

## Lab Manual

*Simulation & Design Tools (2141006)*

BACHELOR OF ENGINEERING  
in  
ELECTRONICS AND COMMUNICATION ENGINEERING

By

Chaitanya Tejaswi (140080111013)

Under The Guidance of  
Prof. Anish Vahora  
Asst. Professor, EC Department.



ELECTRONICS & COMMUNICATION ENGINEERING  
DEPARTMENT  
BVM ENGINEERING COLLEGE  
GUJARAT TECHNOLOGICAL UNIVERSITY  
VALLABH VIDYANAGAR-388120  
Academic Year- 2015-16

## INDEX

No.	PRACTICAL	PAGES	DATE	SIGN
1	Introduction to OrCAD Capture 16.3			
2	To simulate and find Voltage and Current in Resistive network using OrCAD Capture 16.3			
3	To Simulate Diode Clipper Circuit using OrCAD Capture 16.3			
4	To Simulate Diode Clamper Circuit using OrCAD Capture 16.3			
5	To Simulate Diode as Half-wave Rectifier Circuit using OrCAD Capture 16.3			
6	To Simulate Diode as full-wave Rectifier Circuit using OrCAD Capture 16.3			
7	To Simulate Diode as Bridge-wave Rectifier Circuit using OrCAD Capture 16.3			
8	To Simulate Diode characteristics Circuit using OrCAD Capture 16.3			
9	To Simulate transistor characteristics Circuit using OrCAD Capture 16.3			
10	To Simulate transistor as amplifier Circuit using OrCAD Capture 16.3			
11	To Simulate OPAMP Circuit using OrCAD Capture 16.3			
12	To Simulate IC 555 Timer Circuit using OrCAD Capture 16.3			
13	To Simulate All basic Digital gate Circuit using OrCAD Capture 16.3			
14	To Simulate NAND Digital gate as Universal gate Circuit using OrCAD Capture 16.3			
15	To Simulate NOR Digital gate as Universal gate Circuit using OrCAD Capture 16.3			
16	To Simulate Half Adder Circuit using OrCAD Capture 16.3			
17	To Simulate Full Adder Circuit using OrCAD Capture 16.3			
18	To implement and Simulate given Boolean function Circuit using OrCAD Capture 16.3			
19	To implement and Simulate 3 –bit Counter Circuit using OrCAD Capture 16.3			
20	To implement and Simulate modulo-N Counter Circuit using OrCAD Capture 16.3			
21	To study PCB layout design using OrCAD16.3			
22	To study MATLAB basic command.			

# **PRACTICAL: 1**

**AIM:** Introduction to OrCAD.

**SOFTWARE:** Orcad Capture CIS 16.3

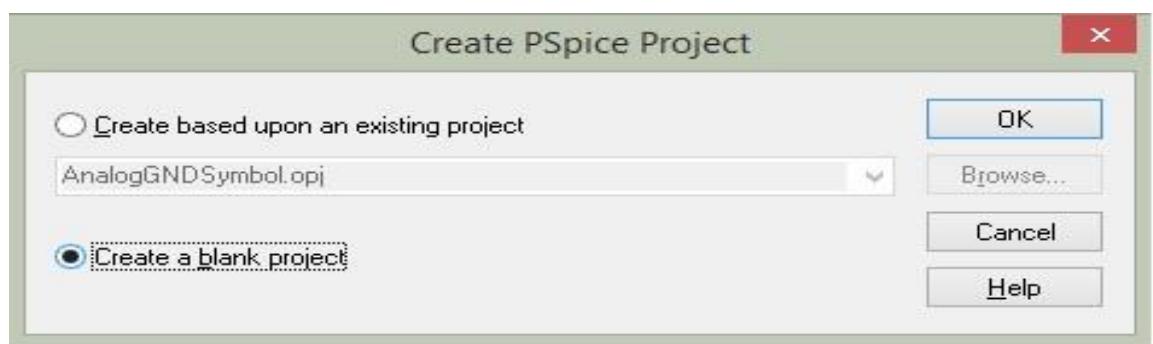
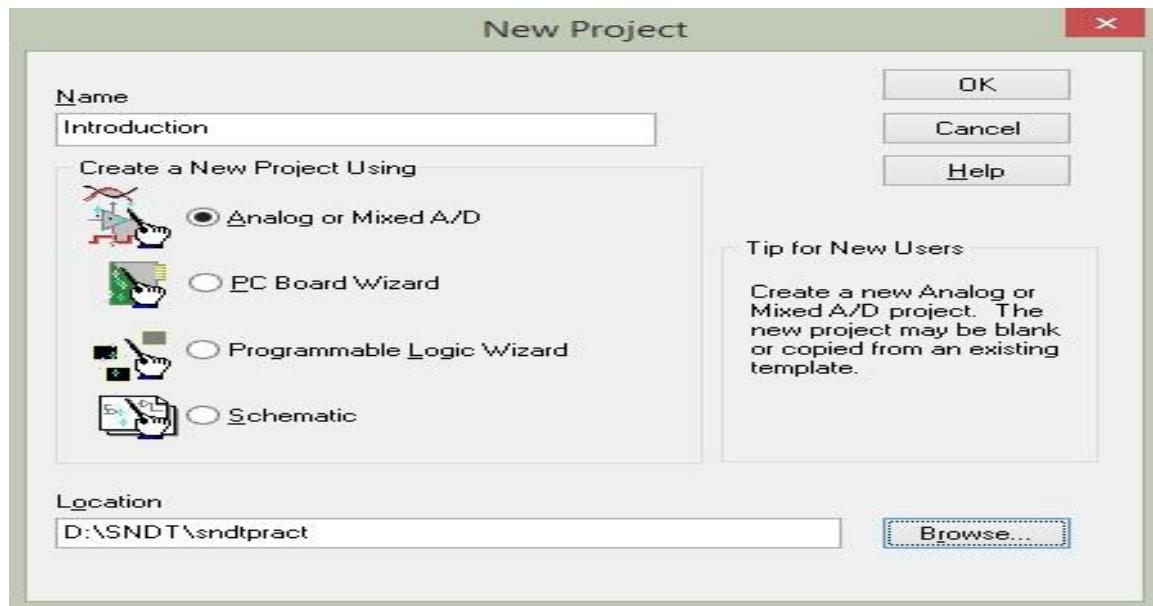
## **PROCEDURE:**

- For any type of simulation design, we got to have the above named software that is, Cadence-OrCAD capture 16.3 installed in your PC where we do the required simulation using PSpice.
- There are four basic steps involved in circuit simulation using PSpice which is:
  - Creating a new project and Schematic Diagram.
  - Selecting circuit components, connecting them together and setting component values and properties and saving the schematic diagram.
  - Creating a new Simulation Profile and setting up simulation.
  - Simulating the circuit and observing the simulation results.
- Creating a new project and Schematic Diagram.
- Open the required software by using the following steps below:

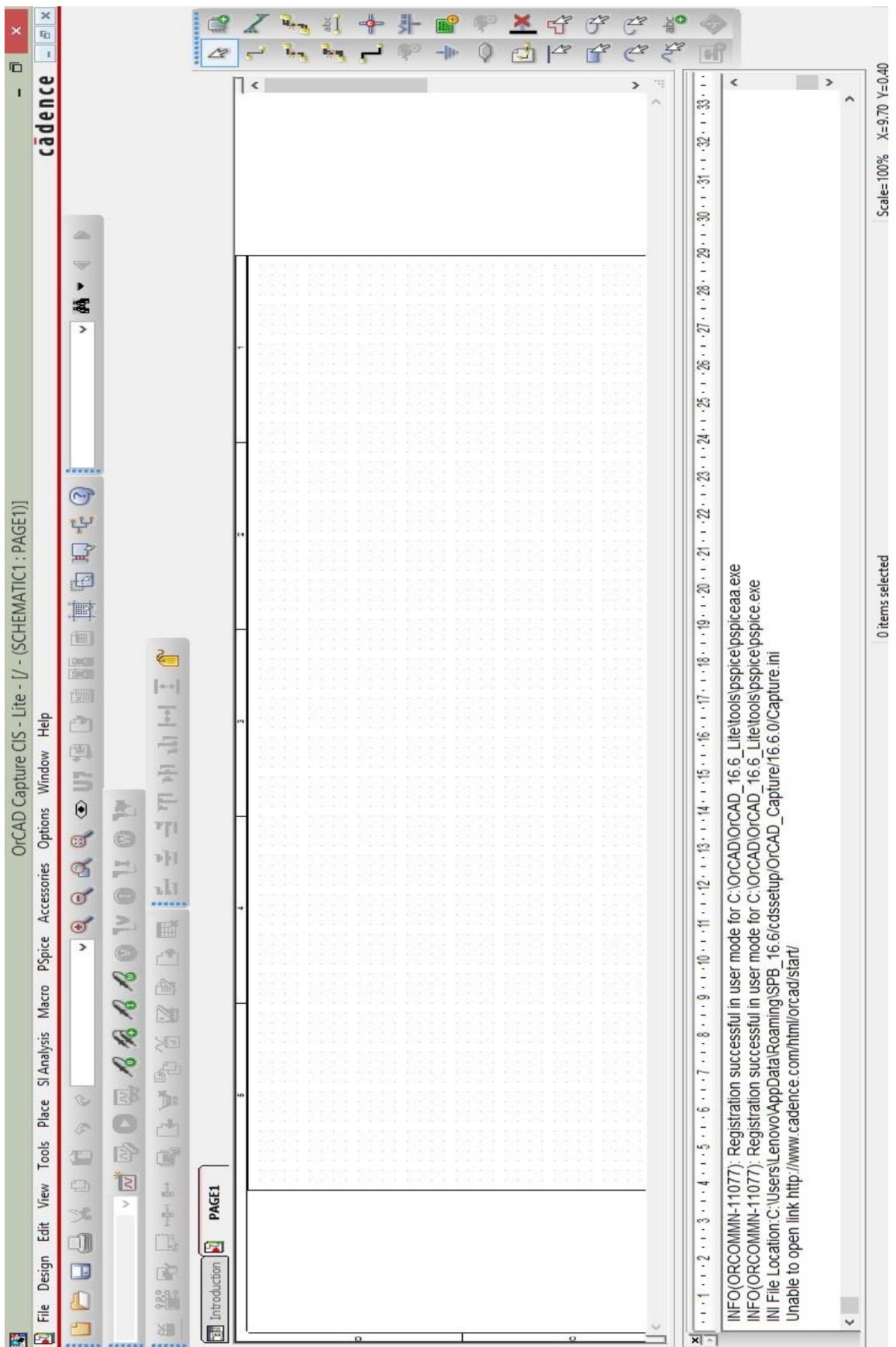
Start → All Programs → Cadence → Release 16.3 → OrCAD Capture

Create a New Project following the steps below:

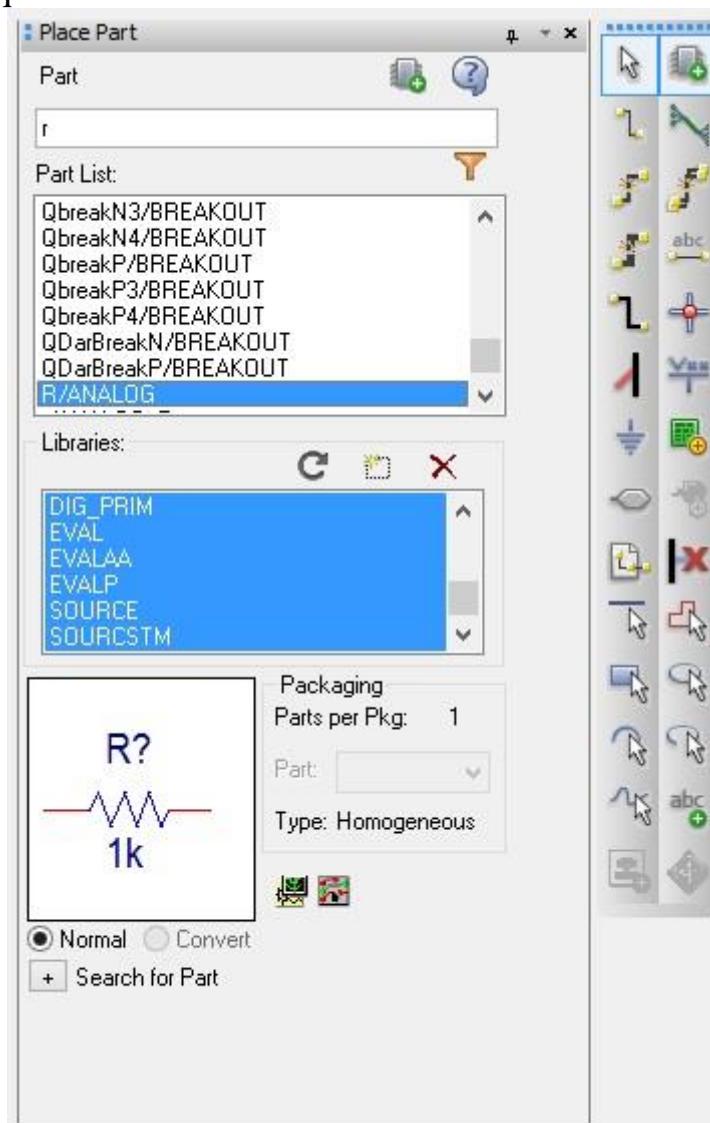
- a) File → New → Project
- b) Enter project name
- c) Select Analog or Mixed A/D.
- d) Create a folder for the project.
- e) Click OK.



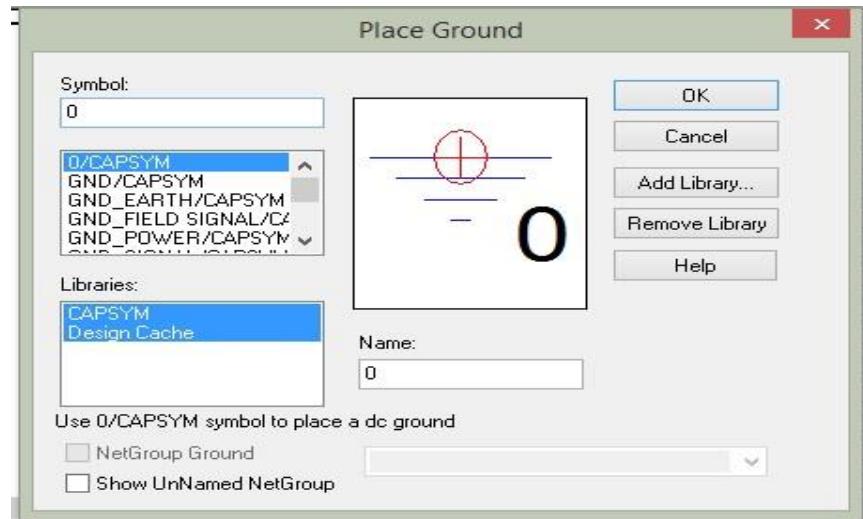
- Select 'Create a Blank Project' and Click OK. You should now see a blank schematic diagram as shown in figure.



