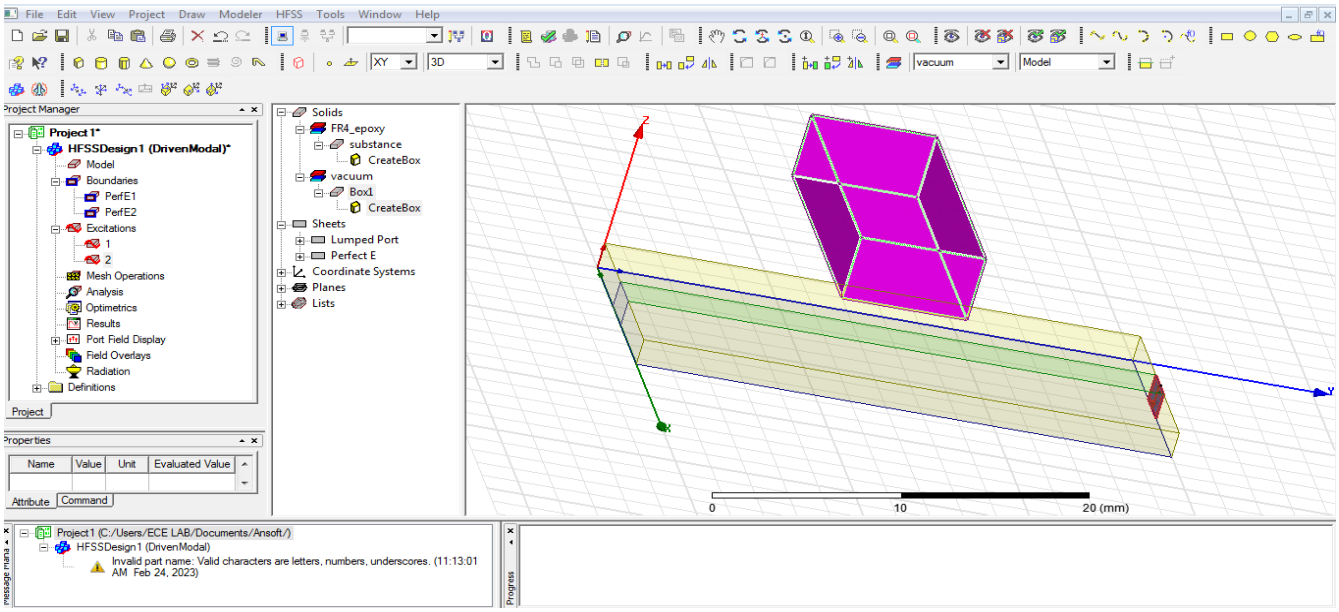
Select port2-right click assign excitation-lumped port-click-name 2-resistance - 50 ohms-next select none new line –draw a line-defined-next-full port impedance - 50 ohms – finish



Select **XY**

Design **radiation boundary**

Draw **box**



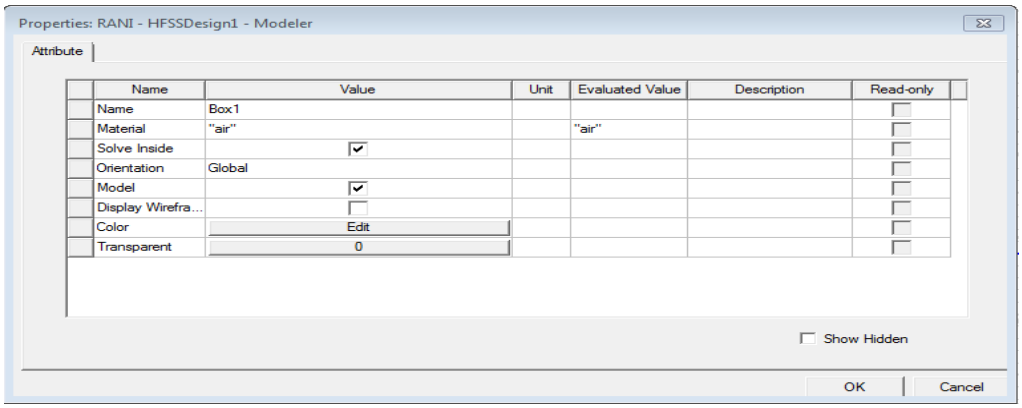
double click **box1**

rename

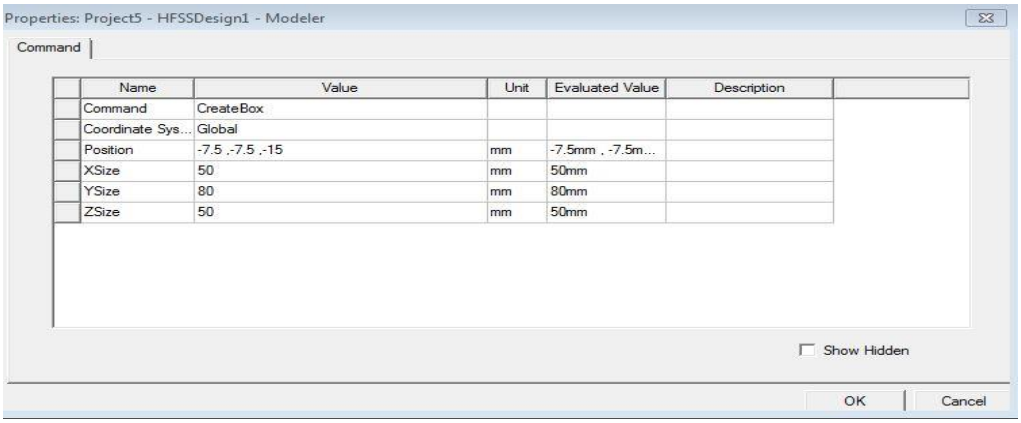
radiation boundary

Material – edit
air
ok
Colour
edit

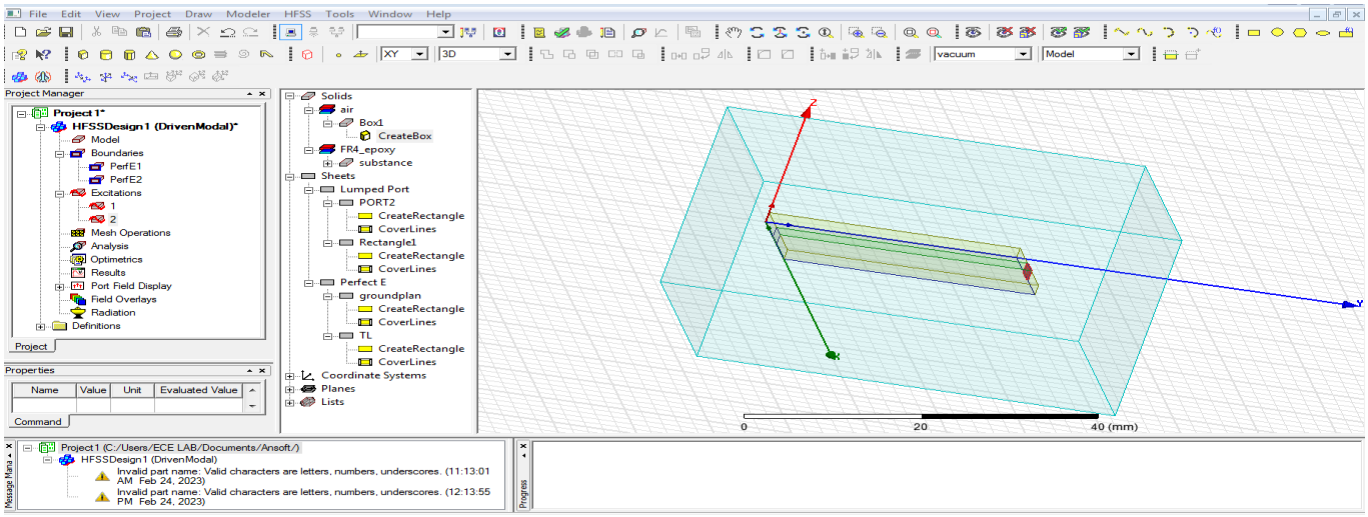
ok

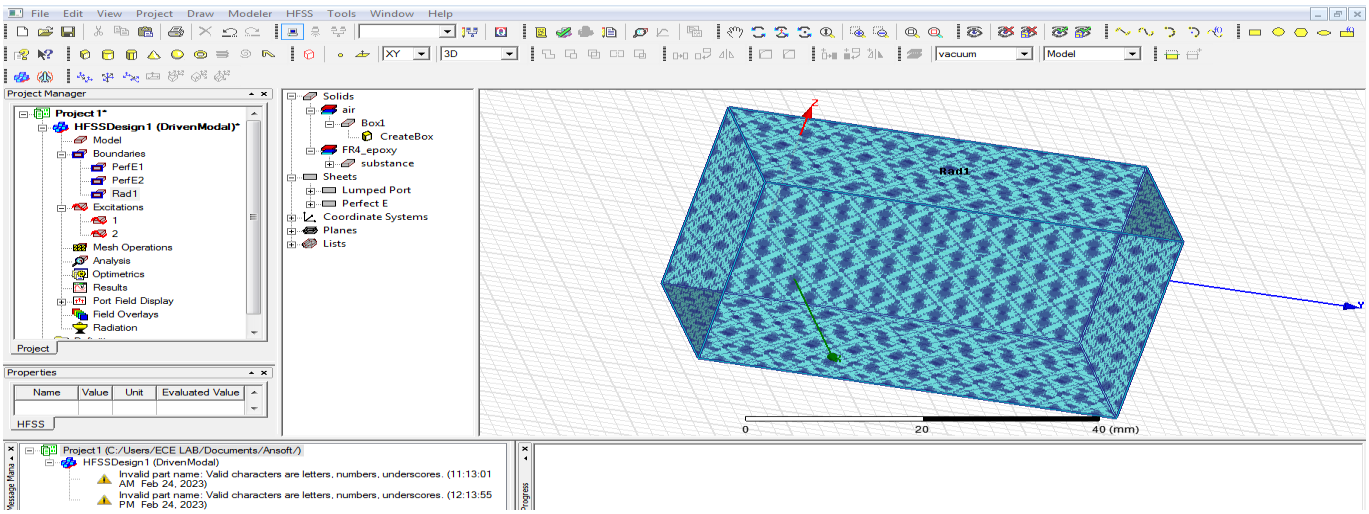


Double click
create box
Position-
(-7.5,-7.5,-15)
X size-
50mm
Y size-
80mm
Z size-
50mm
ok
Fit all



Click radiation boundary-right click-assign boundary –radiation click name **rad1** ok





Analysis –right click add solution setup click name setup1 solution frequency-2.4GHZ & 3.4GHZ

Maximum number of passes -12

Maximum Delta S-0.02 ok

Analysis right click-

setup1 right click –

add frequency sweep

Sweep type : fast

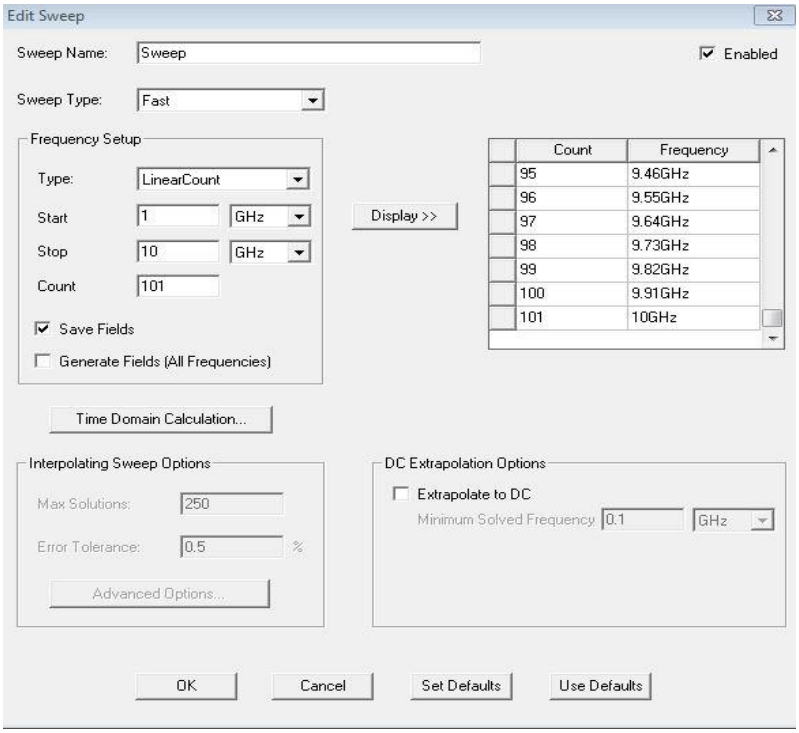
Type : linear count

Start freq:1GHZ

Stop freq:10GHZ

Count:101GHZ

Click display-see all frequencies ok



Result analysis(error checking)

Click double click **validity**

HFSS Design- design setting

3D model

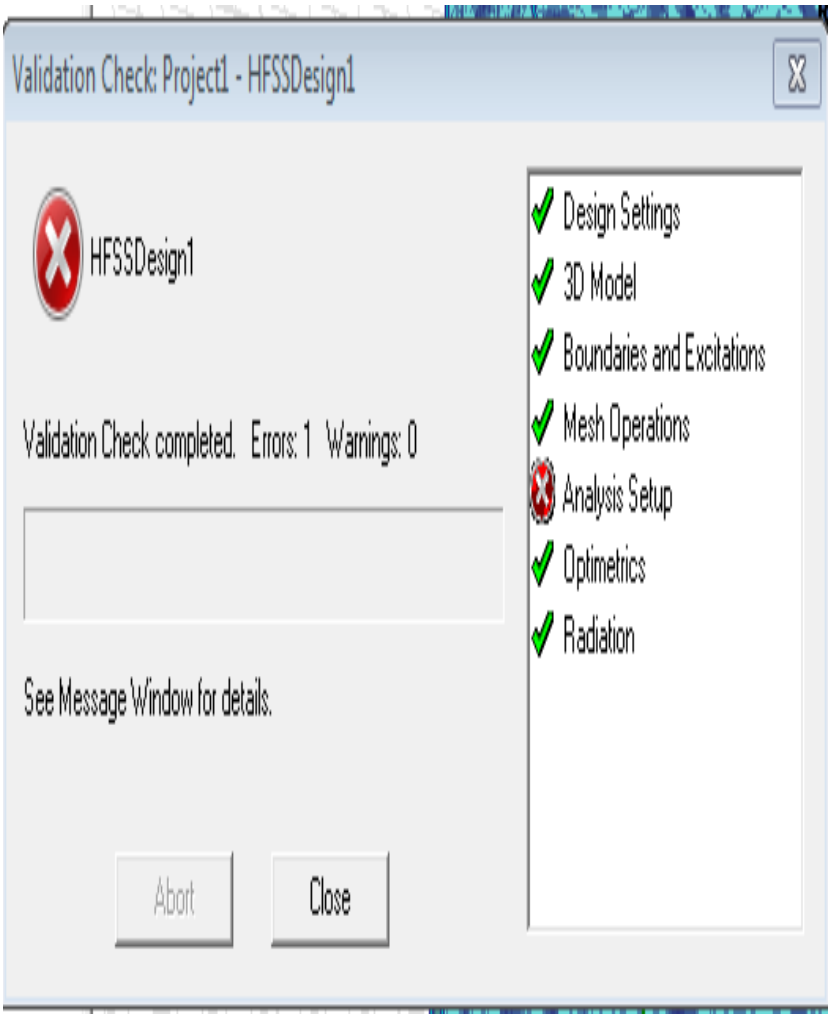
Boundaries and excitations

Mesh operation

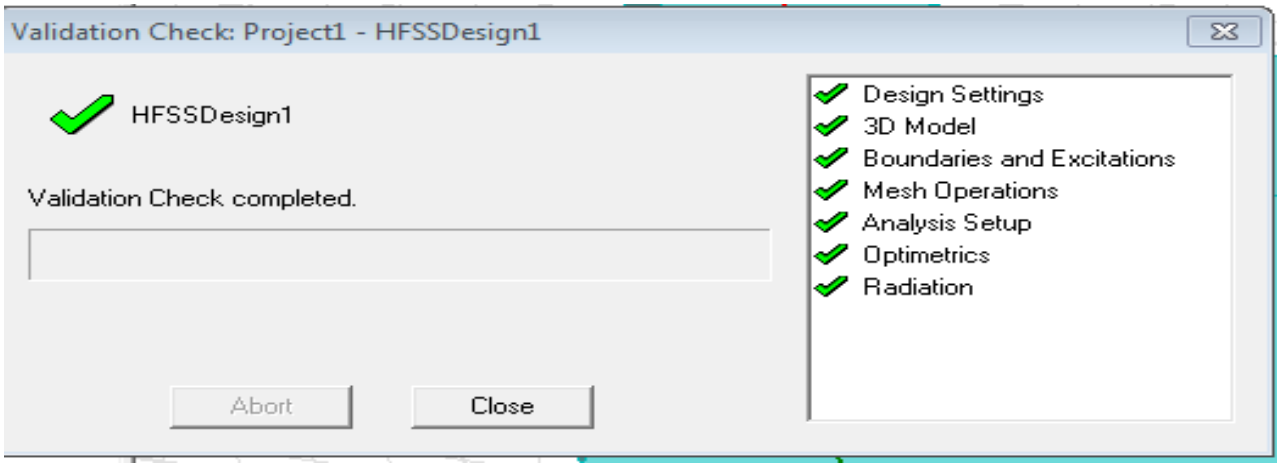
Analysis setup

Optimetrics

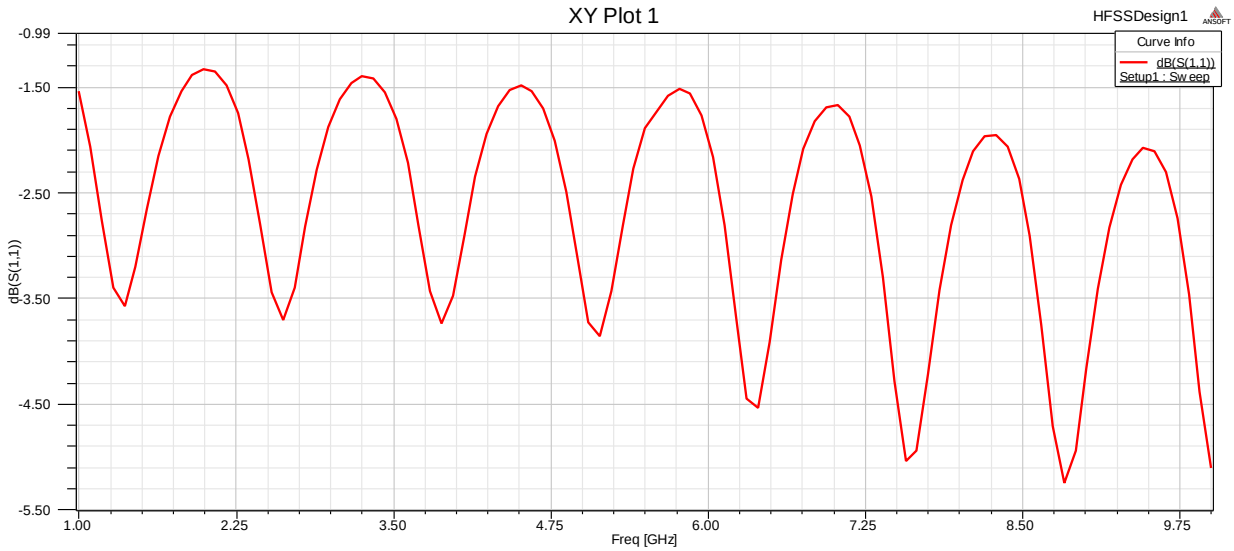
Radiation



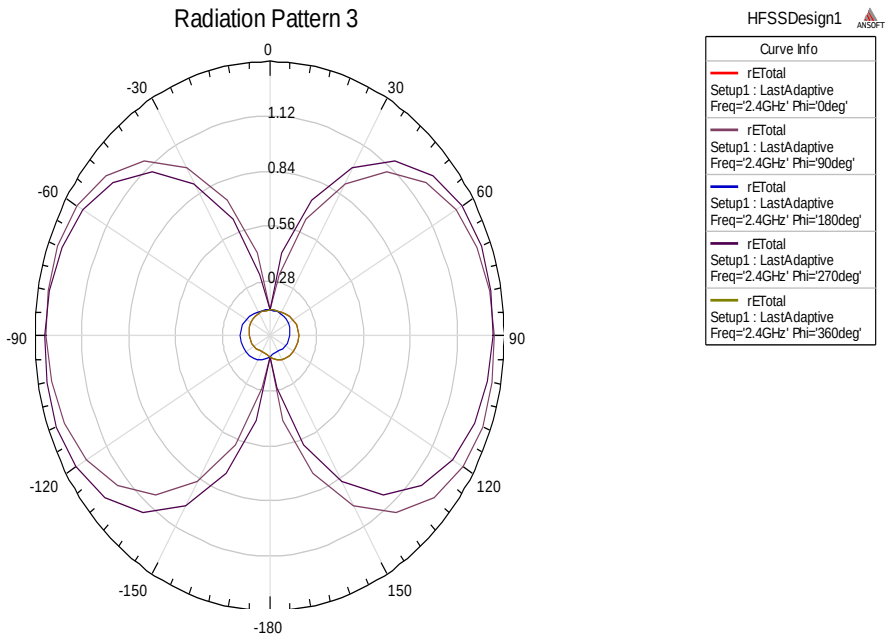
Next **analysis all**-any error rectify-ok

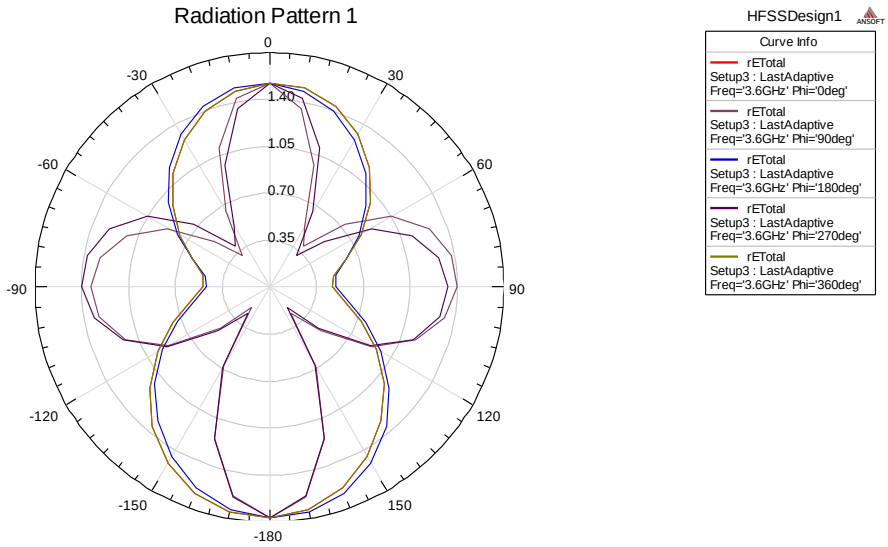


<u>Model Waveforms</u>: 2.4 GHZ (S Parameter)

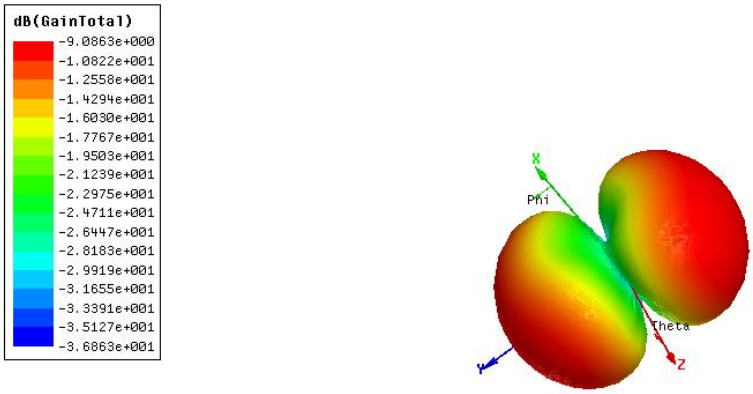


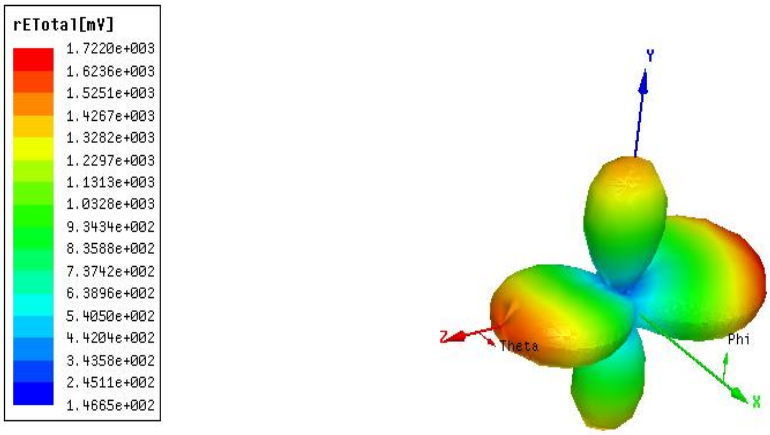
radiation pattern graphs (2.4GHz) & 3.8GHz

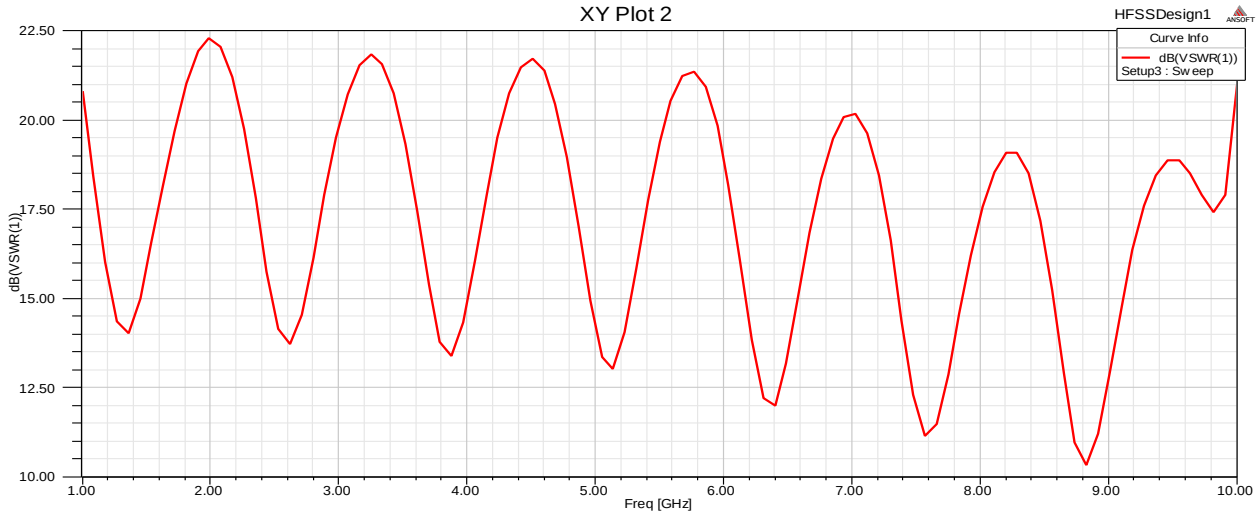




3d graphs **(2.4GHz)&3.8GHz**







vswr graph

Result:

Viva questions:

1. What is transmission line?

2. What is the frequency of microstrip transmission line?

3. What is standing wave pattern in transmission line?

4. What are the different types of microstrip transmission lines?

5. What is the formula of microstrip transmission line?

Exp: 04 **Date:**

S PARAMETER OF A MICRO STRIP TRANSMISSION LINE

Aim: Measure the S parameter of a Micro strip Transmission Line and plot the hoemalised impedance on a smith chart.