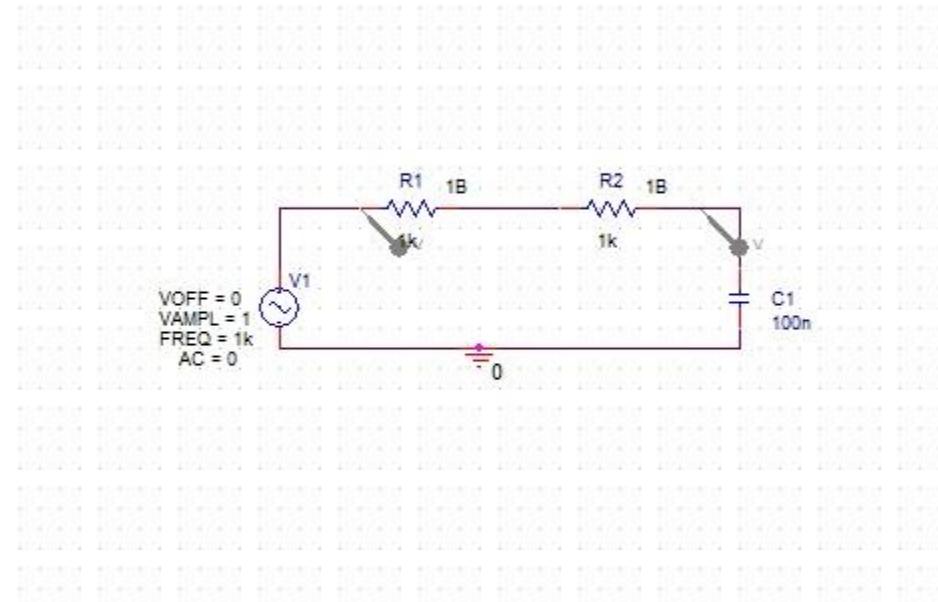
- Selecting circuit components, connecting them together and setting component values and properties and solving the schematic diagram:-
- You will be working essentially with two screens, the capture CIS screen and the PSpice AD screen. These have various toolbars and icons associated with them. We just got them to use them as instructed or as per the requirement.



- Place the required circuit components like resistor, capacitor, inductor, ground, diode, transistor, AC or DC voltage source etc. on the “Breadboard” on the screen. Join them using connecting wires.



- After placing the various required components in the breadboard change their values to the actually require one.
- At last our circuit should look like the following circuit.



- At the end of it all, we just need to put the required “markers” (like-voltage and current markers) near the circuit component under observation (so that we could get the current or the voltage output w.r.t. time as per the marker selected).
- Save it.

## PRACTICAL: 2

**AIM:** Calculate voltage & current of different node & in different loops for resistive network.

**SOFTWARE:** Orcad capture CIS 16.3

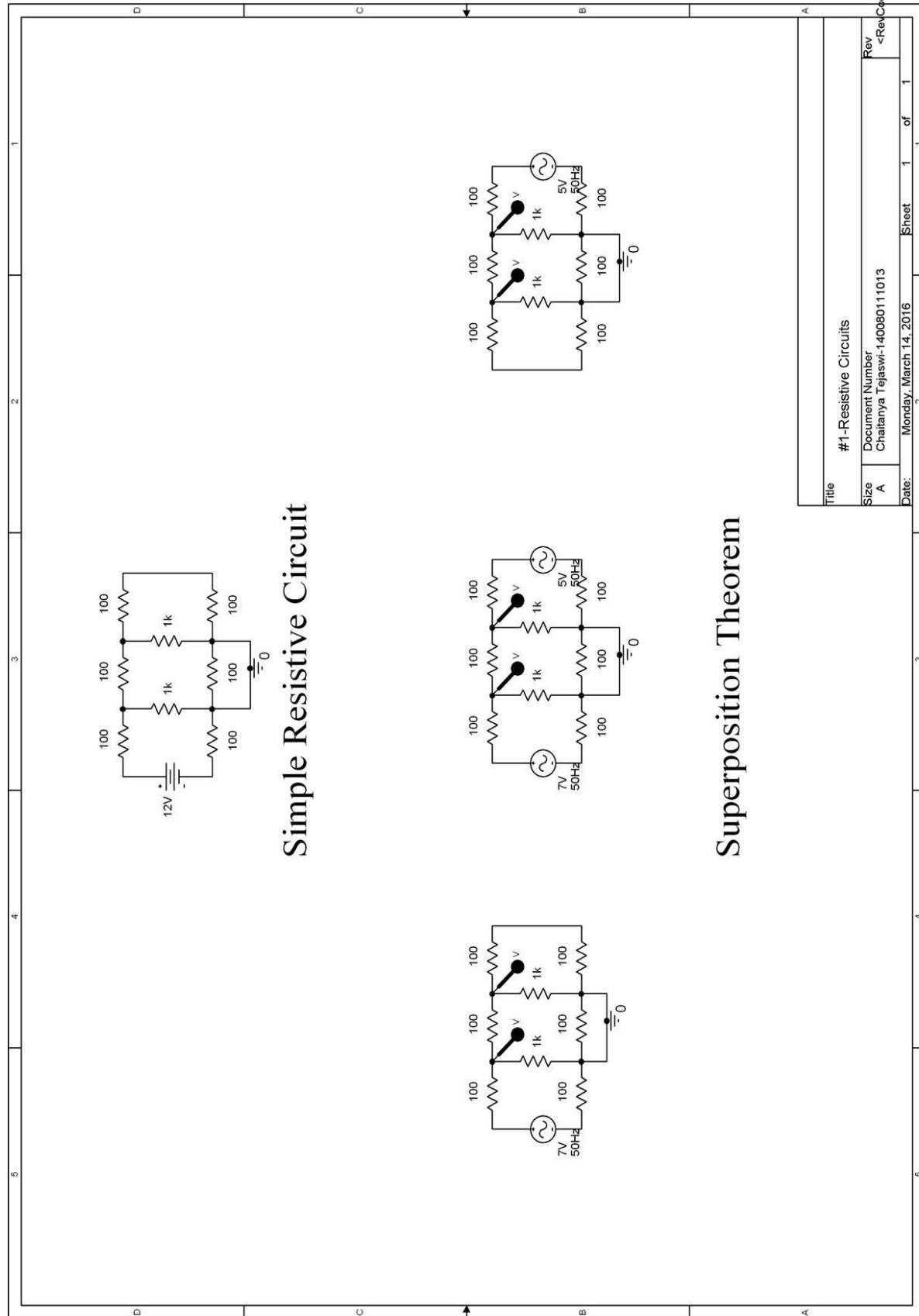
### THEORY:

A network, in the context of electronics, is a collection of interconnected components. Network analysis is the process of finding the voltages across, and the currents through, every component in the network. There are many different techniques for calculating these values. However, for the most part, the applied technique assumes that the components of the network are all linear.

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Now run the project using Pspice.
- Then save the practical.

## CIRCUIT :



## **NETLIST :**

\* source R1

V\_V1 N00144 0 12  
R\_R1 N00144 N00151 100  
R\_R2 N00151 N00155 100  
R\_R3 0 N00151 100  
R\_R4 0 N00155 100

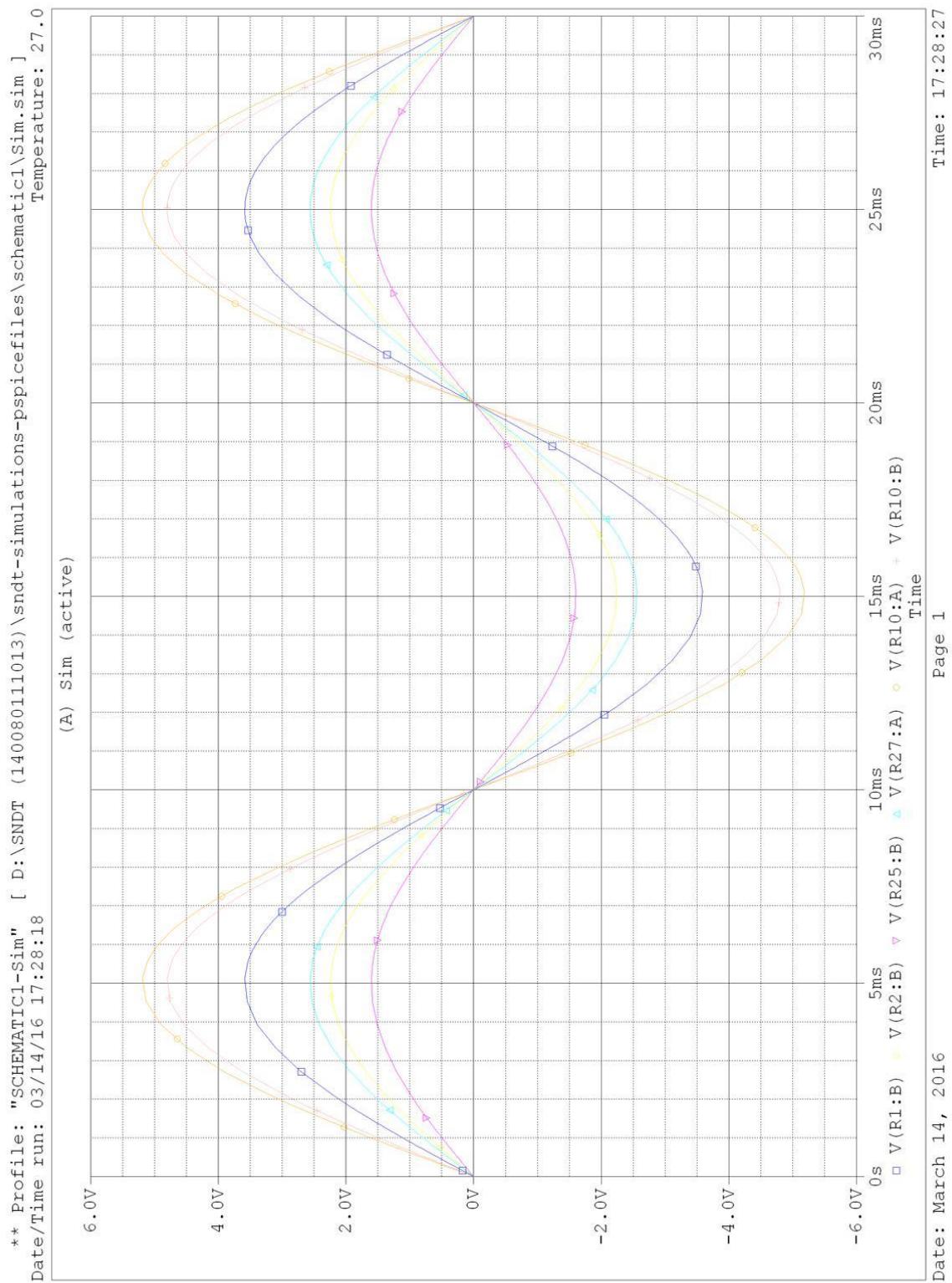
\* source R2

V\_V1 N00180 0 10  
V\_12 N00262 0 12  
R\_R1 N00180 N00187 2  
R\_R2 N00187 N00248 2  
R\_R3 0 N00187 2  
R\_R4 0 N00248 2  
R\_R5 N00248 N00262 2

\* source R3

R\_R1 N00180 N00187 1k  
R\_R2 N00187 N00248 1k  
R\_R3 0 N00187 1k  
R\_R4 0 N00248 1k  
R\_R5 N00248 0 1k  
V\_V2 N00180 0 AC 0  
+SIN 0 12 50 0 0 0

## OUTPUT :



## **CONCLUSION :**

By performing this practical, we concluded that any complicated or non-complicated circuit can be implemented and also stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

## PRACTICAL: 3

**AIM:** To study about the diode clipping circuits.

**SOFTWARE:** OrCAD capture CIS 16.3

### THEORY:

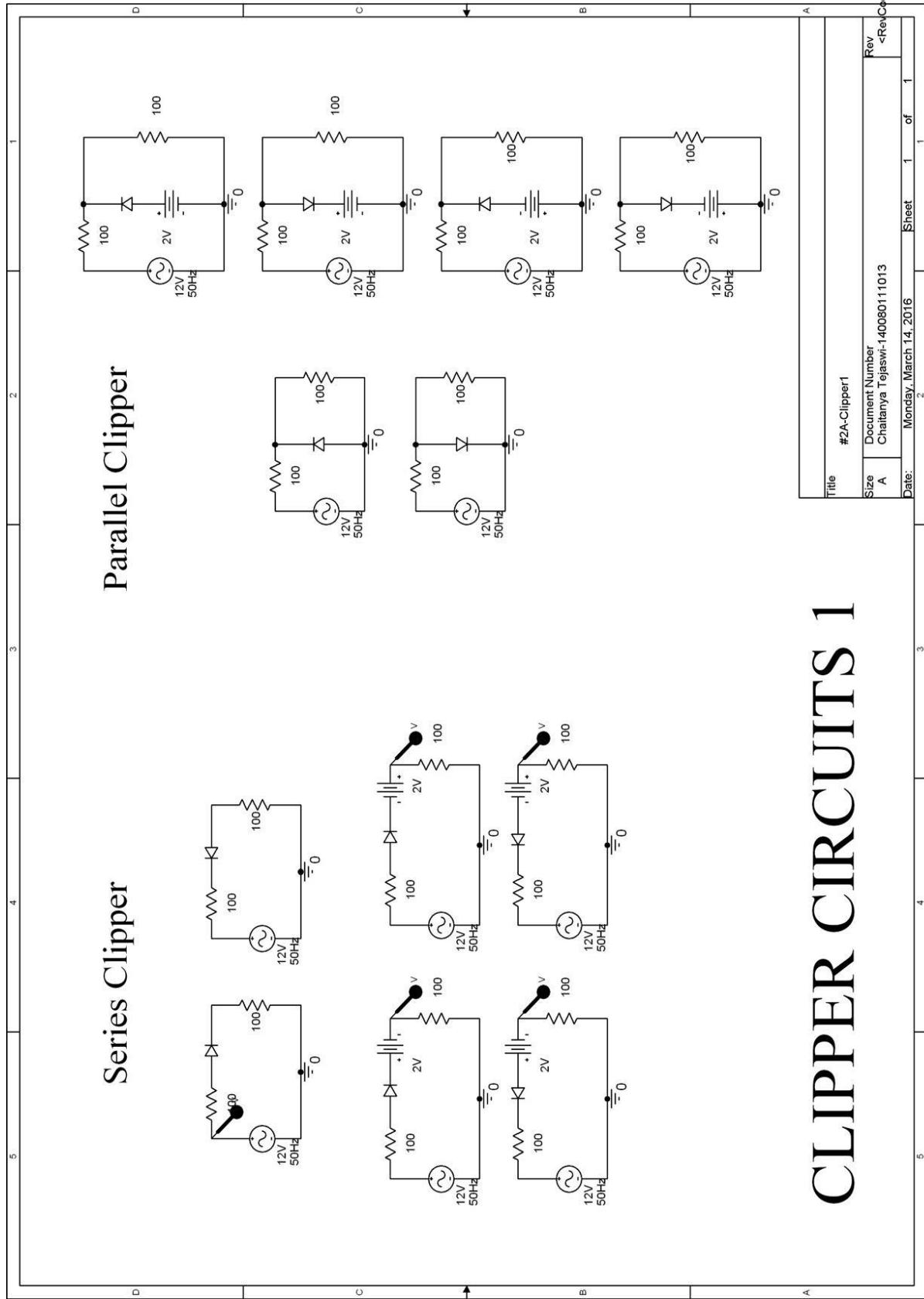
1. In electronics , a clipper is a device designed to prevent the output of a circuit from exceeding a predetermined voltage level without distorting the remaining part of the applied waveform.
2. A clipper circuits consists of linear elements like resistors and non linear elements like diode, but it does not contain energy storage device like capacitor.
3. Clipper circuits are used to select for purpose of transmission , that part of a signal wave form which lies above or below a certain reference voltage level.
4. Here we will study different clipper circuits and their outputs.

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.

- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Now run the project using Pspice and note the characteristic.
- Then save the practical.