Apparatus Required:

1. **Computer**
2. **matlab software.**

Theory:

Procedure:

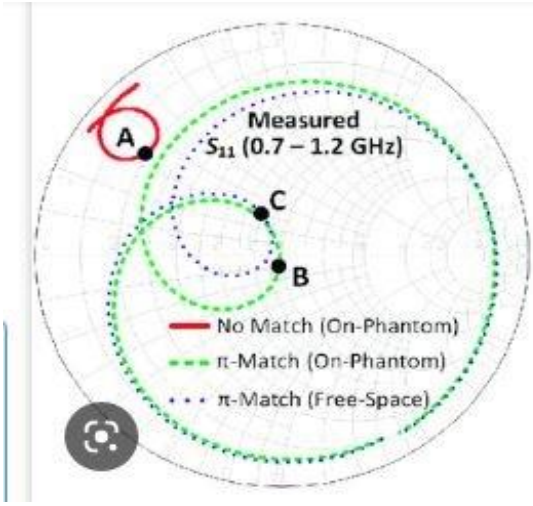
1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

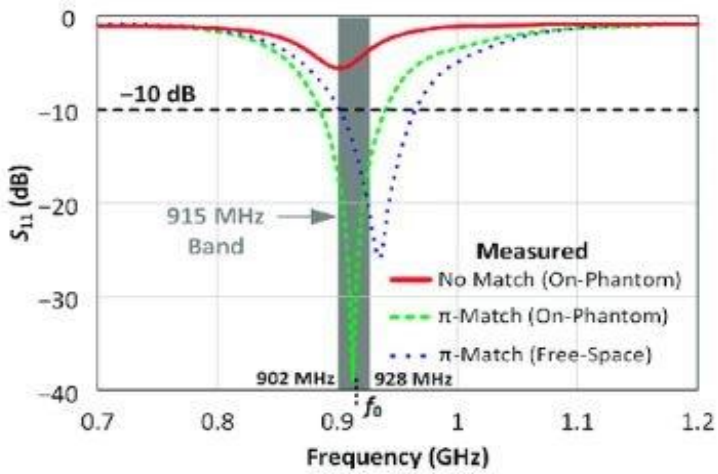
program:

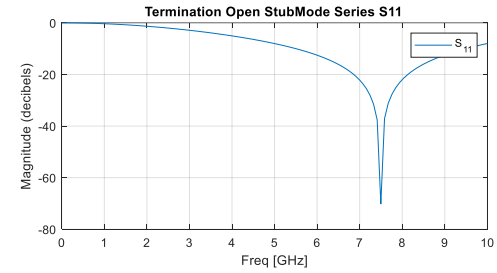
```
clear all
close all
clc
l=rfckt.txline('Termination','Open','StubMode','Series','z0',75)
k=analyze(l,logspace(5,10,1000))
figure
subplot(2,2,1)
plot(l,'S11','dB')
title('Termination Open StubMode Series S11')
subplot(2,2,2)
semilogy(l,'S12','dB')
title('Termination Open StubMode Series S12')
subplot(2,2,3)
```

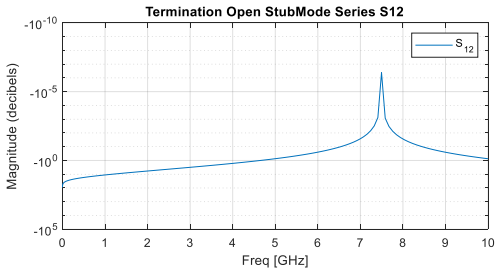
```
semilogy(l,'S21','dB')
title('Termination Open StubMode Series S21')
subplot(2,2,4)
plot(l,'S22','dB')
title('Termination Open StubMode Series S22')
j=rfckt.txline('Termination','Open','StubMode','Shunt','z0',75)
g=analyze(j,logspace(5,10,1000))
figure
subplot(2,2,1)
plot(j,'S11','dB')
title('Termination Open StubMode Shunt S11')
subplot(2,2,2)
semilogy(j,'S12','dB')
title('Termination Open StubMode Shunt S12')
subplot(2,2,3)
semilogy(j,'S21','dB')
title('Termination Open StubMode Shunt S21')
subplot(2,2,4)
plot(j,'S22','dB')
title('Termination Open StubMode Shunt S22')
m=rfckt.txline('Termination','Short','StubMode','Series','z0',75)
v=analyze(m,logspace(5,10,1000))
figure
subplot(2,2,1)
plot(m,'S11','dB')
title('Termination Short StubMode Series S11')
subplot(2,2,2)
semilogy(m,'S12','dB')
title('Termination Short StubMode Series S12')
subplot(2,2,3)
semilogy(m,'S21','dB')
title('Termination Short StubMode Series S21')
subplot(2,2,4)
plot(m,'S22','dB')
title('Termination Short StubMode Series S22')
n=rfckt.txline('Termination','Short','StubMode','Shunt','z0',75)
b=analyze(n,logspace(5,10,1000))
figure
subplot(2,2,1)
plot(n,'S11','dB')
title('Termination Short StubMode Shunt S11')
subplot(2,2,2)
semilogy(n,'S12','dB')
title('Termination Short StubMode Shunt S12')
subplot(2,2,3)
semilogy(n,'S21','dB')
title('Termination Short StubMode Shunt S21')
subplot(2,2,4)
plot(n,'S22','dB')
title('Termination Short StubMode Shunt S22')
```

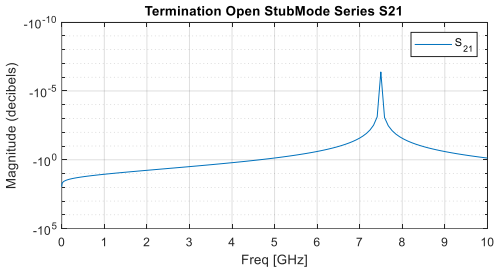
Model Waveforms:

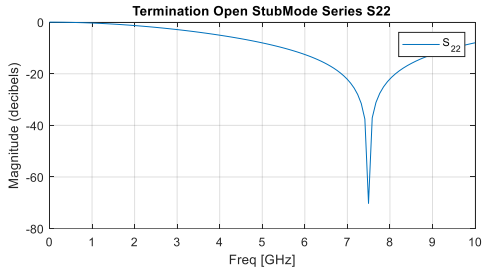


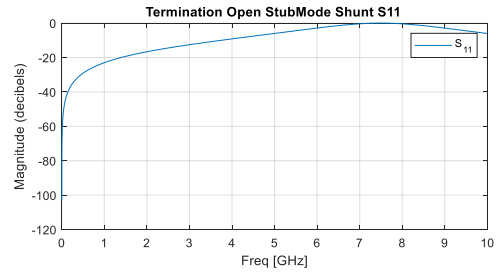


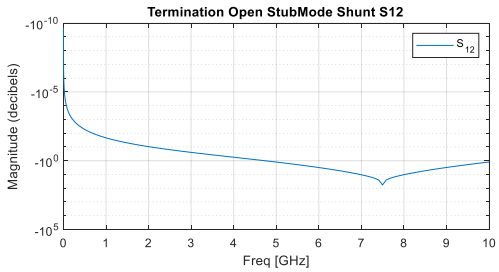


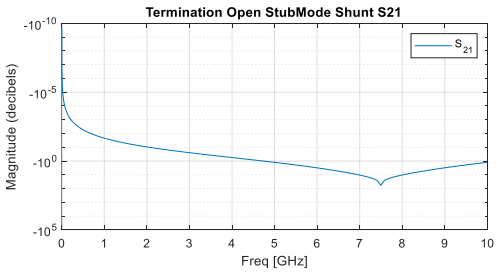


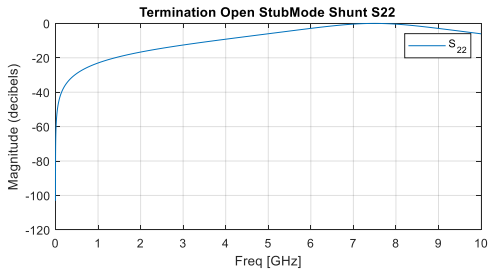


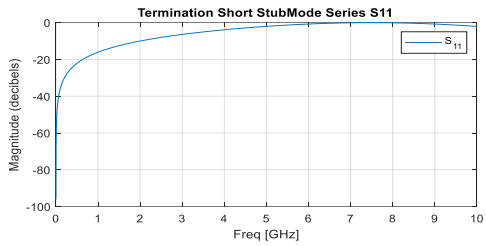


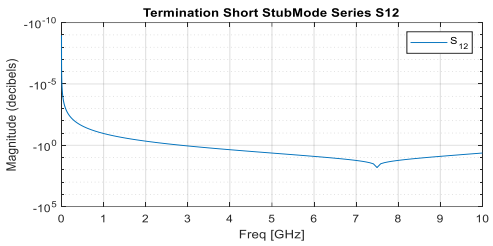


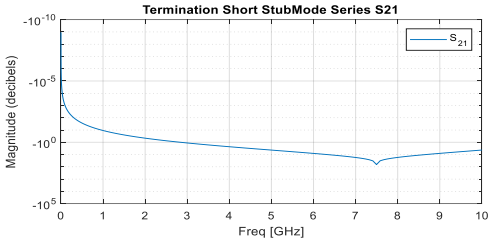


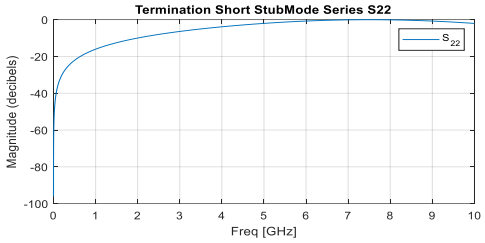


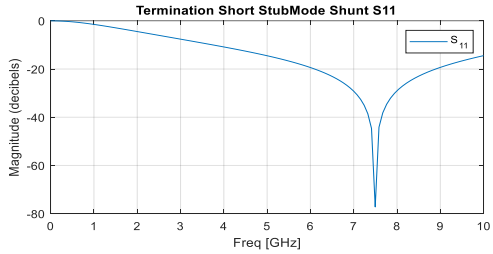


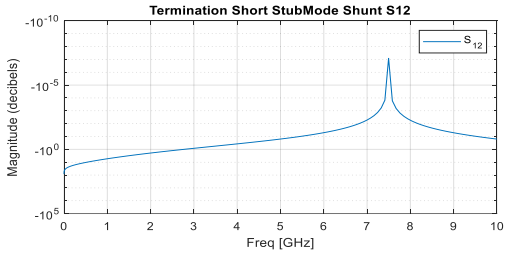


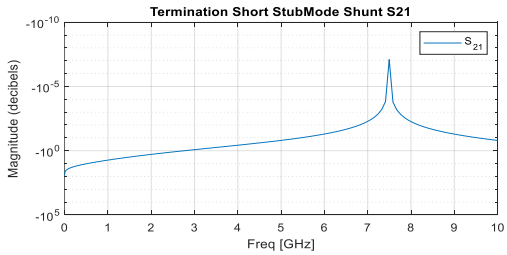


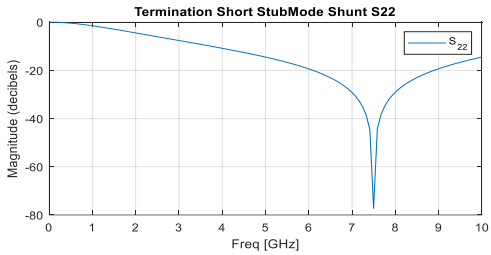












Calculation:

```
l =   rfckt.txline with properties:
      LineLength: 0.0100
        StubMode: 'Series'
     Termination: 'Open'
            Freq: 1.0000e+09
              Z0: 75
              PV: 299792458
            Loss: 0
         IntpType: 'Linear'
           nPort: 2
   AnalyzedResult: []
            Name: 'Transmission Line'
k =   rfckt.txline with properties:
      LineLength: 0.0100
        StubMode: 'Series'
     Termination: 'Open'
            Freq: 1.0000e+09
              Z0: 75
```

```
              PV: 299792458
            Loss: 0
         IntpType: 'Linear'
            nPort: 2
    AnalyzedResult: [1×1 rfdata.data]
            Name: 'Transmission Line'
j =    rfckt.txline with properties:
       LineLength: 0.0100
         StubMode: 'Shunt'
      Termination: 'Open'
            Freq: 1.0000e+09
              Z0: 75
              PV: 299792458
            Loss: 0
         IntpType: 'Linear'
            nPort: 2
    AnalyzedResult: []
            Name: 'Transmission Line'
g =    rfckt.txline with properties:
       LineLength: 0.0100
         StubMode: 'Shunt'
      Termination: 'Open'
            Freq: 1.0000e+09
              Z0: 75
              PV: 299792458
            Loss: 0
         IntpType: 'Linear'
            nPort: 2
    AnalyzedResult: [1×1 rfdata.data]
            Name: 'Transmission Line'
m =    rfckt.txline with properties:
       LineLength: 0.0100
         StubMode: 'Series'
      Termination: 'Short'
            Freq: 1.0000e+09
              Z0: 75
              PV: 299792458
            Loss: 0
         IntpType: 'Linear'
            nPort: 2
    AnalyzedResult: []
            Name: 'Transmission Line'
v =    rfckt.txline with properties:
       LineLength: 0.0100
         StubMode: 'Series'
      Termination: 'Short'
            Freq: 1.0000e+09
              Z0: 75
              PV: 299792458
            Loss: 0
         IntpType: 'Linear'
            nPort: 2
    AnalyzedResult: [1×1 rfdata.data]
```

```
            Name: 'Transmission Line'
n =    rfckt.txline with properties:
       LineLength: 0.0100
         StubMode: 'Shunt'
      Termination: 'Short'
             Freq: 1.0000e+09
               Z0: 75
               PV: 299792458
             Loss: 0
         IntpType: 'Linear'
            nPort: 2
   AnalyzedResult: []
            Name: 'Transmission Line'
b =    rfckt.txline with properties:
       LineLength: 0.0100
         StubMode: 'Shunt'
      Termination: 'Short'
             Freq: 1.0000e+09
               Z0: 75
               PV: 299792458
             Loss: 0
         IntpType: 'Linear'
            nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
            Name: 'Transmission Line'
```

Result:

Conclusion:

Viva questions:

1. Write applications of Smith chart?

2. What is the need of S-parameters in transmission line?

3. What is normalized impedance?

4. Write properties of S-parameters?

5. Define losses in S-parameters?

Exp: 05 **Date:**

MICROSTRIP INDUCTOR AND CAPACITOR

Aim: Design of microstrip inductor and capacitor.

Apparatus Required:

1. **Computer**
2. **matlab software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

program:

```
clc
clear all
close all
h = rfckt.seriesrlc('L', 4e-06, 'C', 5e-7);
y = analyze(h, logspace(4, 10, 500));
g = rfckt.shuntrlc('L', 4.7e-06, 'C', 5e-07);
z = analyze(g, logspace(4, 10, 500));
k = analyze(h, logspace(4, 10, 1000));
figure(1)
subplot(3, 2, 1)
loglog(h, 'S21', 'dB')
title('SeriesRLC')
subplot(3, 2, 2)
loglog(h, 'S21', 'angle')
title('SeriesRLC Phase')
subplot(3, 2, 3)
loglog(g, 'S21', 'dB')
title('ShuntRLC')
subplot(3, 2, 4)
loglog(g, 'S21', 'angle')
title('ShuntRLC Phase')
subplot(3, 2, 5)
```

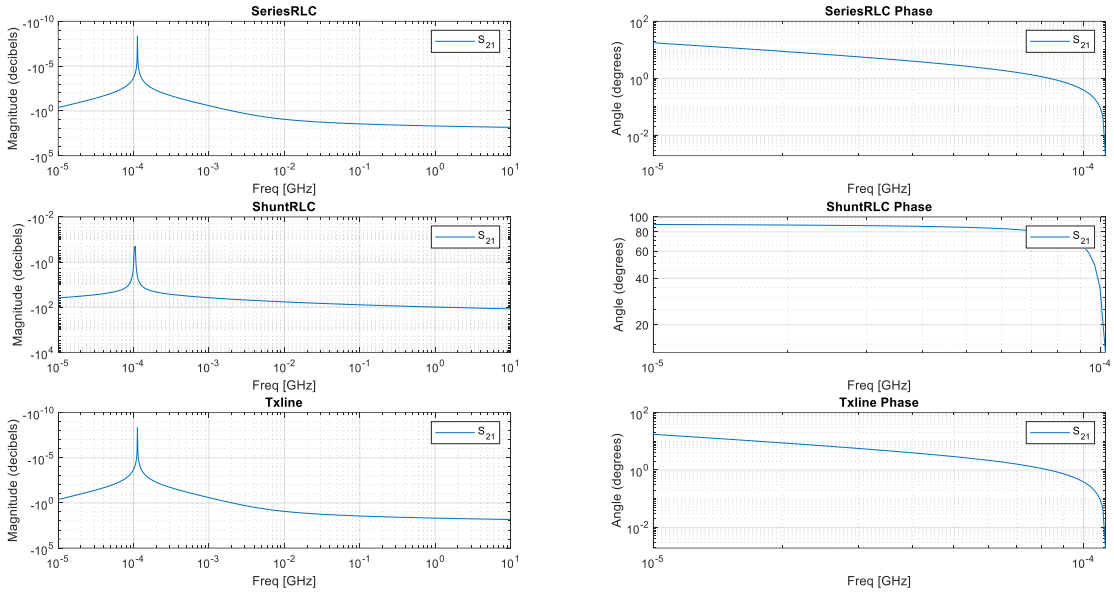
```
loglog(k, 'S21', 'dB')
title('Txline')
subplot(3, 2, 6)
loglog(k, 'S21', 'angle')
title('Txline Phase')
```

Calculation:

```
h =   rfckt.seriesrlc with properties:
            R: 0
            L: 4.0000e-06
            C: 5.0000e-07
         nPort: 2
  AnalyzedResult: []
         Name: 'Series RLC'
y =   rfckt.seriesrlc with properties:
            R: 0
            L: 4.0000e-06
            C: 5.0000e-07
         nPort: 2
  AnalyzedResult: [1×1 rfdata.data]
         Name: 'Series RLC'
g =   rfckt.shuntrlc with properties:
            R: Inf
            L: 4.7000e-06
            C: 5.0000e-07
         nPort: 2
  AnalyzedResult: []
         Name: 'Shunt RLC'
z =   rfckt.shuntrlc with properties:
            R: Inf
            L: 4.7000e-06
            C: 5.0000e-07
         nPort: 2
  AnalyzedResult: [1×1 rfdata.data]
         Name: 'Shunt RLC'
k =   rfckt.seriesrlc with properties:
            R: 0
            L: 4.0000e-06
            C: 5.0000e-07
         nPort: 2
  AnalyzedResult: [1×1 rfdata.data]
         Name: 'Series RLC'
```

Result:

Model Waveforms:



Viva questions:

1. Give the physical dimensions of the spiral inductor?

2. Write the inductance equation in microstrip?

3. Write applications of spiral inductor?

4. Write the capacitance equation in microstrip?

5. Write the applications of capacitor microstrip?

Exp: 06 **Date:**

IMPEDANCE MATCHING NETWORK

Aim: Design of impedance matching network.

Apparatus Required:

1. **Computer**
2. **Mat lab (19b) software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

program:

```
close all
clc
z0=50
d=1
c=3*10e8
B=-500:10:500
zin= -(1i*(z0*cotd(rad2deg(B*d))));
figure
plot(B,abs(zin))
xlabel('B')
ylabel('Zin')
title('Input impedance of an open circuited Tx line')
```

Calculation:

z0 = 50

d = 1

c = 3.0000e+09

B = Columns 1 through 20

-500 -490 -480 -470 -460 -450 -440 -430 -420 -410 -400 -390 -380 -370 -360 -350 -340 -330 -320 -310

Columns 21 through 40

-300 -290 -280 -270 -260 -250 -240 -230 -220 -210 -200 -190 -180 -170 -160 -150 -140 -130 -120 -110

Columns 41 through 60

-100 -90 -80 -70 -60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90

Columns 61 through 80

100 110 120 130 140 150 160 170 180 190 200 210 220 230 240 250 260 270 280 290

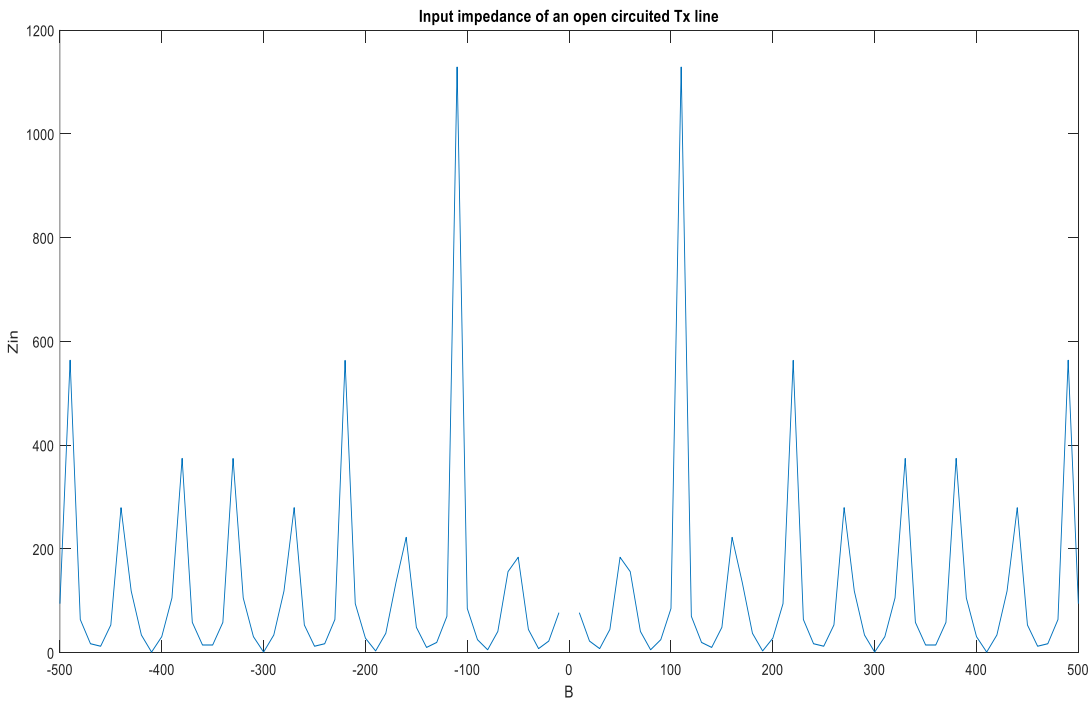
Columns 81 through 100

300 310 320 330 340 350 360 370 380 390 400 410 420 430 440 450 460 470 480 490

Column 101

500

Model Waveforms:



Result:

Viva questions:

1. What is the need for impedance matching?

2. Explain single and double stub tuners?

3. Write all the matching methods?

4. Draw three element matching network?

5. Draw formation of matching network?

Exp: 07 **Date:**

RF BJT AMPLIFIER AND LNA

Aim: Design and characterization of RF BJT Amplifier and LNA

Apparatus Required:

1. **Computer**
2. **Mat lab (19b) software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

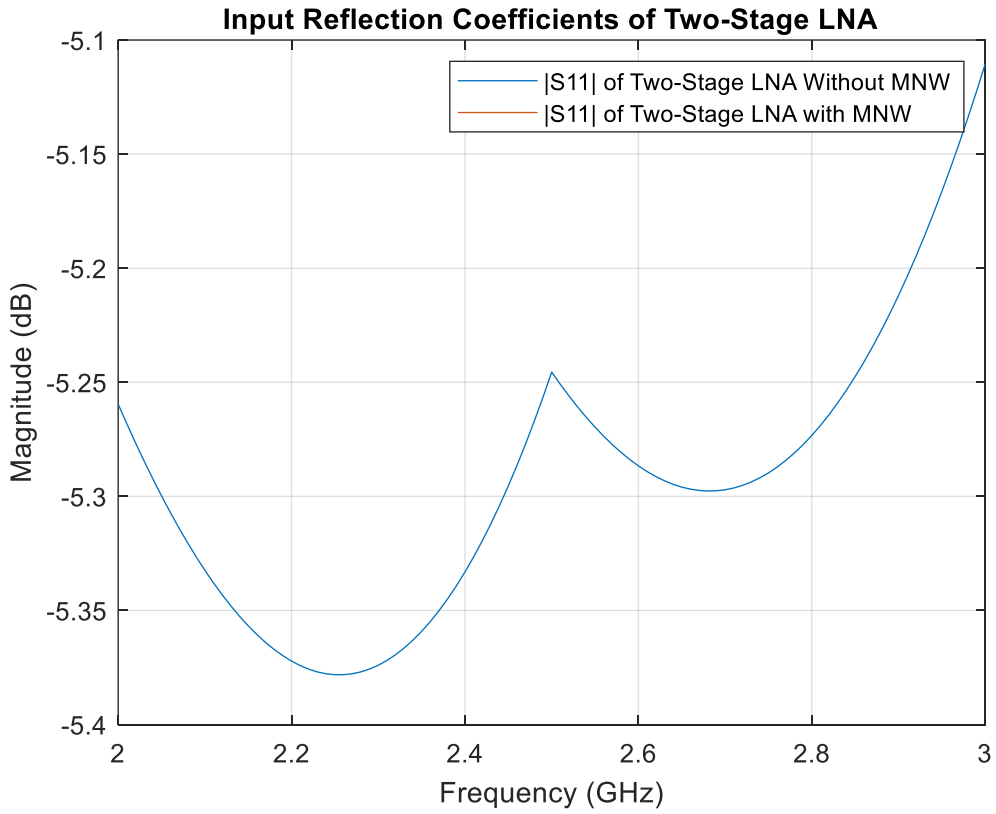
program: RfBJT Amplifier

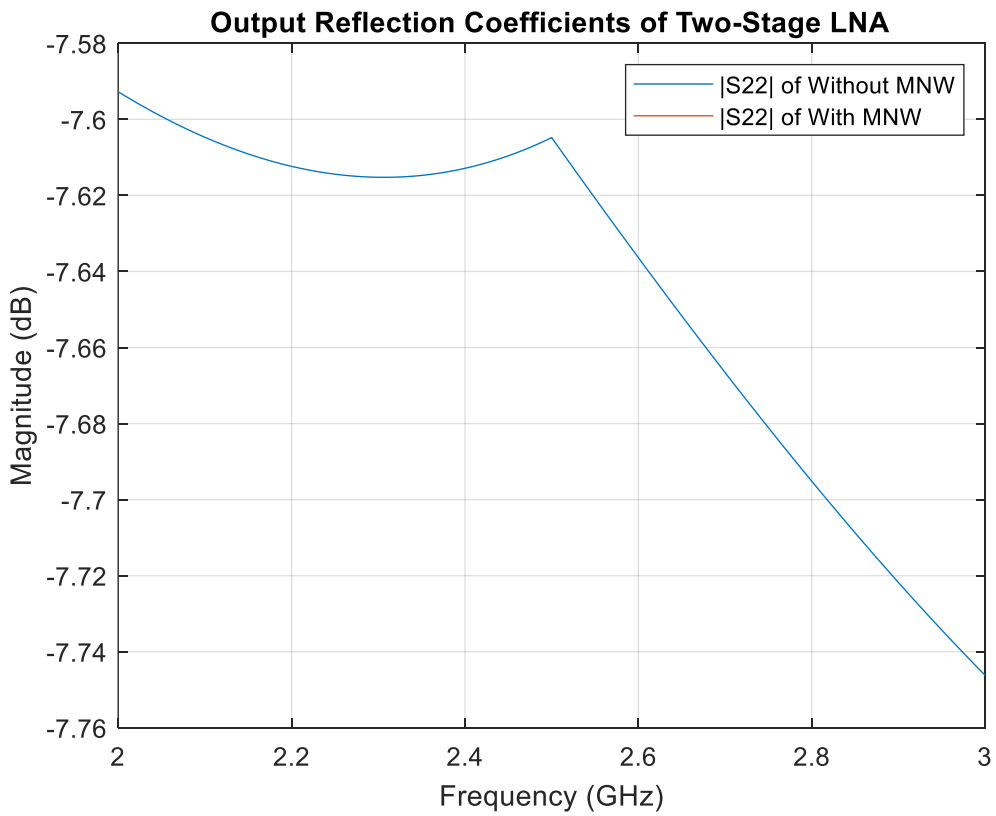
```
clear all
close all
clc
h=rfckt.amplifier()
s=sparameters(h);
figure(),
subplot(2,2,1)
plot(h,'S11','dB')
title('Amplifier S11'),
subplot(2,2,2),
plot(h,'S12','dB')
title('Amplifier S12')
subplot(2,2,3),
plot(h,'S21','dB')
title('Amplifier S11')
subplot(2,2,4),
plot(h,'S22','dB')
title('Amplifier S12')
figure();
plot(h,'Gt','dB');
title('Amplifier gain');
```