## CIRCUIT :



## **NETLIST :**

\* source CLIP1

R\_R6 0 N16409 1k TC=0,0

V\_V3 N16399 0

+SIN 0 12 50 0 0 0

R\_R5 N16399 N16409 1k TC=0,0

D\_D2 0 N16409 D1N4002

\* source CLIP2

R\_R6 0 N16409 1k TC=0,0

V\_V3 N16399 0

+SIN 0 12 50 0 0 0

R\_R5 N16399 N16409 1k TC=0,0

D\_D2 N16409 0 D1N4002

\* source CLIP3

V\_V1 N16901 0

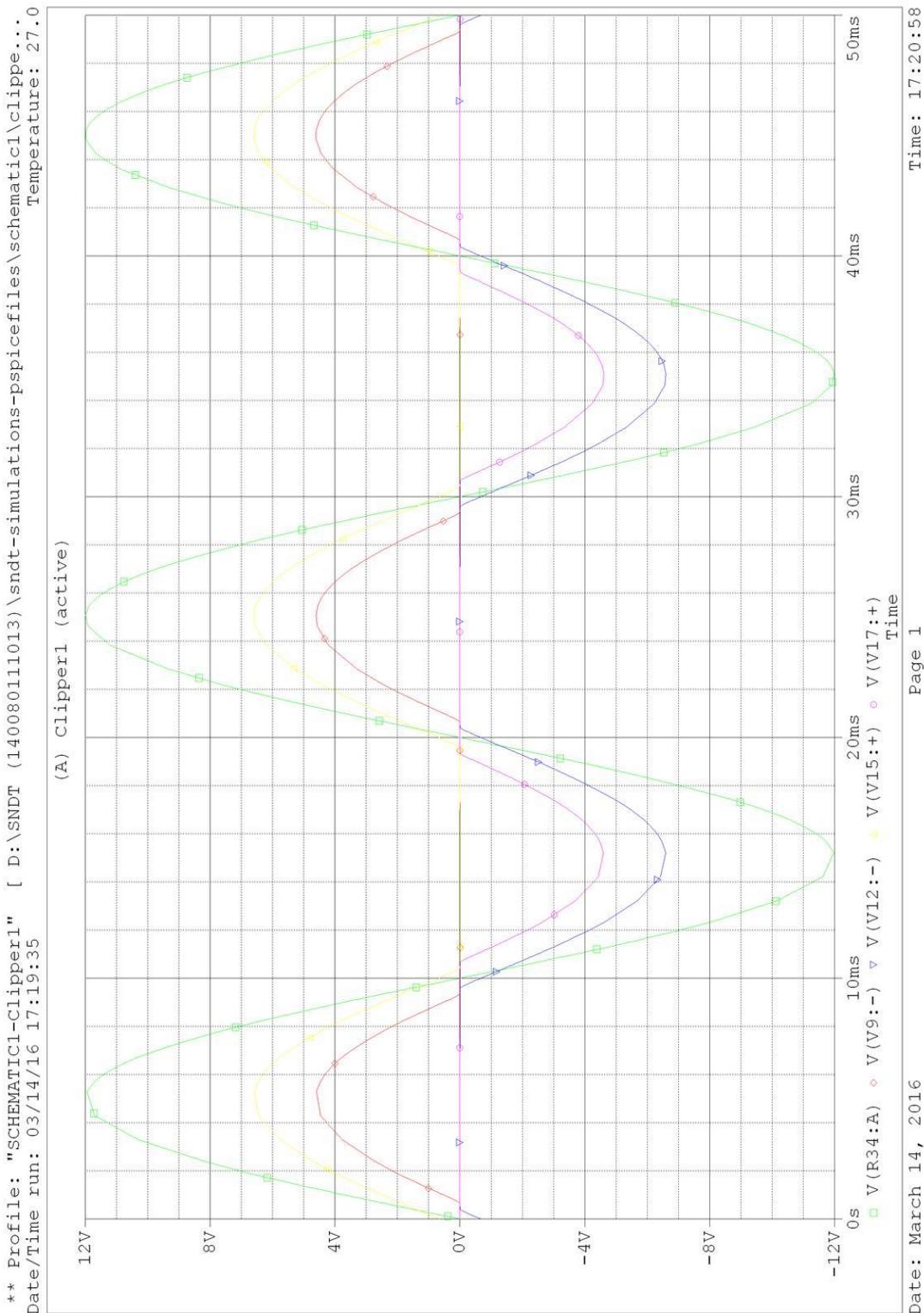
+SIN 0 12 50 0 0 0

R\_R3 0 N16943 1k TC=0,0

R\_R1 N16901 N16911 100 TC=0,0

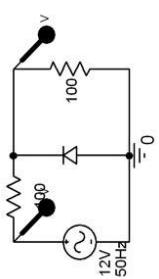
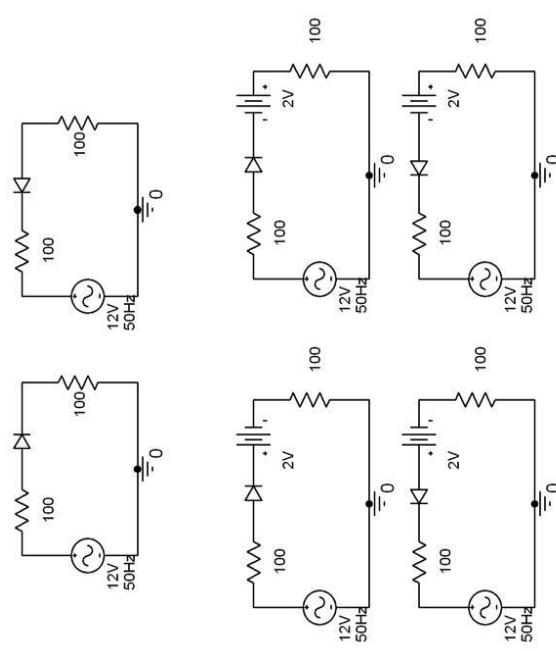
D\_D1 N16911 N16943 D1N4002

## **OUTPUT :**

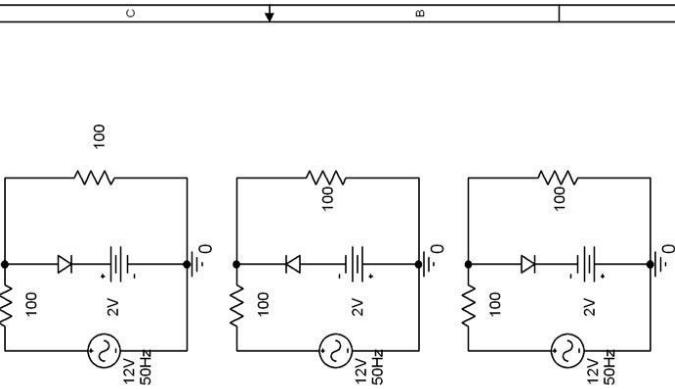
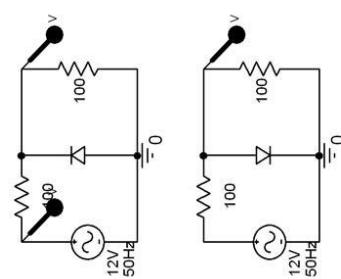


## CIRCUIT :

### Series Clipper



### Parallel Clipper



# CLIPPER CIRCUITS 1

Title		#2A-Clipper1	Rev
Size	A	Document Number	
Chaitanya Tejaswi		140080111013	<RevC>
Date:	Monday, March 14, 2016	Sheet 1 of 1	

## **NETLIST :**

\* source CLIP1

V\_V1 N16901 0

+SIN 0 12 50 0 0 0

R\_R3 0 N16943 1k TC=0,0

R\_R1 N16901 N16911 100 TC=0,0

D\_D1 N16943 N16911 D1N4002

\* source CLIP2

V\_V5 N28203 0

+SIN 0 12 50 0 0 0

R\_R9 N28203 N28303 1k TC=0,0

R\_R10 0 N28175 1k TC=0,0

V\_V6 N28307 N28175 1

D\_D1 N28303 N28307 D1N4002

\* source CLIP3

V\_V5 N28203 0

+SIN 0 12 50 0 0 0

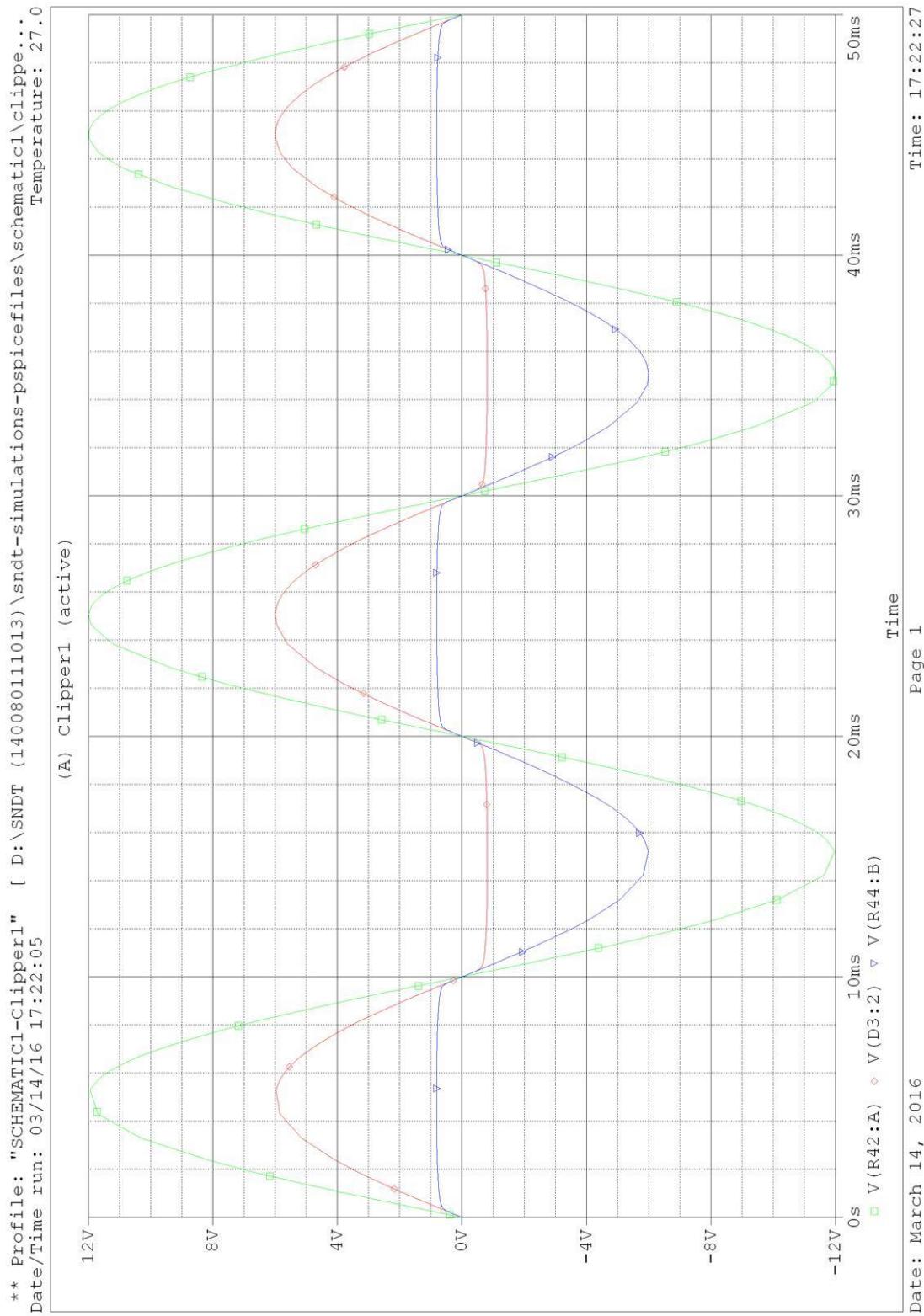
R\_R9 N28203 N28303 1k TC=0,0

R\_R10 0 N28175 1k TC=0,0

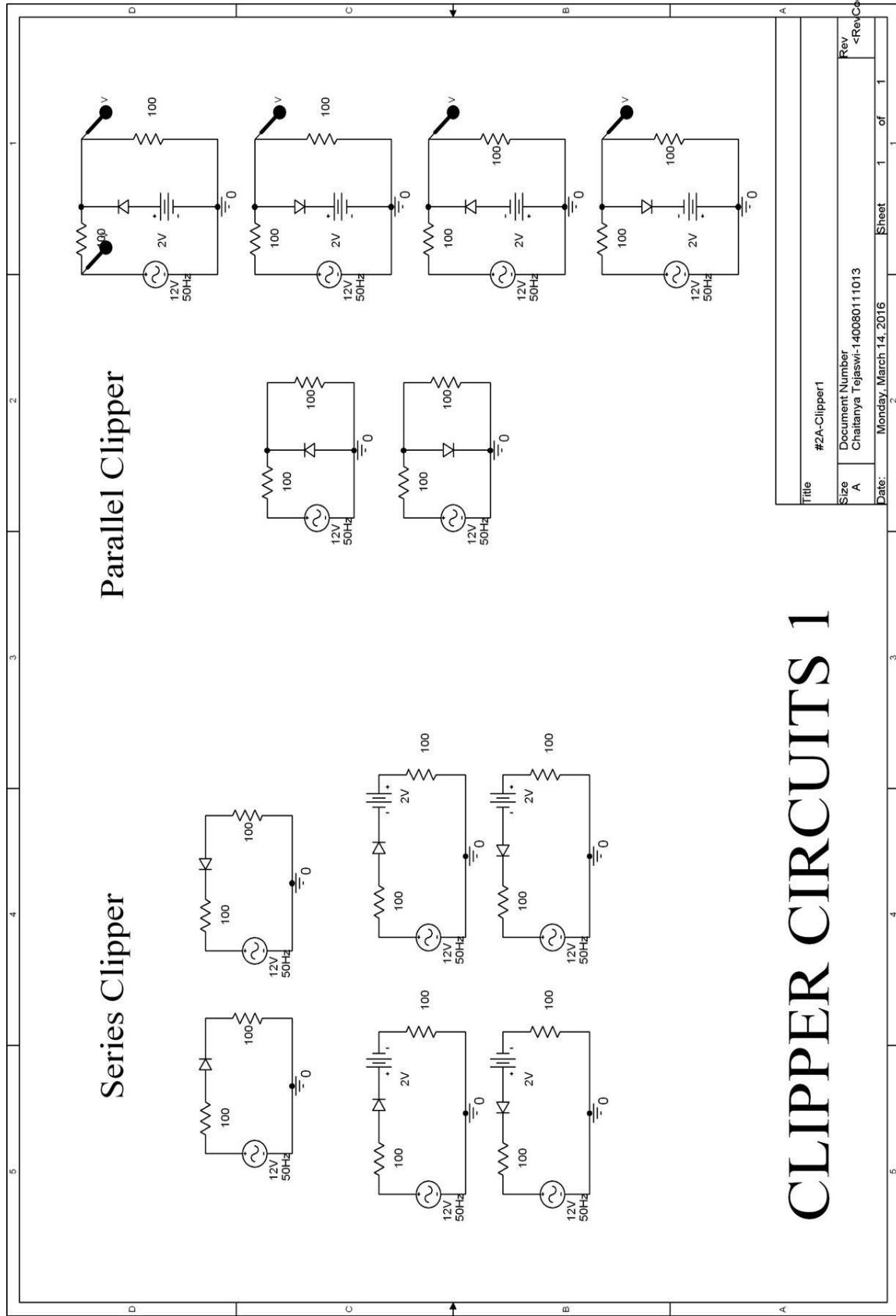
V\_V6 N28307 N28175 1

D\_D1 N28307 N28303 D1N4002

## OUTPUT :



## CIRCUIT :



## **NETLIST :**

\* source CLIP1

V\_V5 N28203 0  
+SIN 0 12 50 0 0 0  
R\_R9 N28203 N28303 1k TC=0,0  
R\_R10 0 N28175 1k TC=0,0  
V\_V6 N28175 N28307 1  
D\_D1 N28303 N28307 D1N4002