program: RfBJT LNAmplifier

```
clear all;
close all;
clc;
TL1=txlineMicrostrip('Width',3.41730e-3,'Height',1.524e-
3,'EpsilonR',3.48,'LossTangent',0.0037,'LineLength',8.9e-3,'Thickness',0.0035e-
3,'StubMode','Shunt','Termination','Open');
TL2=txlineMicrostrip('Width',3.41730e-3,'Height',1.524e-
3,'EpsilonR',3.48,'LossTangent',0.0037,'LineLength',14.7e3,'Thickness',0.0035e-3);
amp1 = nport('f551432p.s2p');
freq=2e9:10e6:3e9;
casamp=circuit([amp1,clone(amp1)],'amplifiers'); % amplifier ciruit without MNW.
S2=sparameters(casamp,freq);
TL3=txlineMicrostrip('Width',3.41730e-3,'Height',1.524e-
3,'EpsilonR',3.48,'LossTangent',0.0037,'LineLength',22.47e-3,'Thickness',0.0035e-3);
TL4=txlineMicrostrip('Width',3.41730e-3,'Height',1.524e-
3,'EpsilonR',3.48,'LossTangent',0.0037,'LineLength',5.66e-3,'Thickness',0.0035e-
3,'StubMode','Shunt','Termination','Open');
c=circuit([TL1, TL2,clone(amp1),clone(amp1),TL3, TL4]); % two-stage LNA with MNW
figure
S3=sparameters(c,freq);
rfplot(S2,1,1)
hold on;
rfplot(S3,1,1)
legend('|S11| of Two-Stage LNA Without MNW','|S11| of Two-Stage LNA with MNW');
title('Input Reflection Coefficients of Two-Stage LNA');
grid on;
figure
rfplot(S2,2,2)
hold on;
rfplot(S3,2,2)
legend('|S22| of Without MNW','|S22| of With MNW');
title('Output Reflection Coefficients of Two-Stage LNA');
grid on;
```

OUTPUT:


Input Reflection Coefficients of Two-Stage LNA
|S11| of Two-Stage LNA Without MNW
|S11| of Two-Stage LNA with MNW
Magnitude (dB)
Frequency (GHz)
-5.1
-5.15
-5.2
-5.25
-5.3
-5.35
-5.4
2
2.2
2.4
2.6
2.8
3

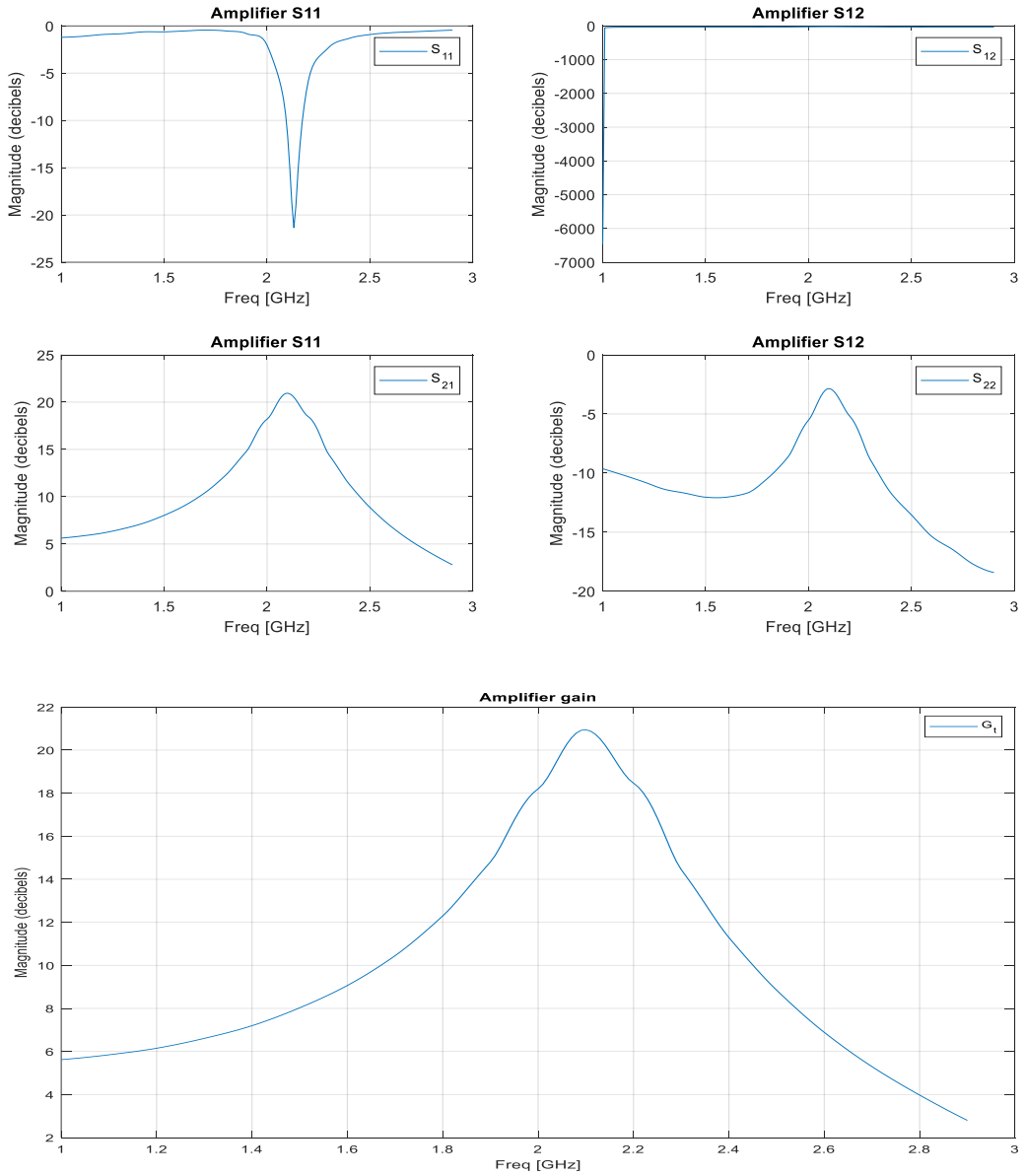

Output Reflection Coefficients of Two-Stage LNA
|S22| of Without MNW
|S22| of With MNW
Magnitude (dB)
Frequency (GHz)
-7.58
-7.6
-7.62
-7.64
-7.66
-7.68
-7.7
-7.72
-7.74
-7.76
2
2.2
2.4
2.6
2.8
3

Calculation:

RfBJT Amplifier :

```
h =    rfckt.amplifier with properties:
        NoiseData: [1×1 rfdata.noise]
     NonlinearData: [1×1 rfdata.power]
         IntpType: 'Linear'
       NetworkData: [1×1 rfdata.network]
             nPort: 2
    AnalyzedResult: [1×1 rfdata.data]
              Name: 'Amplifier'
```

Model Waveforms:



Result:

Viva questions:

1. What is low noise amplifier?

2. Give the noise figure formula of LNA?

3. Write LNA matching technique?

4. Write RF BJT amplifier biasing methods?

5. Write RF BJT and LNA applications?

Exp: 08 **Date:**

RF FILTER

Aim: Design of low pass, high pass, band pass and band stop filter at RF.

Apparatus Required:

1. **Computer**
2. **Matlab(19b) software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

Calculation:

(bpf&bsf)

```
filter1 =    rfckt.lcbandpasspi with properties:
              L: [2×1 double]
              C: [2×1 double]
          nPort: 2
   AnalyzedResult: []
           Name: 'LC Bandpass Pi'
ans =    rfckt.lcbandpasspi with properties:
              L: [2×1 double]
              C: [2×1 double]
          nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
           Name: 'LC Bandpass Pi'
filter2 =    rfckt.lcbandstoppi with properties:
  L: [2×1 double]
              C: [2×1 double]
          nPort: 2
   AnalyzedResult: []
           Name: 'LC Bandstop Pi'
ans =    rfckt.lcbandstoppi with properties:
```

```
            L: [2×1 double]
            C: [2×1 double]
         nPort: 2
  AnalyzedResult: [1×1 rfdata.data]
           Name: 'LC Bandstop Pi'
s1 =  sparameters: S-parameters object
     NumPorts: 2
  Frequencies: [1000×1 double]
   Parameters: [2×2×1000 double]
    Impedance: 50.0000 + 0.0000i
 rfparam(obj,i,j) returns S-parameter Sij
s2 =  sparameters: S-parameters object
     NumPorts: 2
  Frequencies: [1000×1 double]
   Parameters: [2×2×1000 double]
    Impedance: 50.0000 + 0.0000i
 rfparam(obj,i,j) returns S-parameter Sij
```

(lpf&hpf):

```
h =   rfckt.lclowpasstee with properties:
            L: 1.0000e-09
            C: 2.0000e-11
         nPort: 2
  AnalyzedResult: []
           Name: 'LC Lowpass Tee'
m =   rfckt.lclowpasstee with properties:
            L: 1.0000e-09
            C: 2.0000e-11
         nPort: 2
  AnalyzedResult: [1×1 rfdata.data]
           Name: 'LC Lowpass Tee'
l =   rfckt.lchighpasstee with properties:
            L: 5.0000e-07
            C: 2.0000e-05
         nPort: 2
  AnalyzedResult: []
           Name: 'LC Highpass Tee'
n =   rfckt.lchighpasstee with properties:
            L: 5.0000e-07
            C: 2.0000e-05
         nPort: 2
  AnalyzedResult: [1×1 rfdata.data]
           Name: 'LC Highpass Tee'
```

<u>program:</u> (bpf&bsf)

```
clear all
close all
clc
filter1 = rfckt.lcbandpasspi('C',[1e-12 4e-12],'L',[2e-9 2.5e-9]);
analyze(filter1,logspace(5,10,1000));
filter2 = rfckt.lcbandstoppi('C',[1e-12 4e-12],'L',[2e-9 2.5e-9]);
analyze(filter2,logspace(5,10,1000));
s1=sparameters(filter1);
s2=sparameters(filter2);
figure()
```

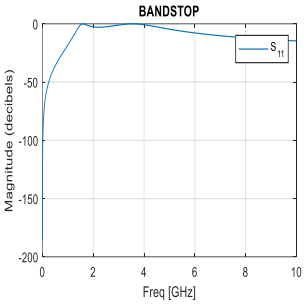
```
subplot(2,2,1);
plot(filter1,'S11','dB');
title('BANDPASS');
subplot(2,2,2);
plot(filter1,'S12','dB');
title('BANDPASS');
subplot(2,2,3);
plot(filter1,'S21','dB');
title('BANDPASS');
subplot(2,2,4);
plot(filter1,'S22','dB');
title('BANDPASS');
figure()
subplot(2,2,1);
plot(filter2,'S11','dB');
title('BANDSTOP');
subplot(2,2,2);
plot(filter2,'S12','dB');
title('BANDSTOP');
subplot(2,2,3);
plot(filter2,'S21','dB');
title('BANDSTOP');
subplot(2,2,4);
plot(filter2,'S22','dB');
title('BANDSTOP');
```

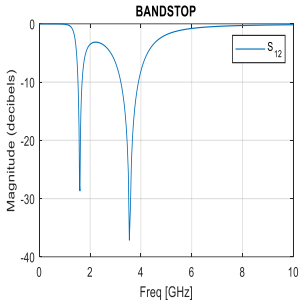
program: (lpf&hpf)

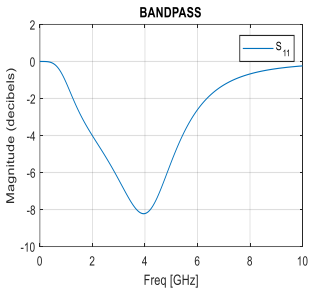
```
clear all
close all
clc
h = rfckt.lclowpasstee('C',20e-12,'L',10e-10)
m = analyze(h,logspace(1,9,1000))
s = extract(h,'S-parameters',75);
l = rfckt.lchighpasstee('C',2e-5,'L',5e-7)
n = analyze(l,logspace(1,9,1000))
figure
subplot(2,2,1)
plot(h,'S11','dB')
title('S11 lowpass')
subplot(2,2,2)
semilogy(h,'S12','dB')
title('S12 lowpass')
subplot(2,2,3)
semilogy(h,'S21','dB')
title('S21 lowpass')
subplot(2,2,4)
plot(h,'S22','dB')
title('S22 lowpass')
figure
subplot(2,2,1)
plot(l,'S11','dB')
title('S11 highpass')
subplot(2,2,2)
semilogy(l,'S12','dB')
title('S12 highpass')
subplot(2,2,3)
semilogy(l,'S21','dB')
title('S21 highpass')
subplot(2,2,4)
```

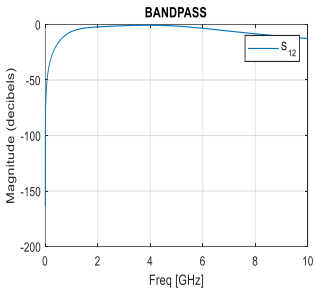
```
plot(l,'S22','dB')
title('S22 highpass')
```

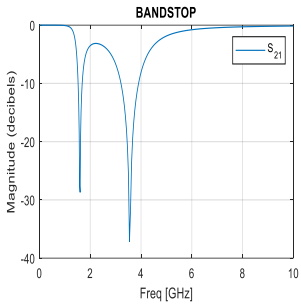
Model Waveforms:

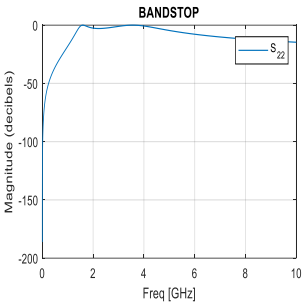


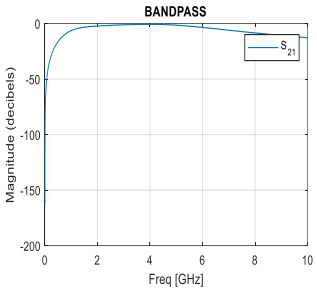


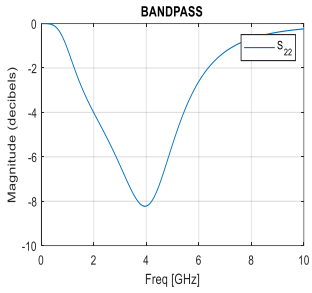


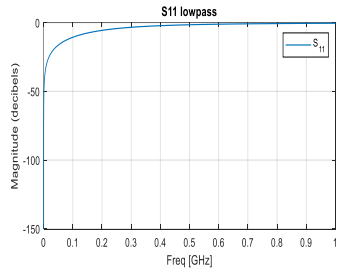


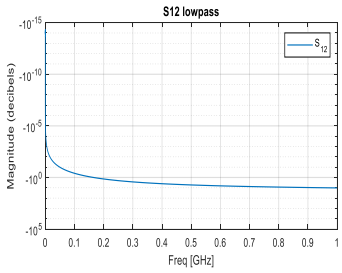


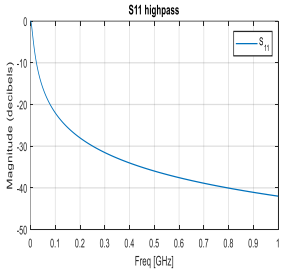


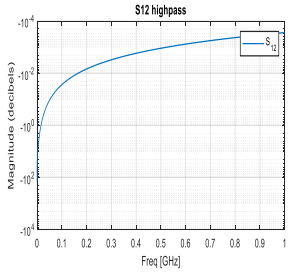


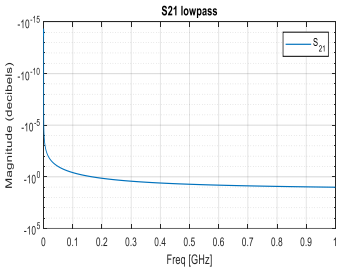


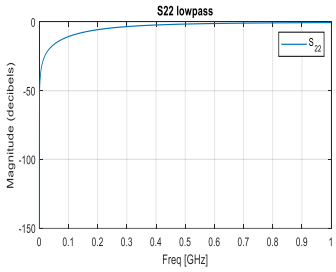


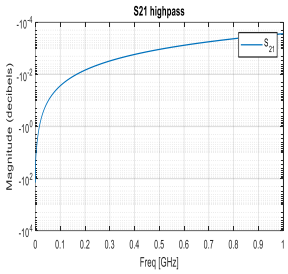


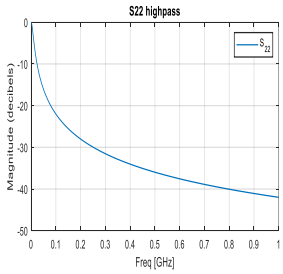


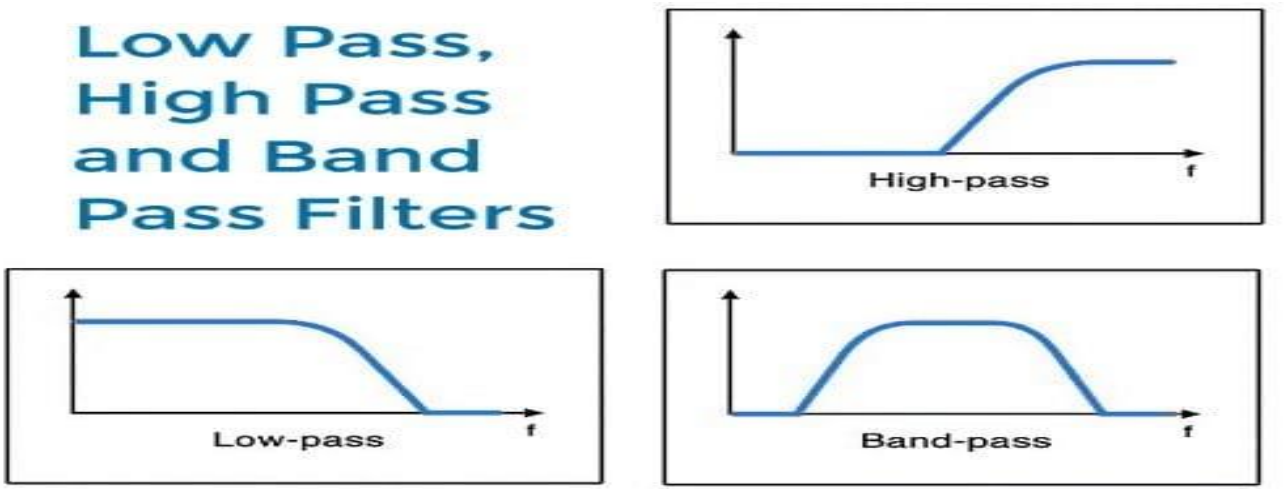












Result:

Viva questions:

1. Define all pass filters?

2. Define notch filters?

3. Write characteristics of filter at RF filters?

4. Applications of LPF,HPF and BPF?

5. What is the need of RF filters?

Exp: 09 **RF MIXER** **Date:**

Aim: Design and characterization of RF Mixer.

Apparatus Required:

1. **Computer**
2. **Matlab(19b) software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

Program:

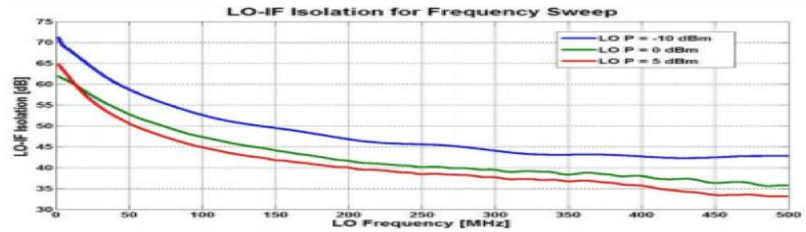
```
clc;
clear all;
close all;
%Create RF and IF bandpass filters
fi1 = read(rfckt.passive,'RFBudget_RF.s2p');
fi2 = read(rfckt.passive,'RFBudget_IF.s2p');
%Create RF and IF amplifiers
ai1 = rfckt.amplifier('NetworkData', ...
   rfdata.network('Type','S','Freq',2.1e9,'Data',[0,0;3.98,0]), ...
   'NoiseData',2,'NonlinearData',35);
ai2 = rfckt.amplifier('NetworkData', ...
   rfdata.network('Type','S','Freq',2.1e9,'Data',[0,0;31.66,0]), ...
   'NoiseData',8,'NonlinearData',37);
%Create a demodulator and microstrip transmission line
mi1 = rfckt.mixer('NetworkData', ...
   rfdata.network('Type','S','Freq',2.1e9,'Data',[0,0;0.501,0]),...
   "MixerType",'Downconverter','FLO',2.03e9,'NoiseData',4,'NonlinearData',50);
tx1 = rfckt.microstrip('Thickness',0.0075e-6);
%Cascade the circuit
c = rfckt.cascade('Ckts',{fi1 ai1 mi1 fi2 ai2 tx1});
% Analyze the cascaded circuit
analyze(c,linspace(2.08e9,2.12e9,100));
% Plot the magnitude of the S21 parameter for the cascade
plot(c,'s21','db')
hold on;
plot(c,'s11','db')
```

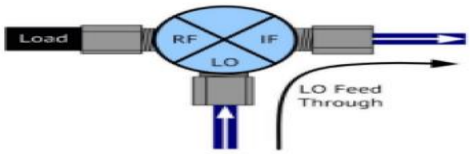
Calculation:

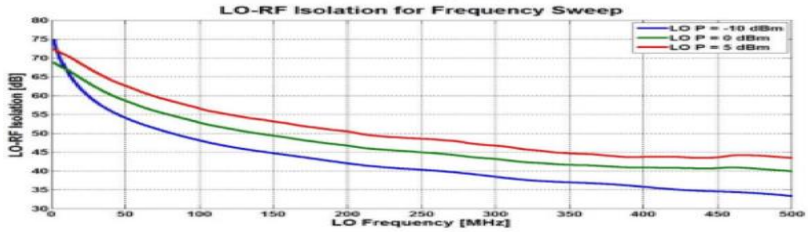
```
fi1 =    rfckt.passive with properties:
         IntpType: 'Linear'
      NetworkData: [1×1 rfdata.network]
            nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
             Name: 'Passive'
fi2 =    rfckt.passive with properties:
         IntpType: 'Linear'
      NetworkData: [1×1 rfdata.network]
            nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
             Name: 'Passive'
ai1 =    rfckt.amplifier with properties:
        NoiseData: 2
    NonlinearData: 35
         IntpType: 'Linear'
      NetworkData: [1×1 rfdata.network]
            nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
             Name: 'Amplifier'
ai2 =    rfckt.amplifier with properties:
        NoiseData: 8
    NonlinearData: 37
         IntpType: 'Linear'
      NetworkData: [1×1 rfdata.network]
            nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
             Name: 'Amplifier'
mi1 =    rfckt.mixer with properties:
     MixerSpurData: []
         MixerType: 'Downconverter'
               FLO: 2.0300e+09
        FreqOffset: []
   PhaseNoiseLevel: []
         NoiseData: 4
    NonlinearData: 50
         IntpType: 'Linear'
      NetworkData: [1×1 rfdata.network]
            nPort: 2
   AnalyzedResult: [1×1 rfdata.data]
             Name: 'Mixer'
tx1 =    rfckt.microstrip with properties:
            Width: 6.0000e-04
           Height: 6.3500e-04
        Thickness: 7.5000e-09
         EpsilonR: 9.8000
      LossTangent: 0
        SigmaCond: Inf
       LineLength: 0.0100
         StubMode: 'NotAStub'
      Termination: 'NotApplicable'
            nPort: 2
   AnalyzedResult: []
             Name: 'Microstrip Transmission Line'
```

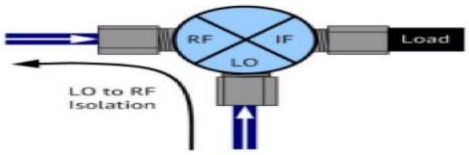
```
c =    rfckt.cascade with properties:
            Ckts: {1×6 cell}
           nPort: 2
    AnalyzedResult: []
            Name: 'Cascaded Network'
ans =    rfckt.cascade with properties:
            Ckts: {1×6 cell}
           nPort: 2
    AnalyzedResult: [1×1 rfdata.data]
            Name: 'Cascaded Network'
```

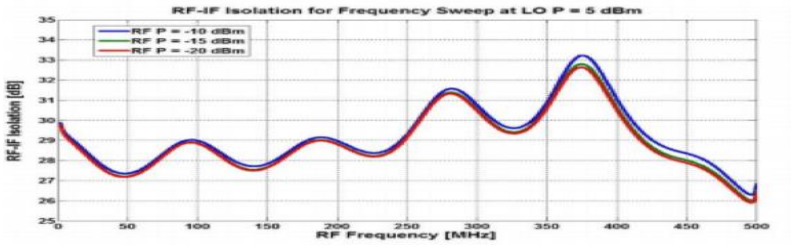
Model Waveforms:

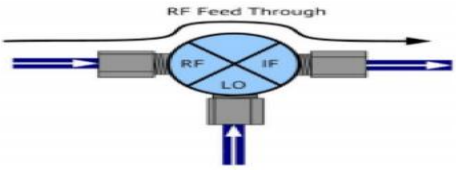


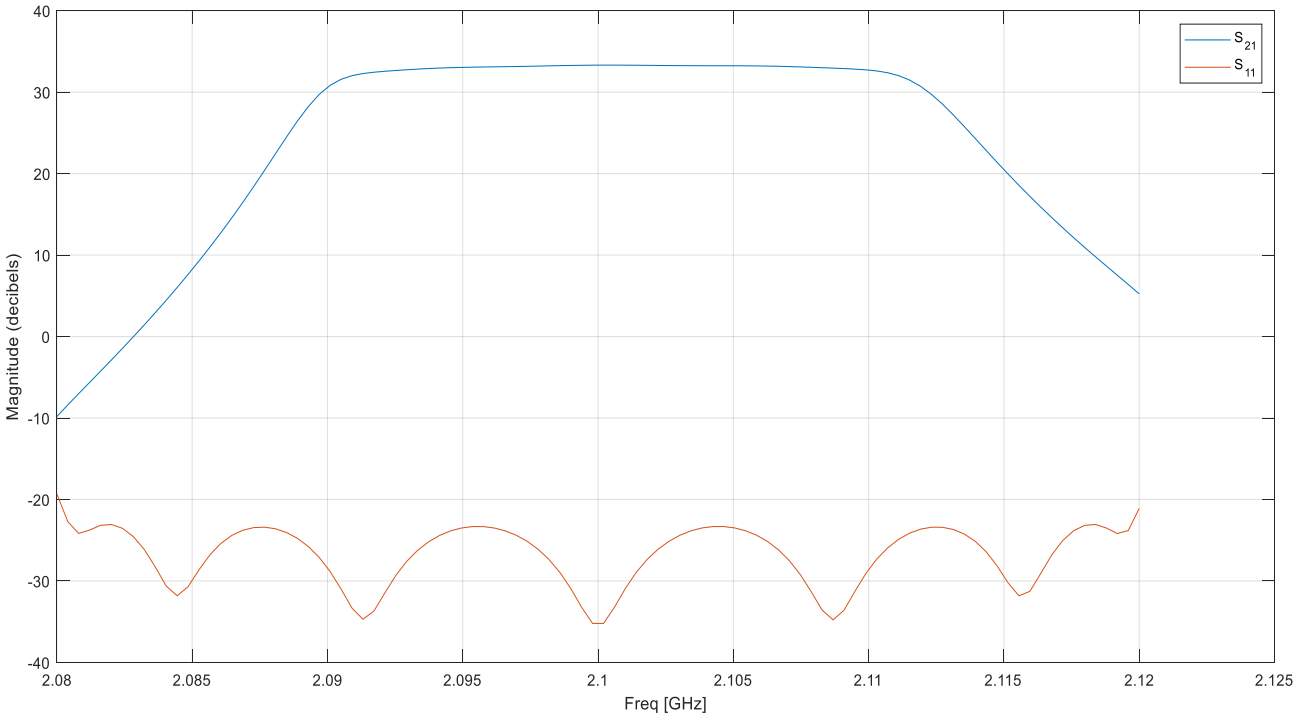












Result:

Viva questions:

1. What is mixer?

2. What is the need for mixer in communications?

3. Write ports in RF mixer?

4. Define isolation in RF mixer?

5. Define image rejection?

Exp: 10 **Date:**

SCHOTTKY DIODE AND RF SWITCH

Aim: Design and simulate a Schottky Diode and RF Switch.