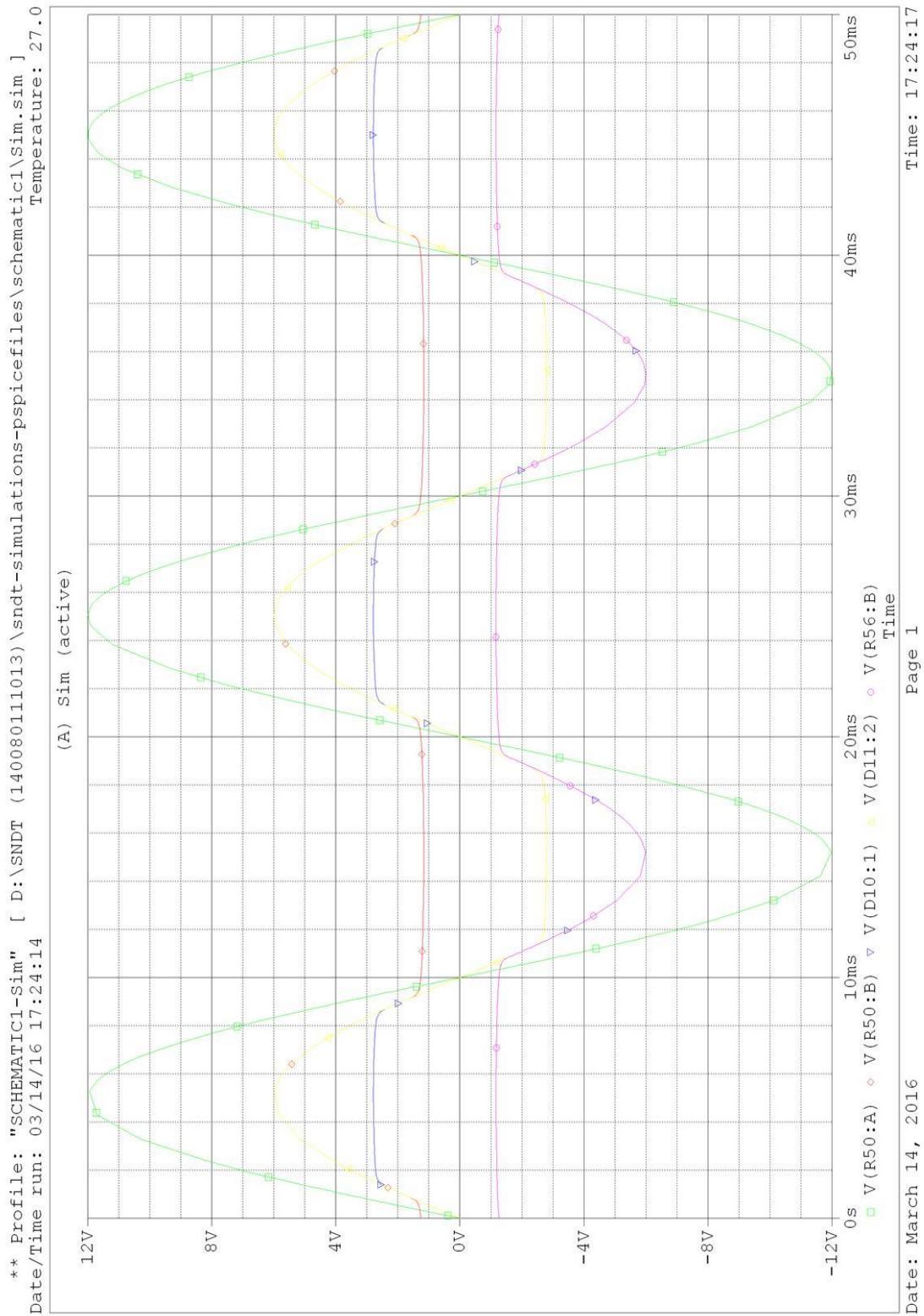
\* source CLIP2

V\_V16 N28775 0 1  
R\_R20 0 N28683 1k TC=0,0  
V\_V15 N28673 0  
+SIN 0 12 50 0 0 0  
R\_R19 N28673 N28683 1k TC=0,0  
D\_D1 N28775 N28683 D1N4002

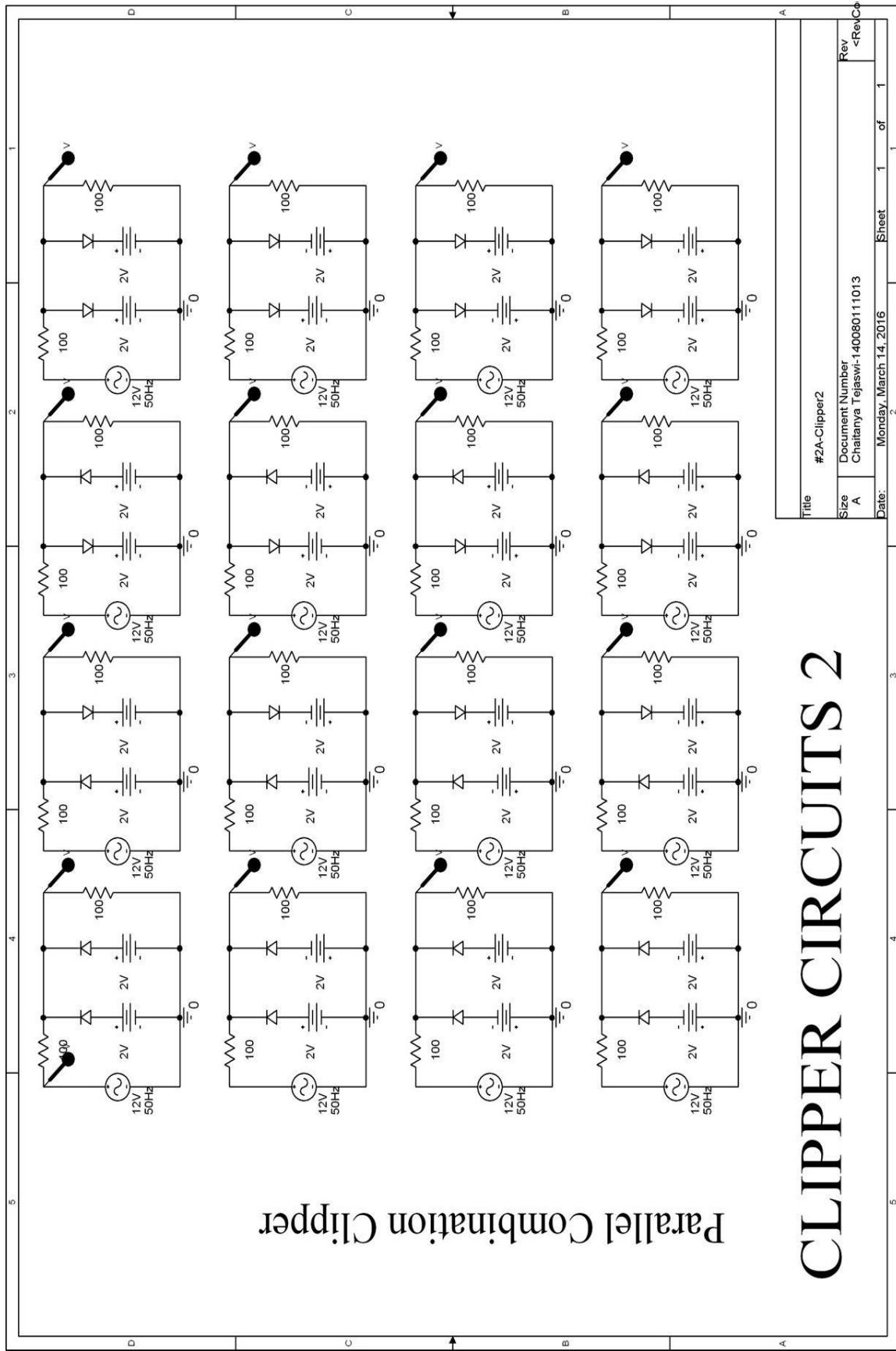
\* source CLIP3

V\_V16 N28775 0 1  
R\_R20 0 N28683 1k TC=0,0  
V\_V15 N28673 0  
+SIN 0 12 50 0 0 0  
R\_R19 N28673 N28683 1k TC=0,0  
D\_D1 N28683 N28775 D1N4002

## OUTPUT :



## CIRCUIT :



## **NETLIST :**

\* source CLIP1

V\_V16 0 N28775 1  
R\_R20 0 N28683 1k TC=0,0  
V\_V15 N28673 0  
+SIN 0 12 50 0 0 0  
R\_R19 N28673 N28683 1k TC=0,0  
D\_D1 N28683 N28775 D1N4002

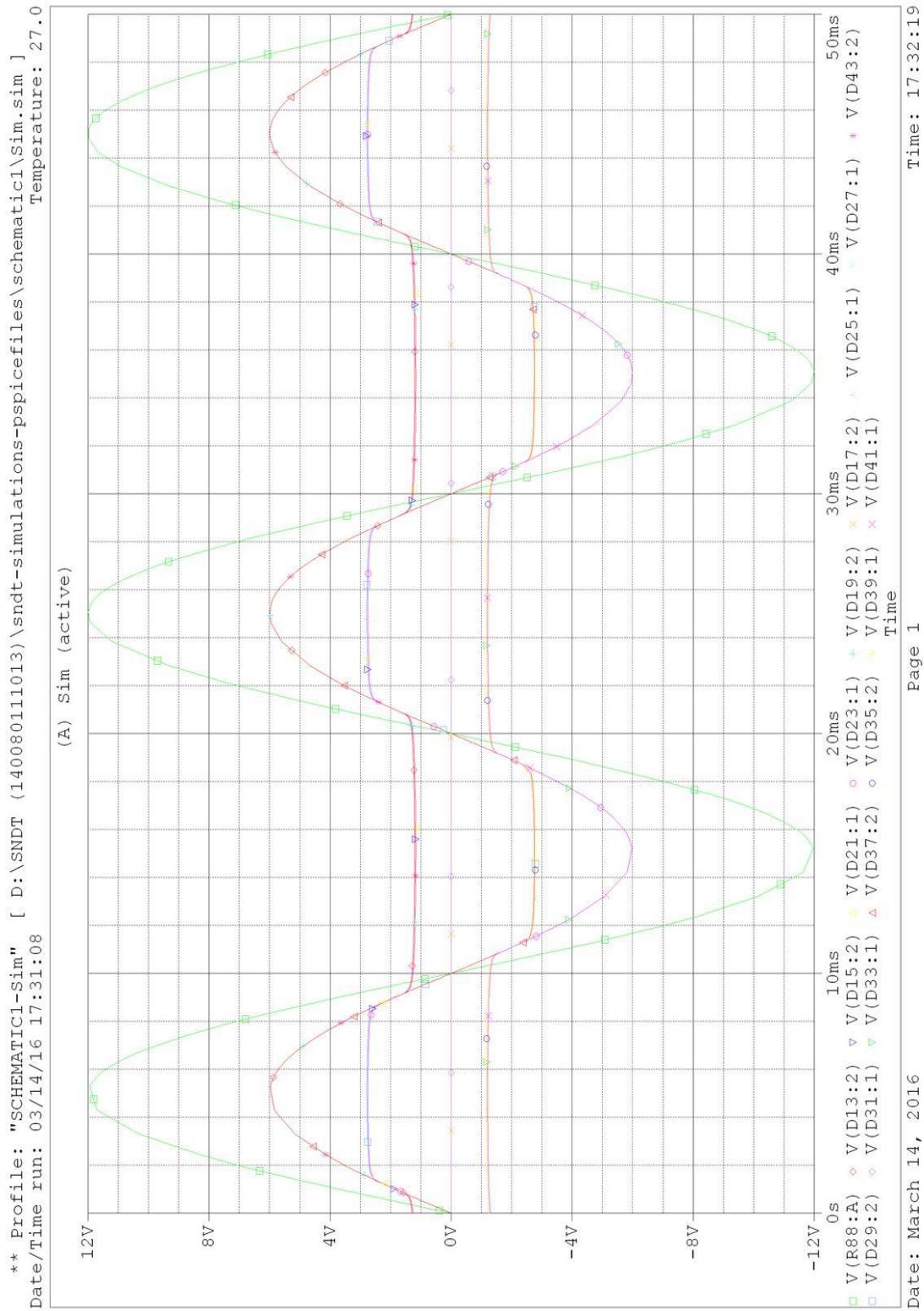
\* source CLIP2

V\_V16 0 N28775 1  
R\_R20 0 N28683 1k TC=0,0  
V\_V15 N28673 0  
+SIN 0 12 50 0 0 0  
R\_R19 N28673 N28683 1k TC=0,0  
D\_D1 N28775 N28683 D1N4002

\* source CLIP3

V\_V23 N29264 0  
+SIN 0 12 50 0 0 0  
V\_V24 N29366 0 1  
V\_V25 N29410 0 1.5  
R\_R28 0 N29274 1k TC=0,0  
R\_R27 N29264 N29274 1k TC=0,0  
D\_D16 N29366 N29274 D1N4002  
D\_D17 N29274 N29410 D1N4002

## OUTPUT :



## **CONCLUSION :**

By performing this practical, we conclude that any clipping circuits can be implemented and also stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

## PRACTICAL: 4

**AIM:** To study about the diode clamping circuits.

**SOFTWARE:** OrCAD capture CIS 16.3

### **THEORY:**

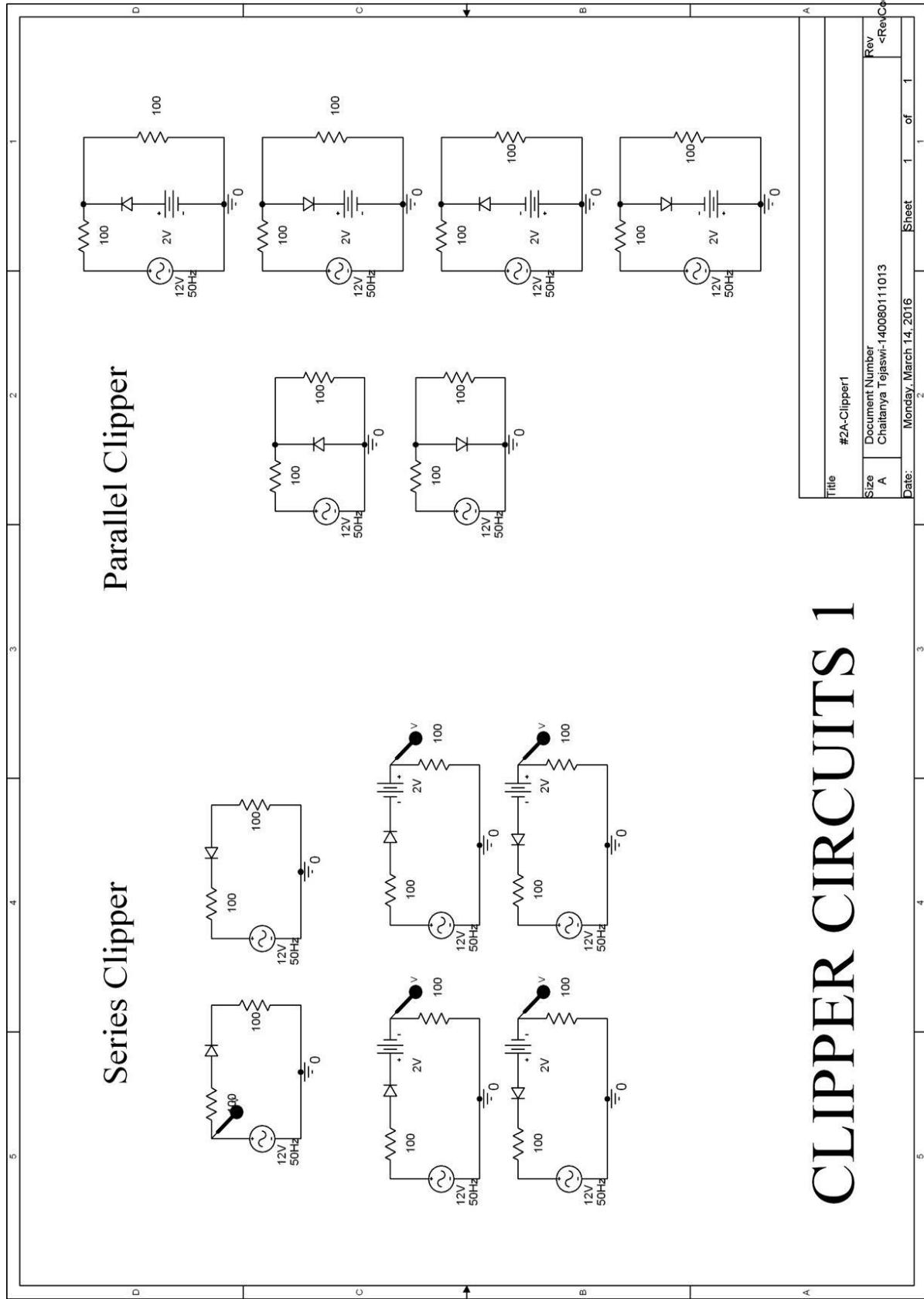
1. A Camper is an electronics circuit that fixes either the positive or the negative peak excursions of a signal to a defined value by shifting its DC value.
2. The clamper does not restrict the peak-to-peak excursion of the signal , it moves the whole signal up and down so as to place the peaks at the reference level.
3. A Diode clamper consist of a diode, which conduct the electric current in only one direction and prevent the signal exceeding the reference value ;and a capacitor which provides a DC Offset from the stored charge.
4. The capacitor forms a time constant with the resistor load which determines the range of frequencies over which the clamper will be effective.

### **PROCEDURE:**

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.

- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Now run the project using Pspice and note the characteristic.
- Then save the practical.

## CIRCUIT :



## **NETLIST :**

\* source CLAMP1

V\_V1 N00141 0 AC 0

+SIN 0 5 30khz 0 0 0

C\_C1 N00141 N00148 0.01uf

D\_D1 N00148 0 D1N4002

R\_R1 0 N00148 560k

\* source CLAMP2

C\_C2 N00821 N00855 0.01uf

D\_D2 0 N00855 D1N4002

R\_R2 0 N00855 560k

V\_V2 N00821 0 AC 0

+SIN 0 5 30khz 0 0 0

\* source CLAMPER1

R\_R3 0 N01393 560k

V\_V3 N01359 0 AC 0

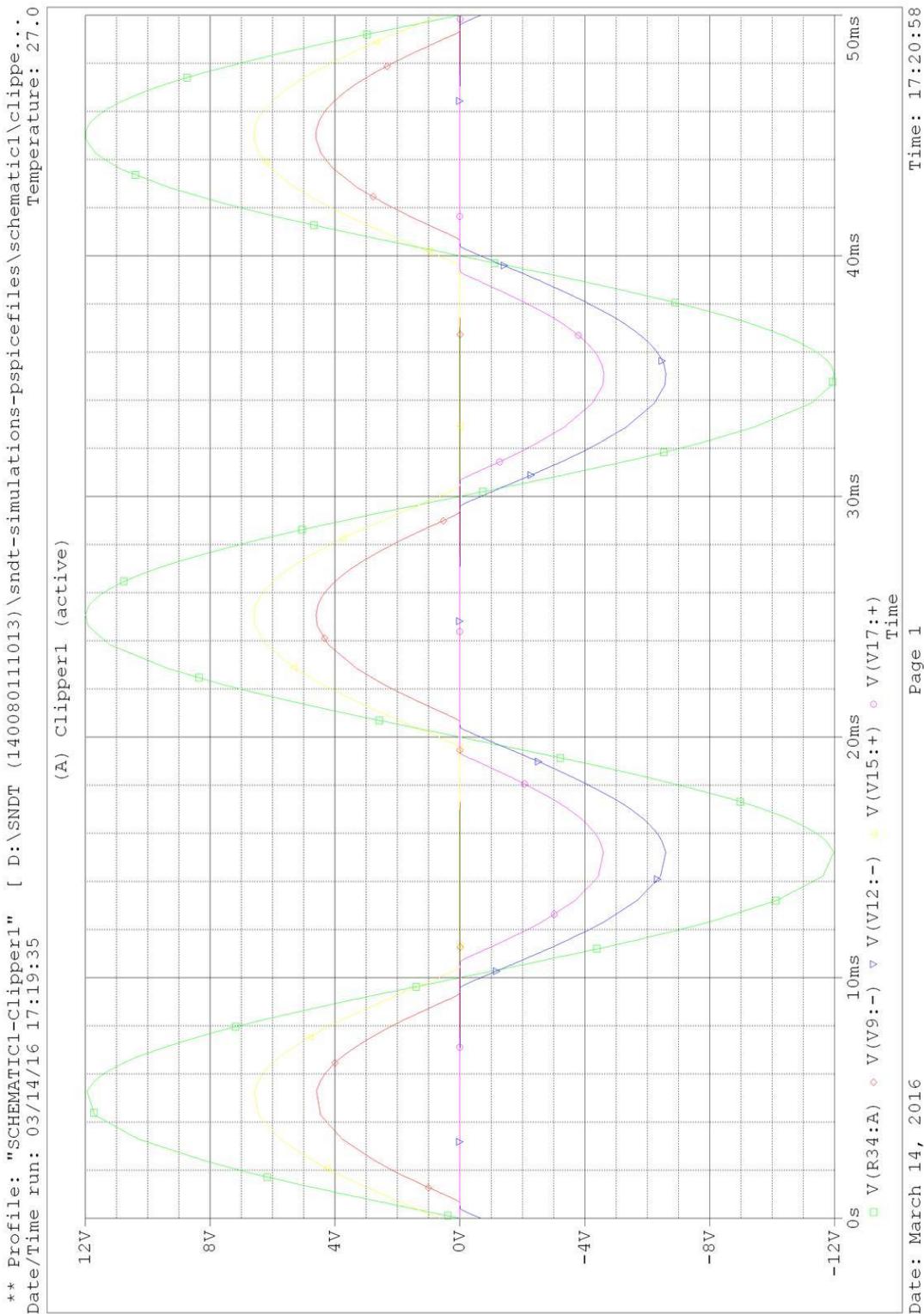
+SIN 0 5 30khz 0 0 0

D\_D3 N01393 N01600 D1N4002

C\_C3 N01359 N01393 0.01uf

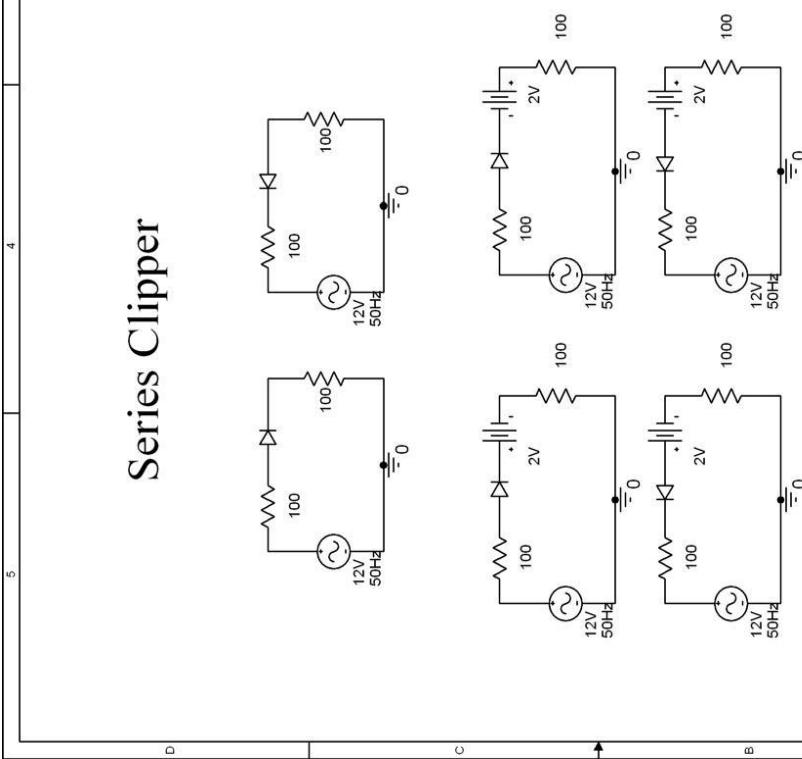
V\_V4 N01600 0 1Vdc

## **OUTPUT :**

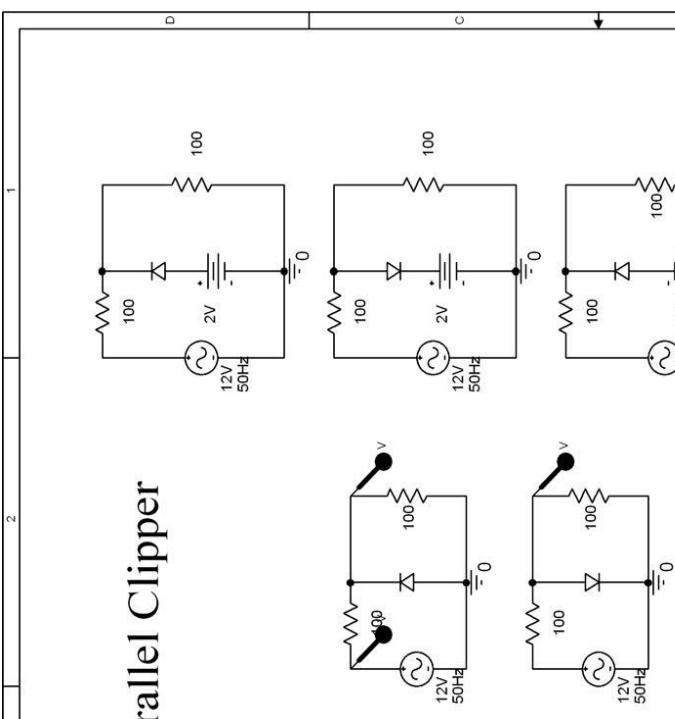


## CIRCUIT :

Series Clipper



Parallel Clipper



# CLIPPER CIRCUITS 1

Title	#2A-Clipper1
Size A	Document Number Chaitanya Tejswi-140080111013
Date:	Monday, March 14, 2016
Rev	<RevCo

## **NETLIST :**

\* source CLAMP1

V\_V1 N00141 0 AC 0

+SIN 0 5 30khz 0 0 0

C\_C1 N00141 N00148 0.01uf

D\_D1 N00148 0 D1N4002

R\_R1 0 N00148 560k

\* source CLAMP2

C\_C2 N00821 N00855 0.01uf

D\_D2 0 N00855 D1N4002

R\_R2 0 N00855 560k

V\_V2 N00821 0 AC 0

+SIN 0 5 30khz 0 0 0

\* source CLAMPER1

R\_R3 0 N01393 560k

V\_V3 N01359 0 AC 0

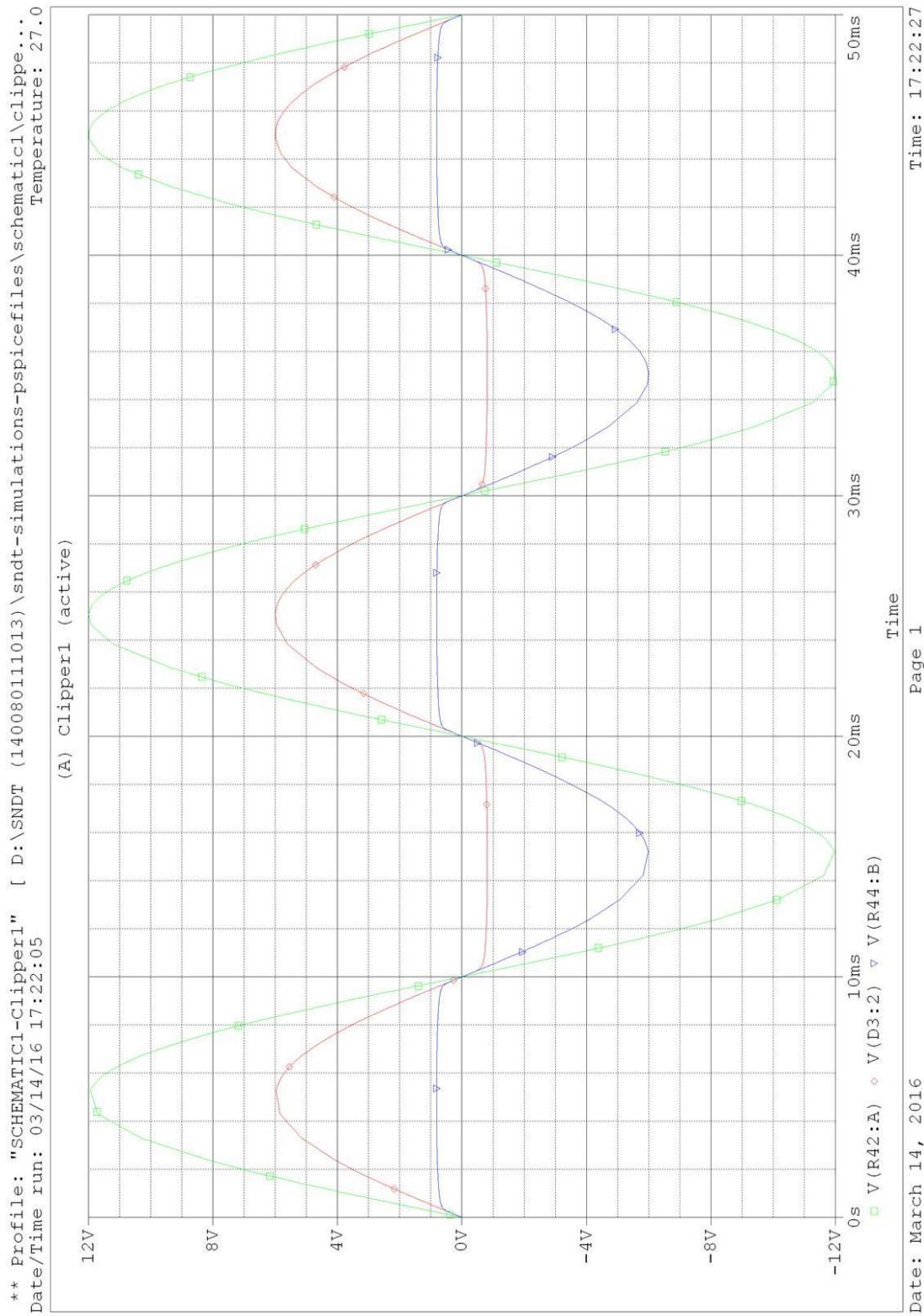
+SIN 0 5 30khz 0 0 0

D\_D3 N01393 N01600 D1N4002

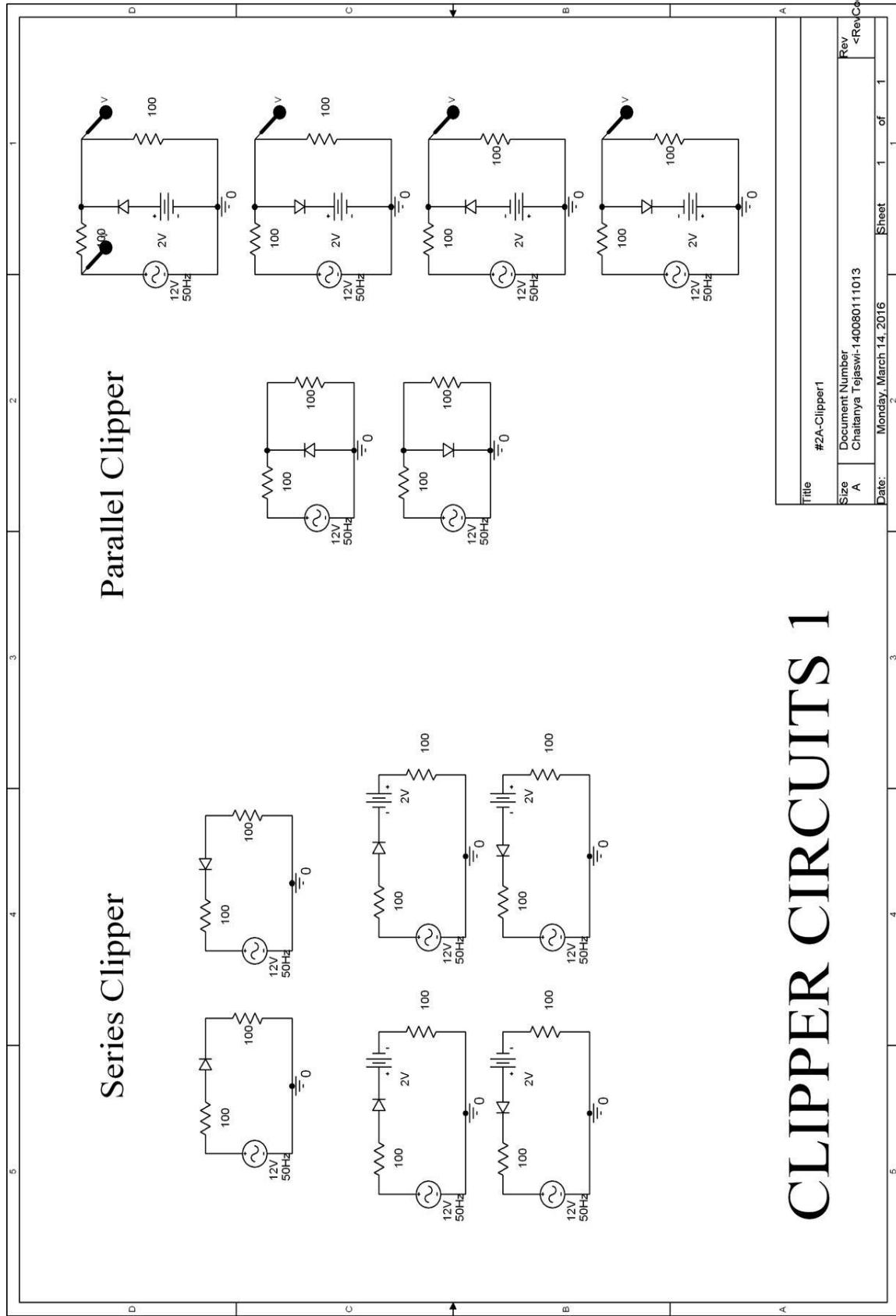
C\_C3 N01359 N01393 0.01uf

V\_V4 N01600 0 1Vdc

## OUTPUT :



## CIRCUIT :



## **NETLIST :**

\* source CLAMP1

V\_V1 N00141 0 AC 0

+SIN 0 5 30khz 0 0 0

C\_C1 N00141 N00148 0.01uf

D\_D1 N00148 0 D1N4002

R\_R1 0 N00148 560k

\* source CLAMP2

C\_C2 N00821 N00855 0.01uf

D\_D2 0 N00855 D1N4002

R\_R2 0 N00855 560k

V\_V2 N00821 0 AC 0

+SIN 0 5 30khz 0 0 0

\* source CLAMPER1

R\_R3 0 N01393 560k

V\_V3 N01359 0 AC 0

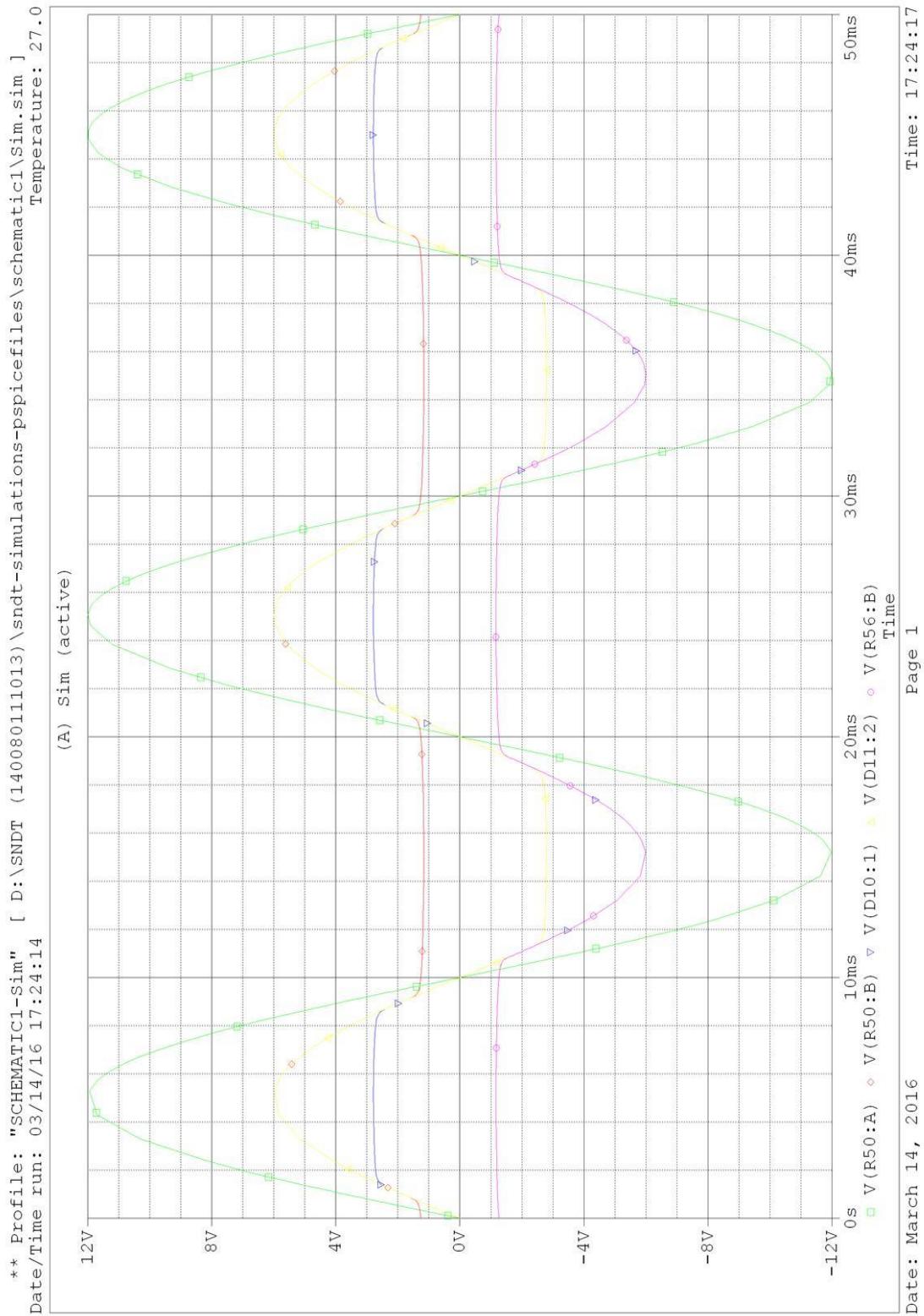
+SIN 0 5 30khz 0 0 0

D\_D3 N01393 N01600 D1N4002

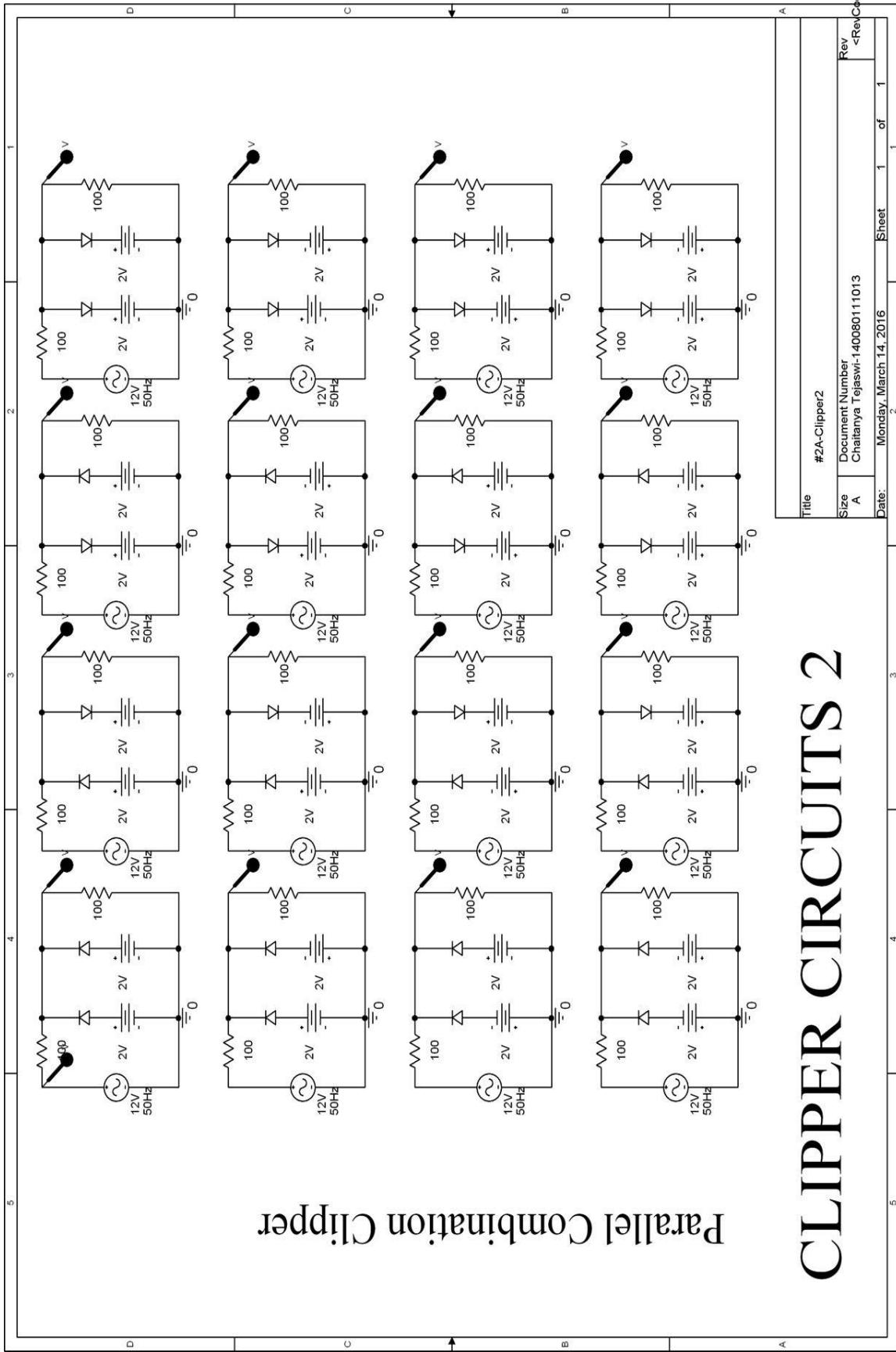
C\_C3 N01359 N01393 0.01uf

V\_V4 N01600 0 1Vdc

## OUTPUT :



## CIRCUIT :



## **NETLIST :**

\* source CLAMP1

V\_V1 N00141 0 AC 0

+SIN 0 5 30khz 0 0 0

C\_C1 N00141 N00148 0.01uf

D\_D1 N00148 0 D1N4002

R\_R1 0 N00148 560k

\* source CLAMP2

D\_D5 N02475 N02525 D1N4002

V\_V7 N02419 0 AC 0

+SIN 0 5 30khz 0 0 0

R\_R5 0 N02475 560k

V\_V8 0 N02525 1Vdc

C\_C5 N02419 N02475 0.01uf

\* source CLAMPER1

R\_R3 0 N01393 560k

V\_V3 N01359 0 AC 0

+SIN 0 5 30khz 0 0 0

D\_D3 N01393 N01600 D1N4002

C\_C3 N01359 N01393 0.01uf

V\_V4 N01600 0 1Vdc

D\_D5 N02475 N02525 D1N4002

V\_V7 N02419 0 AC 0

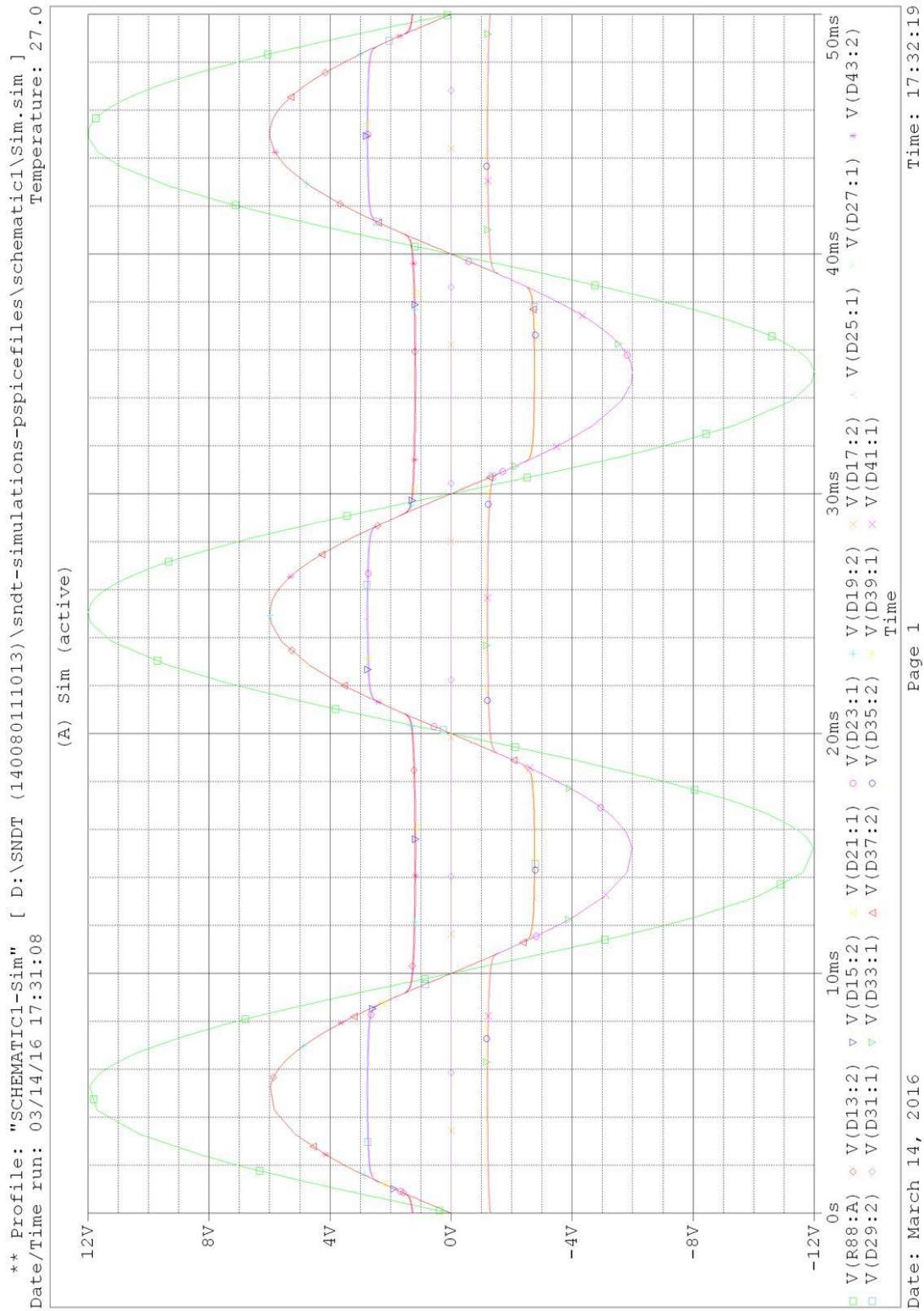
+SIN 0 5 30khz 0 0 0

R\_R5 0 N02475 560k

V\_V8 0 N02525 1Vdc

C\_C5 N02419 N02475 0.01uf

## OUTPUT :



## **CONCLUSION :**

By performing this practical, we conclude that any clipping circuits can be implemented and also stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

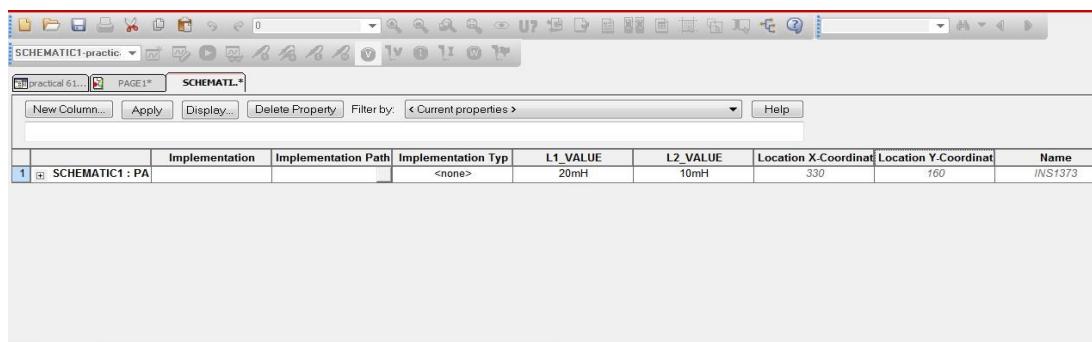
## PRACTICAL: 5

**AIM:** To study about Half wave rectifier.

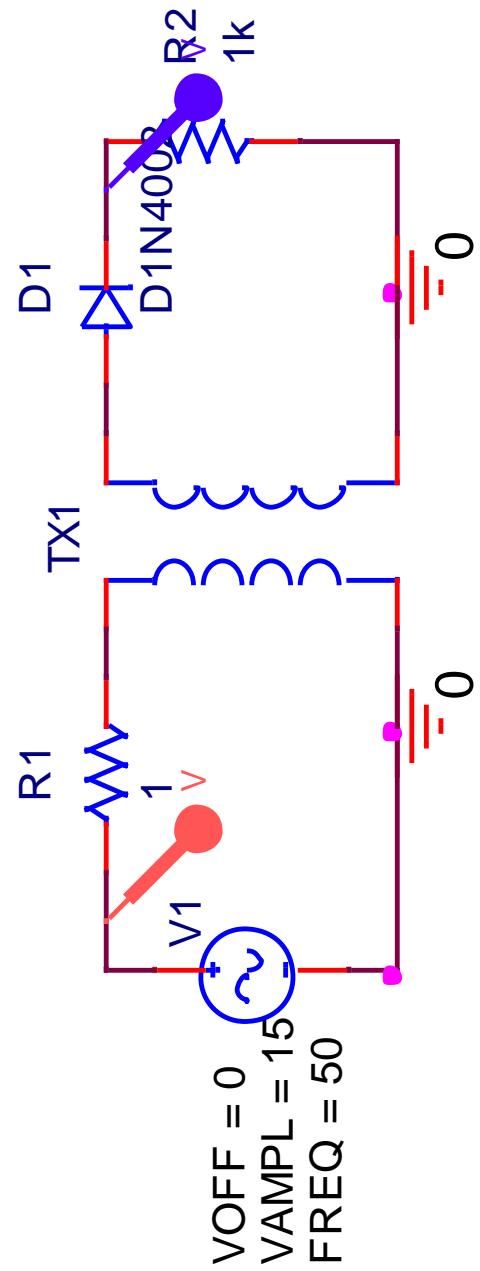
**SOFTWARE USED:** Orcad capture CIS 16.3

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Then save the practical. The Change is to be made when working with transformers as per shown



## CIRCUIT :



Half wave rectifier

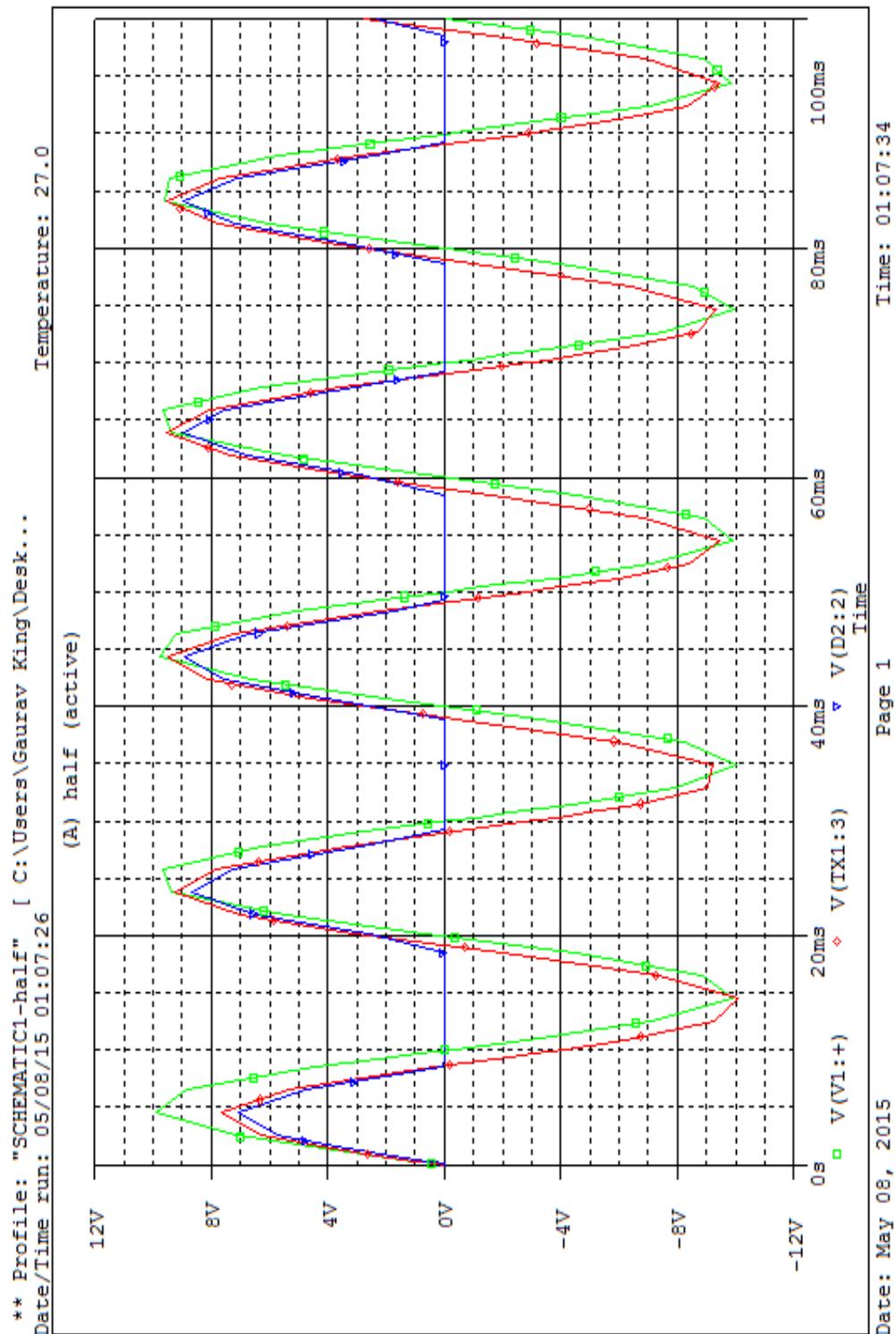
## NET LIST

```
* source HALFWAVE RECTIFIER
V_V1      N00137 0
+SIN 0 10 50 0 0 0
R_R1      N00137 N00144 .001 TC=0,0
R_R2      0 N00198 10k TC=0,0
X_TX1    N00144 0 N00160 0 SCHEMATIC1_TX1
D_D2      N00160 N00198 D1N4002

.subckt SCHEMATIC1_TX1 1 2 3 4
K_TX1    L1_TX1 L2_TX1 1
L1_TX1   1 2 10uH
L2_TX1   3 4 10uH
.ends SCHEMATIC1_TX1

**** RESUMING half.cir ****
.END
```

# OUTPUT



## **CONCLUSION :**

By performing this practical, we conclude that any clipping circuits can be implemented and also stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

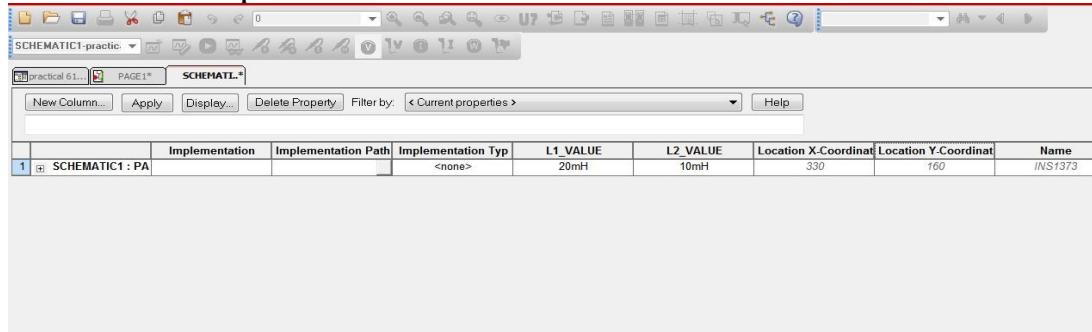
# PRACTICAL: 6

**AIM:** To study about Full wave rectifier.

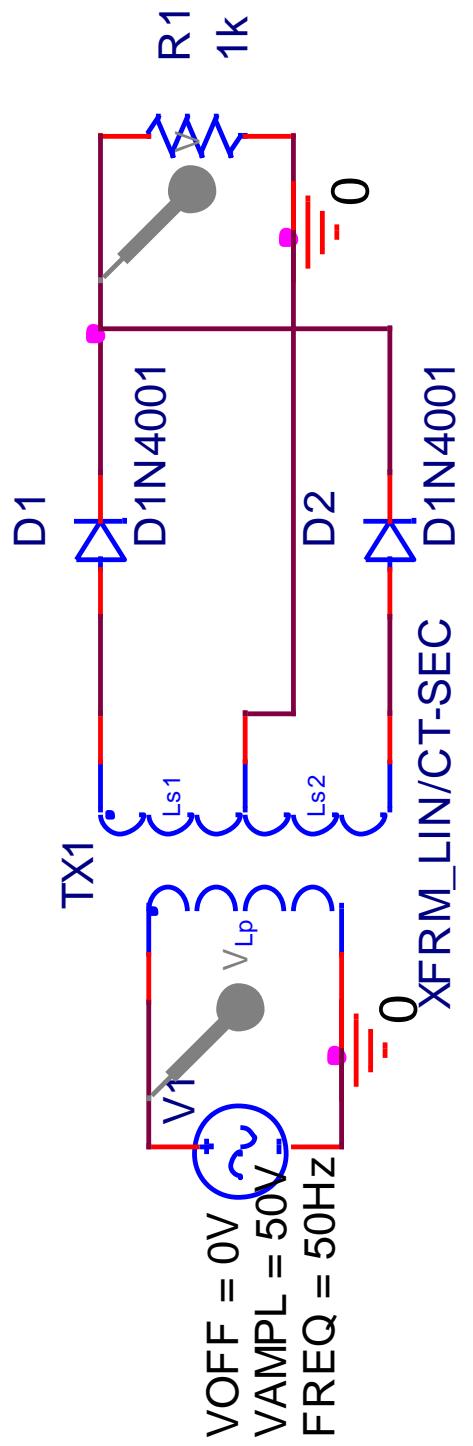
**SOFTWARE USED:** Orcad capture CIS 16.3

## PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Then save the practical.
- Then save the practical. The Change is to be made when working with transformers as per shown



## CIRCUIT :

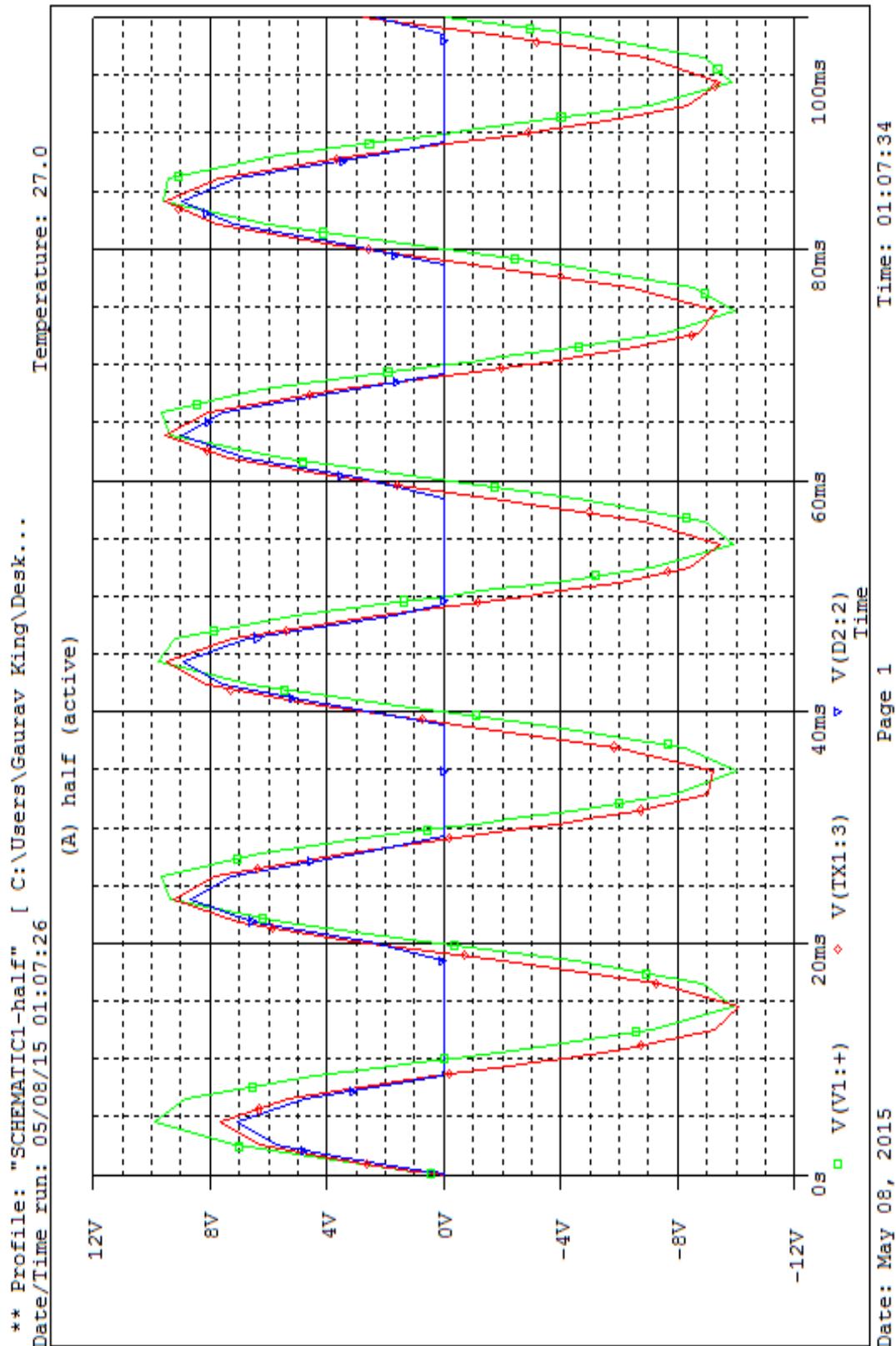


Full wave rectifier

## **NET LIST**

```
* source FULLWAVE RECTIFIER
X_TX1      N00091 0 N00206 0 N00213 XFRM_LIN/CT-SEC PARAMS:
LP_VALUE=20MH
+ LS1_VALUE=10MH LS2_VALUE=10MH COUPLING=.99
RP_VALUE=0.1 RS_VALUE=0.1
V_V1       N00091 0
+SIN 0 10 50 0 0 0
R_R1       0 N00220 1k TC=0,0
D_D1       N00206 N00220 D1N4002
D_D2       N00213 N00220 D1N4002
```

# OUTPUT



## **CONCLUSION :**

By performing this practical, we conclude that any clipping circuits can be implemented and also stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

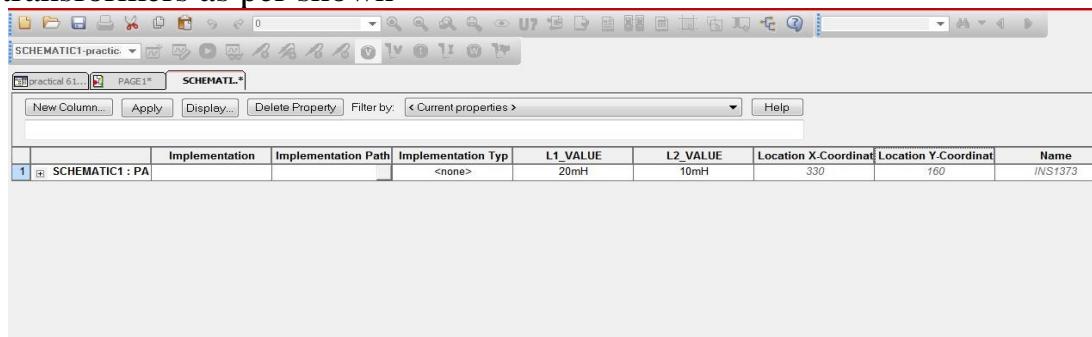
## PRACTICAL: 7

**AIM:** To study about Bridge rectifier.

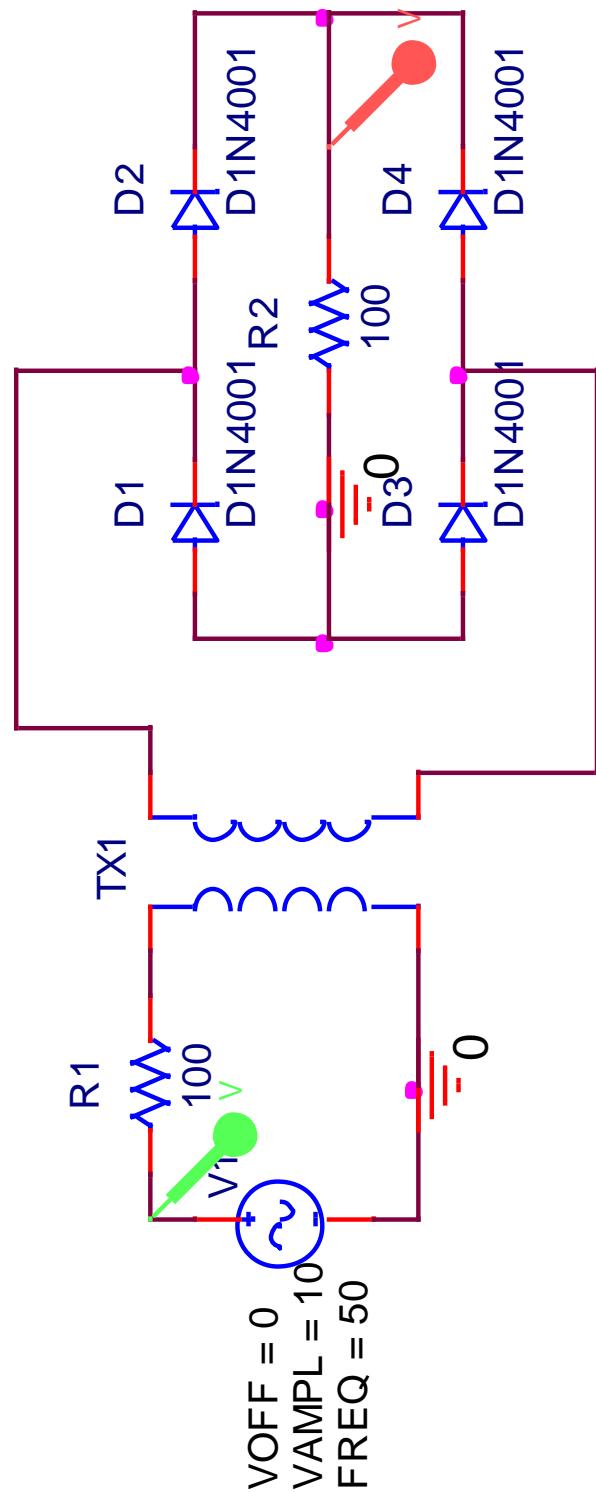
**SOFTWARE USED:** Orcad capture CIS 16.3

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Then save the practical. .
- Then save the practical. The Change is to be made when working with transformers as per shown



## CIRCUIT :



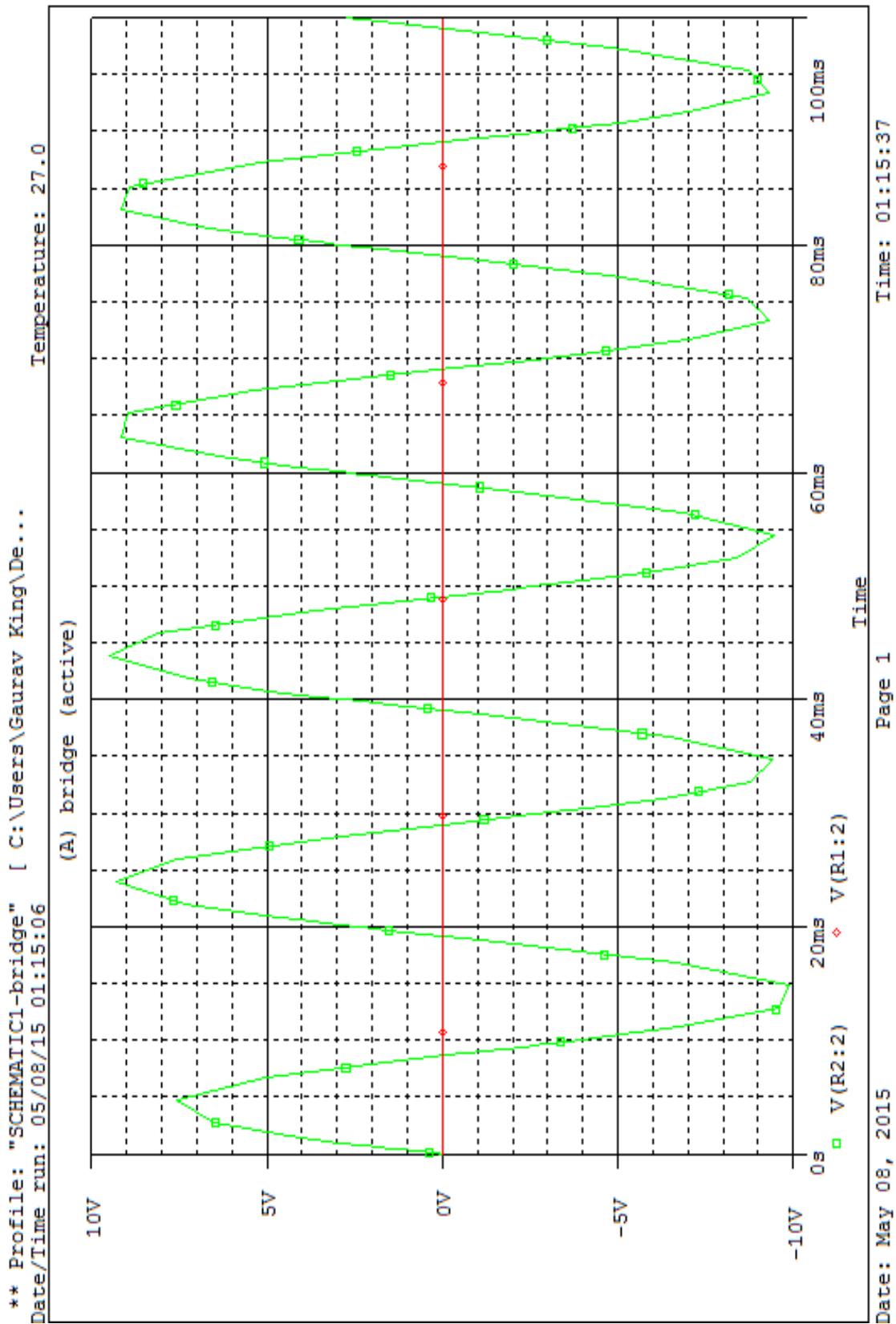
Bridge wave rectifier

## NET LIST

```
* source BRIDGE RECTIFIER
X_TX1 N00429 0 N00226 N00230 SCHEMATIC1_TX1
V_V1      N00087 0
+SIN 0 10 50 0 0 0
R_R1      N00234 0 1k TC=0,0
R_R2      N00087 N00429 .001 TC=0,0
D_D5      0 N00226 D1N4002
D_D6      0 N00230 D1N4002
D_D7      N00226 N00234 D1N4002
D_D8      N00230 N00234 D1N4002

.subckt SCHEMATIC1_TX1 1 2 3 4
K_TX1      L1_TX1 L2_TX1 1
L1_TX1    1 2 10uH
L2_TX1    3 4 10uH
.ends SCHEMATIC1_TX1
.ends SCHEMATIC1_TX
```

## OUTPUT



## **CONCLUSION :**

By performing this practical, we concluded that half wave, full wave and bridge rectifier can be implemented and also can be stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

## PRACTICAL: 08

**AIM:** To study about diode characteristics.

**SOFTWARE:** Orcad capture CIS 16.3

### THEORY:

In electronics, a **diode** is a two-terminal electronic component that conducts primarily in one direction (asymmetric conductance); it has low (ideally zero) resistance to the flow of current in one direction, and high (ideally infinite) resistance in the other. A **semiconductor diode**, the most common type today, is a crystalline piece of semiconductor material with a p–n junction connected to two electrical terminals.<sup>[5]</sup>

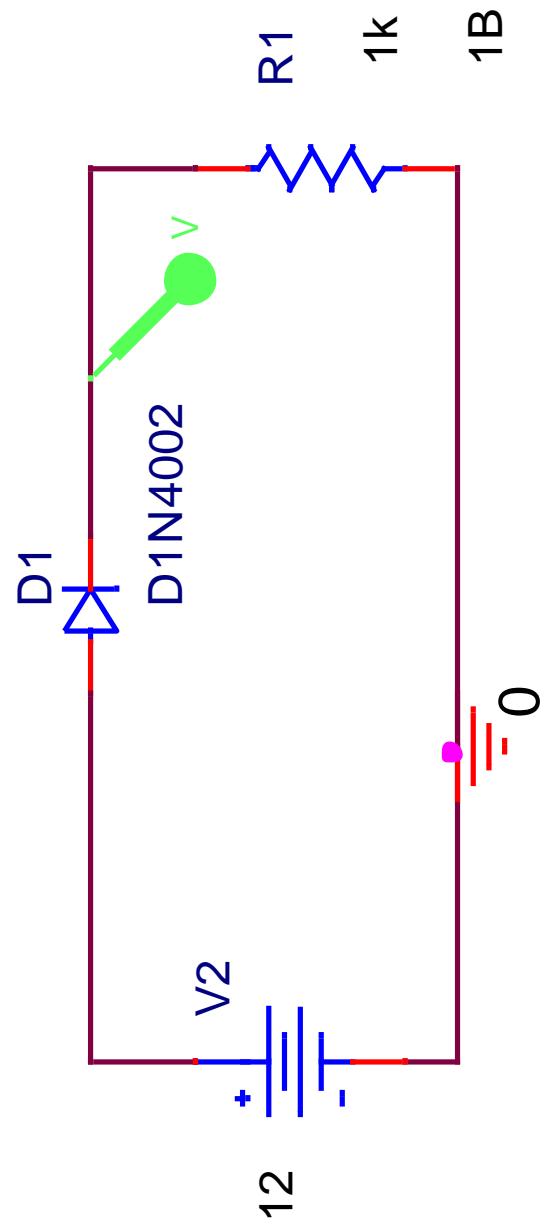
A vacuum tube diode has two electrodes, a plate (anode) and a heated cathode. Semiconductor diodes were the first semiconductor electronic devices. The discovery of crystals' rectifying abilities was made by German physicist Ferdinand Braun in 1874.

The first semiconductor diodes, called cat's whisker diodes, developed around 1906, were made of mineral crystals such as galena. Today, most diodes are made of silicon, but other semiconductors such as selenium or germanium are sometimes used

## **PROCEDURE:**

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Now run the project using Pspice and note the waveforms for diode characteristic.
- Then, save the practical.

## CIRCUIT DIAGRAM:



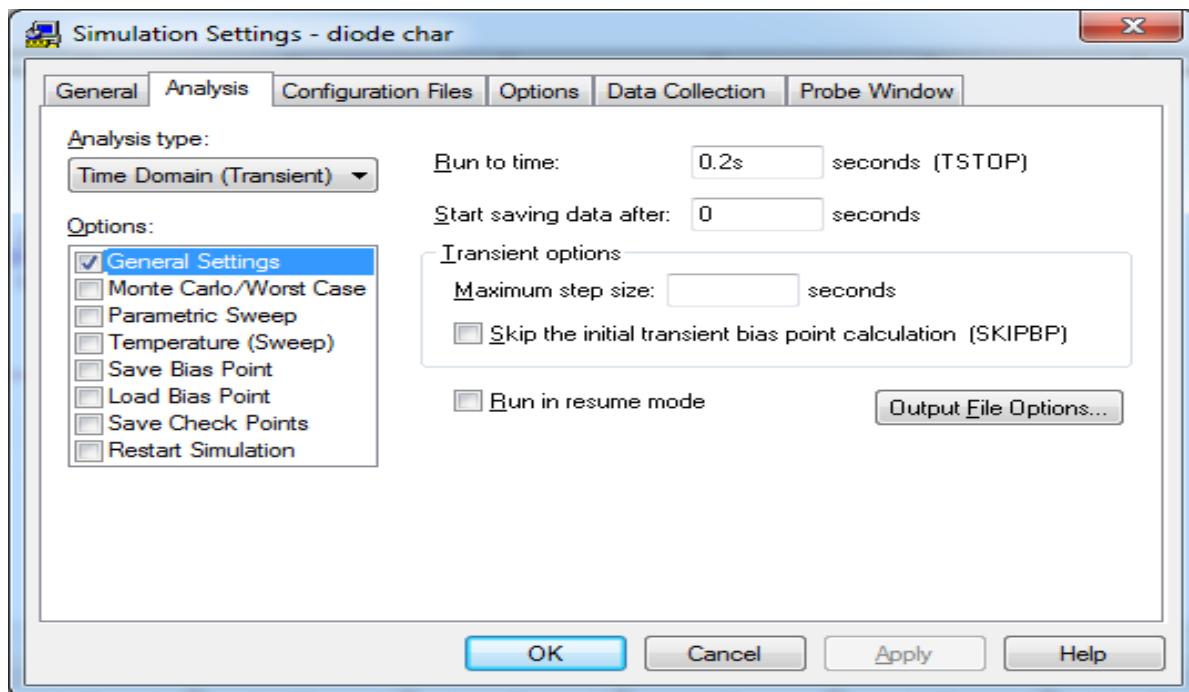
## NET LIST

\* source DIODE CHAR