Apparatus Required:

1. **Computer**
2. **Matlab(19b) software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

program:

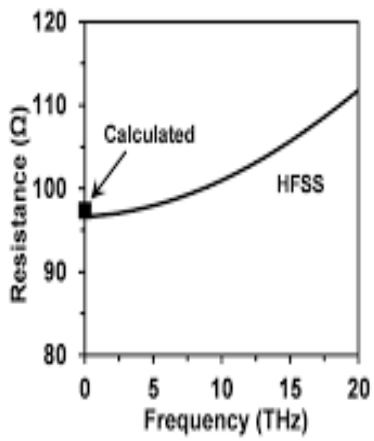
```
%%% Create the variables and equations
clc;
clear all;
close all;
clear Degrees Time R Vin Vr Vd Id;
Vd = 0.3;
R = 2;
for (j=0:1:9)
for (k=1:1:360)
Vin(k + (j* 360)) = 1.0 * sind(k - 1);
Time(k + (j* 360)) = (k + (j * 360)) / 360;
end
end
for (i=1:1:length(Vin))
if (Vin(i) > 0.3)
Vr(i) = (Vin(i) - Vd);
else
Vr(i) = 0;
end
end
Id = (Vr / R);
%%% Create the subplots
figure(1)
```

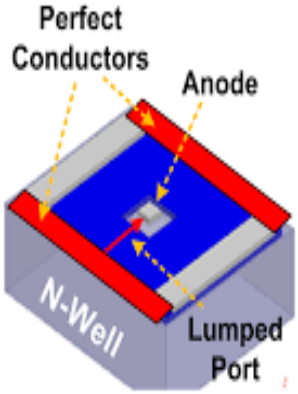
```
 subplot(3,1,1);
plot(Time, Vin);
grid;
title ('Time Vs Vin');
xlabel('Time (seconds)');
ylabel('Vin (V)');
 subplot(3,1,2);
plot(Time, Vr, 'r');
grid;
title ('Time Vs Vr');
xlabel('Time (seconds)');
ylabel('Vr (V)');
subplot(3,1,3);
plot(Time, Id, 'm');
grid;
title ('Time Vs Id');
xlabel('Time (seconds)');
ylabel('Id (A)');
%% %Test conditions
Rm = 997;
Vin = [0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6 6.5 7 7.5 8 8.5 9 9.5 10];
%%% Diode in forward bias
% Vr_forward =Vr;
Vr_forward = [0 0.268 0.739 1.223 1.712 2.204 2.697 3.190 3.685 4.180 ...
4.675 5.170 5.666 6.162 6.659 7.155 7.651 8.148 8.645 9.141 9.637];
Vd_forward = (Vin - Vr_forward);
Id_forward = (Vr_forward / Rm);
%%% Diode in reverse bias
% Vr_reverse =zeros(1,length(Vr));
Vr_reverse = [0 0 0 0 0 0 0 0 0 0 0 0 ...
0 0 0 0 0 0 0 0 0];
Vd_reverse = -(Vin - Vr_reverse);
Id_reverse = -(Vr_reverse / Rm);
%%% Plot results
figure(2)
subplot(2, 1, 1);
plot(Vd_forward, Id_forward, 'k-o');
grid
title('Vd Vs Id for measured Schottky Barrier Diode (forward bias)');
xlabel('Vd (V)');
ylabel('Id (A)');
subplot(2, 1, 2);
plot(Vd_forward, Id_forward, 'k-o');
hold on
grid on
plot(Vd_reverse, Id_reverse, 'r-o');
title('Vd Vs Id for measured Schottky Barrier Diode');
xlabel('Vd (V)');
ylabel('Id (A)');
```

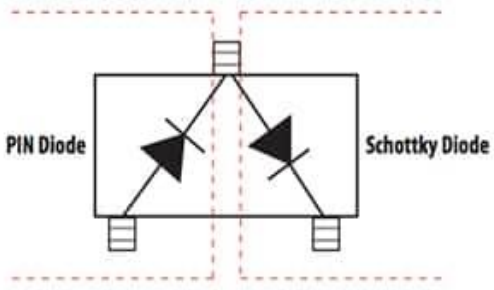
Calculation:

Vd_forward = Columns 1 through 12
0 0.2320 0.2610 0.2770 0.2880 0.2960 0.3030 0.3100 0.3150 0.3200 0.3250 0.3300
Columns 13 through 21
0.3340 0.3380 0.3410 0.3450 0.3490 0.3520 0.3550 0.3590 0.3630
Id_forward = Columns 1 through 12
0 0.0003 0.0007 0.0012 0.0017 0.0022 0.0027 0.0032 0.0037 0.0042 0.0047 0.0052
Columns 13 through 21
0.0057 0.0062 0.0067 0.0072 0.0077 0.0082 0.0087 0.0092 0.0097
Vd_reverse = Columns 1 through 12
0 -0.5000 -1.0000 -1.5000 -2.0000 -2.5000 -3.0000 -3.5000 -4.0000 -4.5000 -5.0000 -5.5000
Columns 13 through 21
-6.0000 -6.5000 -7.0000 -7.5000 -8.0000 -8.5000 -9.0000 -9.5000 -10.0000
Id_reverse = Columns 1 through 20
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Column 21

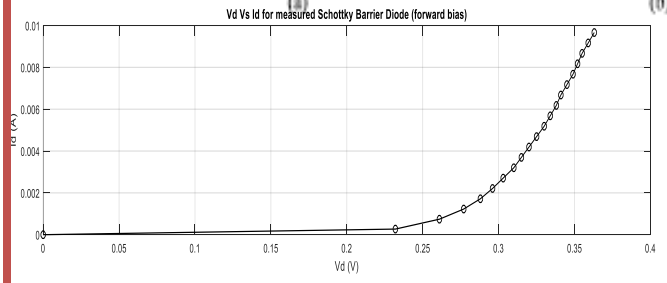
Model Waveforms:

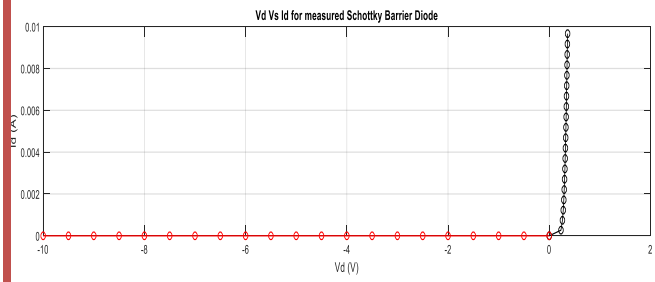




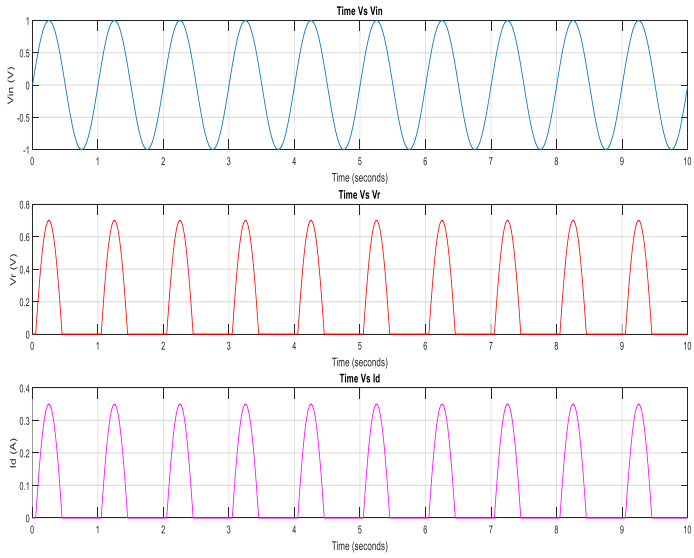


(a)





(b)



Result:

Viva questions:

1. What is schottkey diode?

2. Write applications of schottky diode?

3. Define RF switch?

4. Give the differences of switch and relay?

5. Give the range of radio frequency?

Exp: 11 **Date:**

POWER AMPLIFIER

Aim: Analyse and measure the gain of a Power Amplifier and equalize its gain using an Equalizer.

Apparatus Required:

1. **Computer**
2. **Matlab(19b) software.**

Theory:

Procedure:

1. Open MATLAB
2. Open new M-file
3. Type the program
4. Save in current directory
5. Compile and Run the program
6. For the output see command window\ Figure window

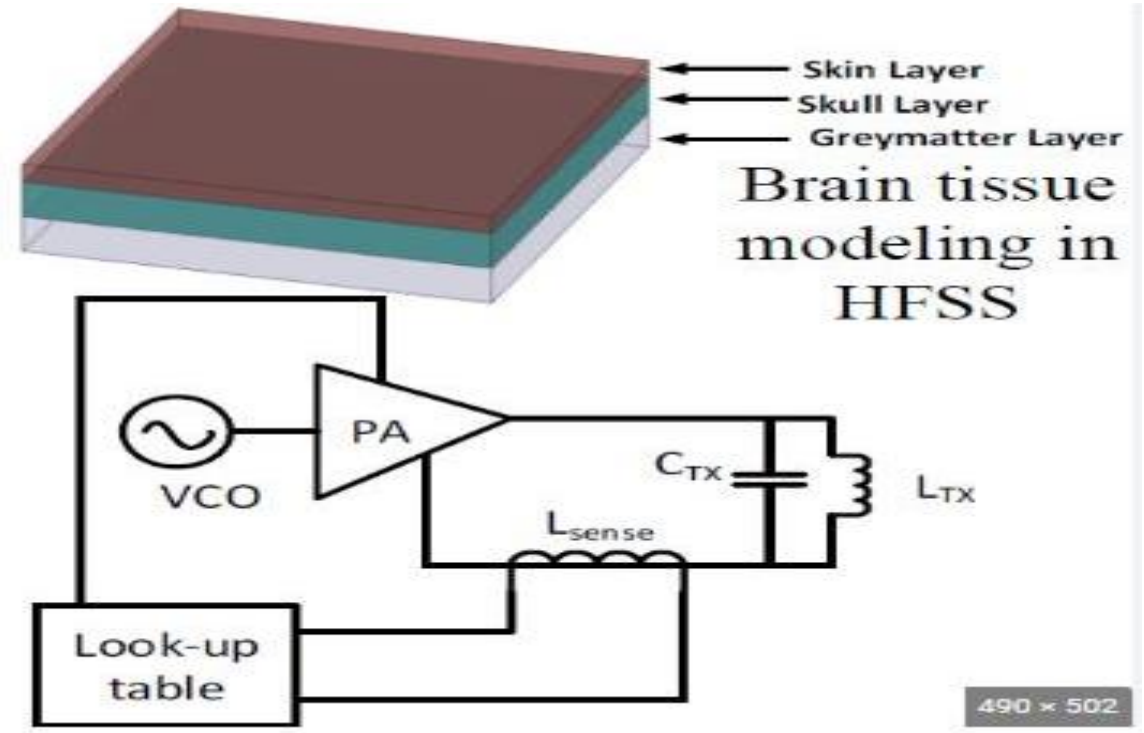
Applications:

Calculation:

Program:

```
clc;
clear all;
close all;
t=0:0.01:10;
%%%input
Yin = sin(2*pi*2*t)+cos(2*pi*3*t);
%%%amplifier output with noise
A=5;
Yout = A*(sin(2*pi*2*t)+cos(2*pi*3*t));
figure(1)
plot(t,Yin,t,Yout)
%%%amplifier output with noise
A=5;
Yout_noise = A*(sin(2*pi*2*t)+cos(2*pi*3*t))+rand(1,length(Yin));
figure(2)
plot(t,Yin,t,Yout_noise)
%%%gain calculation
gain = abs(Yout/Yin );
gain_noise = abs(Yout_noise/Yin );
   figure;
   TransferPA = abs(Yout./Yin );
   TransferPA_noise = abs(Yout_noise./Yin );
   plot(abs(Yin),20*log10(TransferPA),'r-o',abs(Yin),20*log10(TransferPA_noise),'b.');
%     plot(20*log10(abs(Yin)),20*log10(TransferPA),'r-
.',20*log10(abs(Yin)),20*log10(TransferPA_noise),'b.');
   xlabel('Input Voltage Absolute Value(V)')
   ylabel('Magnitude Power Gain (dB)');
```

Model Waveforms:



Result:

Conclusion:

Viva questions:

1. What is power amplifier?

2. Define low loss RF equalizer?

3. What is gain equalizer?

4. Define lattice and bridged T-equalizers?

5. Write applications of power amplifiers?

Exp:12 **Date:**

VCO

Aim: Design and characterization of VCO.

Apparatus Required:

1.Computer

2.Matlab(19b) software.

Theory:

Procedure:

1.Open MATLAB

2.Open new M-file

3.Type the program

4.Save in current directory

5.Compile and Run the program

6.For the output see command window\ Figure window

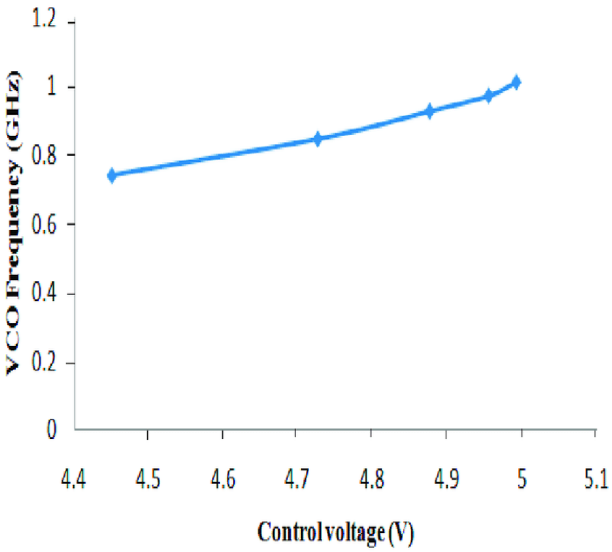
Applications:

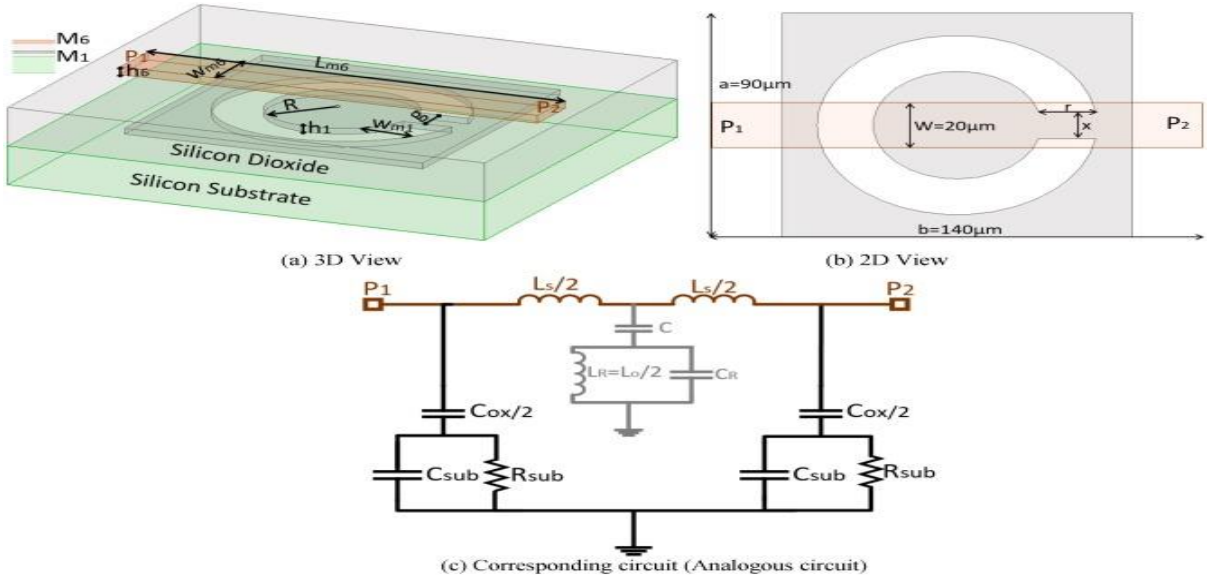
Calculation:

Program:

```
clc;
clear all;
close all;
  Fs = 10000;
 t = 0:1/Fs:2;
   x = sin(2*pi*t);
  range=[0.1 0.4]*Fs;
  Fc = mean(range);
   x_max = max(max(abs(x)));
     kf = (Fc/Fs)*2*pi/x_max;
        y = cos(2*pi*Fc*t + kf*cumsum(x));
      plot(t,y);
   figure(2)
 spectrogram(y,kaiser(256,5),220,512,Fs,'yaxis');
 %%%%Using command VCO
%   fs = 10000;
%   t = 0:1/fs:2;
%    x = sin(2*pi*t);
%   y= vco(x,[0.1 0.4]*fs,fs);
%   plot(y);
%   figure(2)
%   spectrogram(y,kaiser(256,5),220,512,fs,'yaxis')
```

Model Waveforms:





(a) 3D View (b) 2D View

(c) Corresponding circuit (Analogous circuit)

Result:

Conclusion:

Viva questions:

1. What is VCO?

2. Give the differences VCO and oscillator?

3. What is PLL?

4. Write the working principle of VCO?

5. Applications of VCO?

ADVANCED EXPERIMENTS

Exp:01 **Date:**

Patch antenna using MATLAB

Aim: Design and characterization of Micro strip patch antenna.

Apparatus Required:

1.Computer

2.Matlab(19b) software.

Theory:

Procedure:

1.Open MATLAB

2.Open new M-file

3.Type the program

4.Save in current directory

5.Compile and Run the program

6.For the output see command window\ Figure window

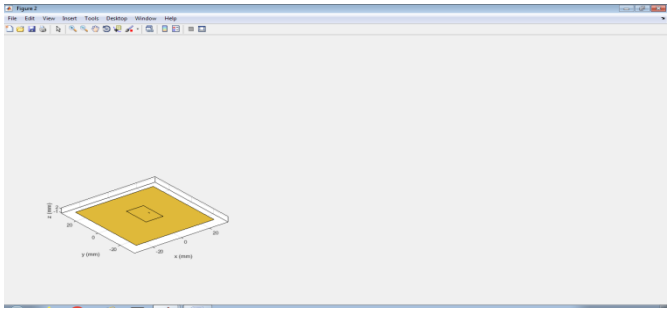
Applications:

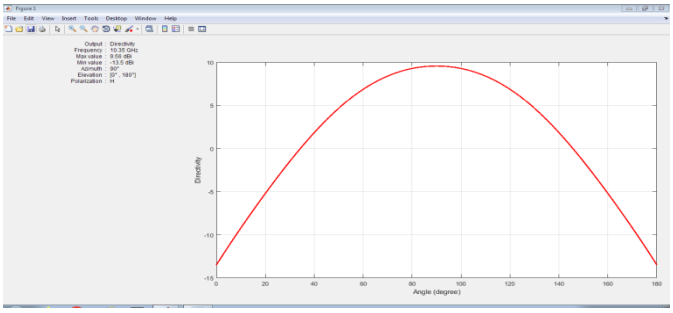
Calculation:

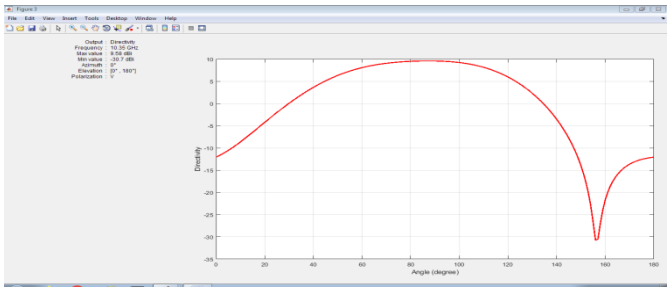
Program:

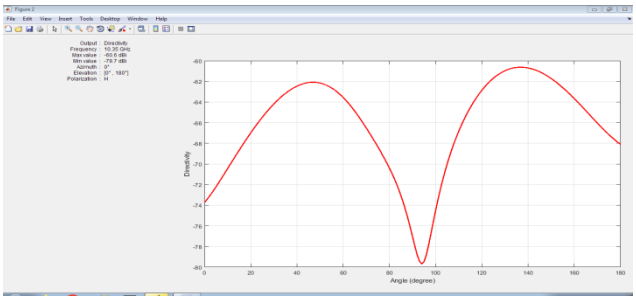
```
clc;
clear all;
close all;
freq        = 10.35e9;
patchLength = 12e-3;
patchWidth  = 17.73e-3;
patchHeight = 1.56e-3;
lengthgp    = 55e-3;
widthgp     = 55e-3;
feedoffset  = [2.9e-3 0];
ant = patchMicrostrip('Length', patchLength, 'Width', patchWidth,       ...
    'Height', patchHeight, 'GroundPlaneLength', lengthgp,               ...
    'GroundPlaneWidth', widthgp, 'FeedOffset', feedoffset);
show(ant);
figure(1);
pattern(ant, freq, 90, 0:1:180, 'CoordinateSystem', 'rectangular',      ...
    'Polarization', 'H');
figure(2);
pattern(ant, freq, 90, 0:1:180, 'CoordinateSystem', 'rectangular',      ...
    'Polarization', 'V');
figure(3);
pattern(ant, freq, 0, 0:1:180, 'CoordinateSystem', 'rectangular',       ...
    'Polarization', 'V');
figure(4);
pattern(ant, freq, 0, 0:1:180, 'CoordinateSystem', 'rectangular',       ...
    'Polarization', 'H');
```

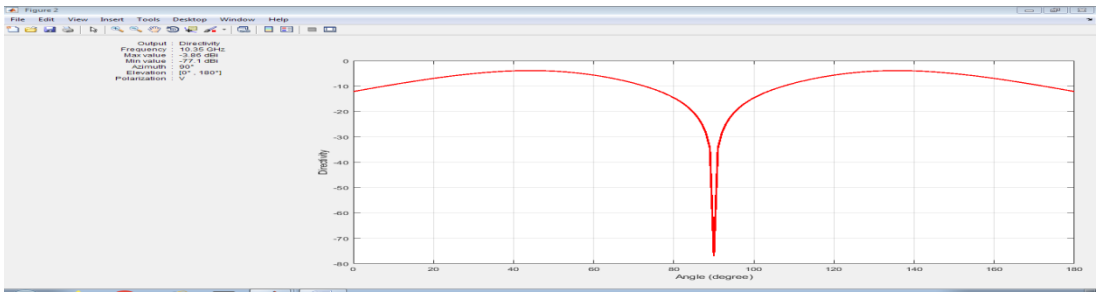
Model Graph











Result:

Conclusion:

Viva questions:

Exp:02 **Date:**

Folded dipole (yagi-uda) antenna using MATLAB

Aim: Design and characterization of folded dipole (yagi-uda) antenna**.**

Apparatus Required:

1.Computer

2.Matlab(19b) software.

Theory:

Procedure:

1.Open MATLAB

2.Open new M-file

3.Type the program

4.Save in current directory

5.Compile and Run the program

6.For the output see command window\ Figure window

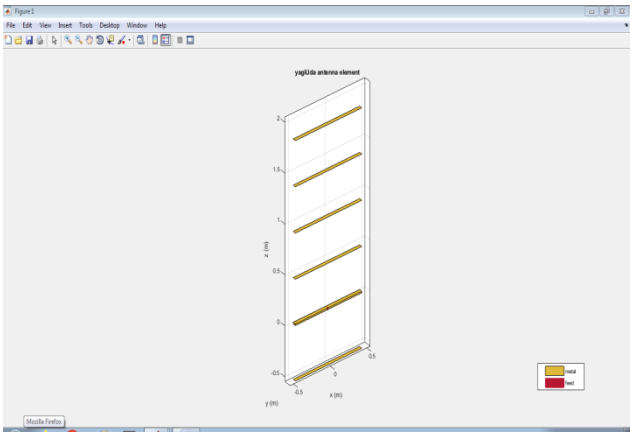
Applications:

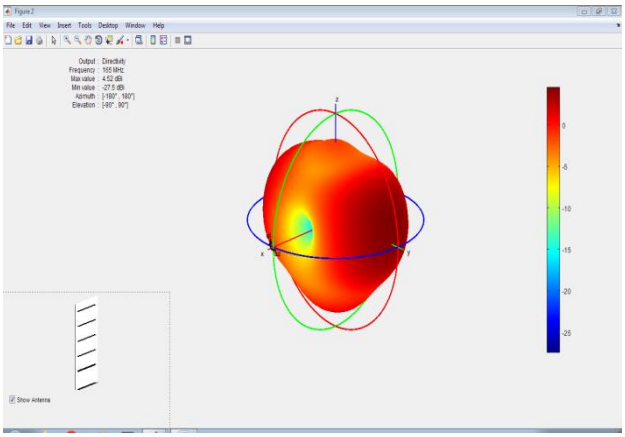
Calculation:

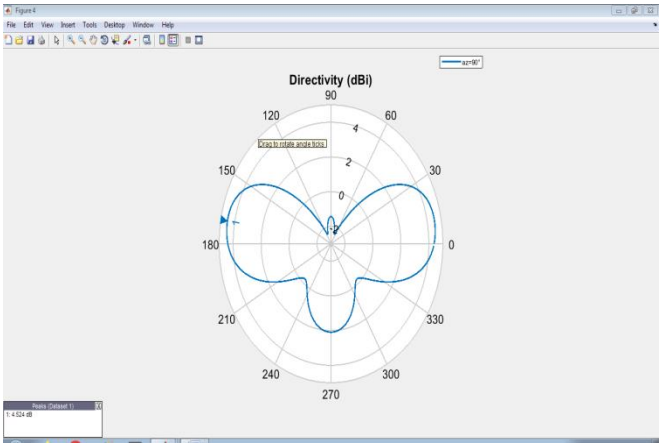
Program:

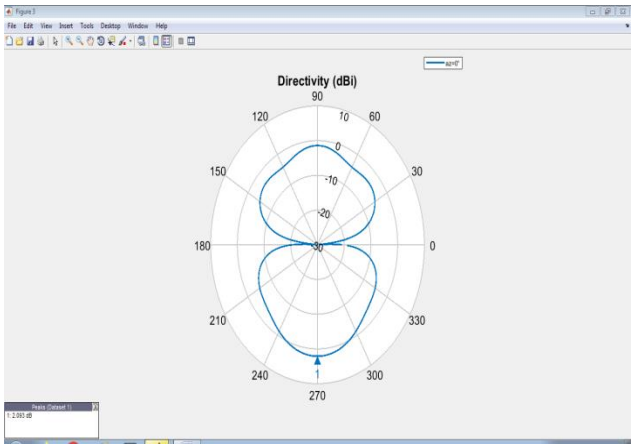
```
clc;
clear all;
close all;
freq = 165e6;
wirediameter = 19e-3;
c = physconst('lightspeed');
lambda = c/freq;
d = dipoleFolded;
d.Length = lambda/2;
d.Width = cylinder2strip(wirediameter/2);
d.Spacing = d.Length/60;
Numdirs = 4;
refLength = 0.5;
dirLength = 0.5*ones(1,Numdirs);
refSpacing = 0.3;
dirSpacing = 0.25*ones(1,Numdirs);
initialdesign = [dirLength refSpacing dirSpacing].*lambda;
yagidesign = yagiUda;
yagidesign.Exciter = d;
yagidesign.NumDirectors = Numdirs;
yagidesign.ReflectorLength = refLength*lambda;
yagidesign.DirectorLength = dirLength.*lambda;
yagidesign.ReflectorSpacing = refSpacing*lambda;
yagidesign.DirectorSpacing = dirSpacing*lambda;
show(yagidesign)
fig1 = figure;
pattern(yagidesign,freq);
fig3 = figure;
pattern(yagidesign,freq,0,0:1:359);
fig4 = figure;
pattern(yagidesign,freq,90,0:1:359);
```

Model Graph









Result:

Conclusion:

Viva questions: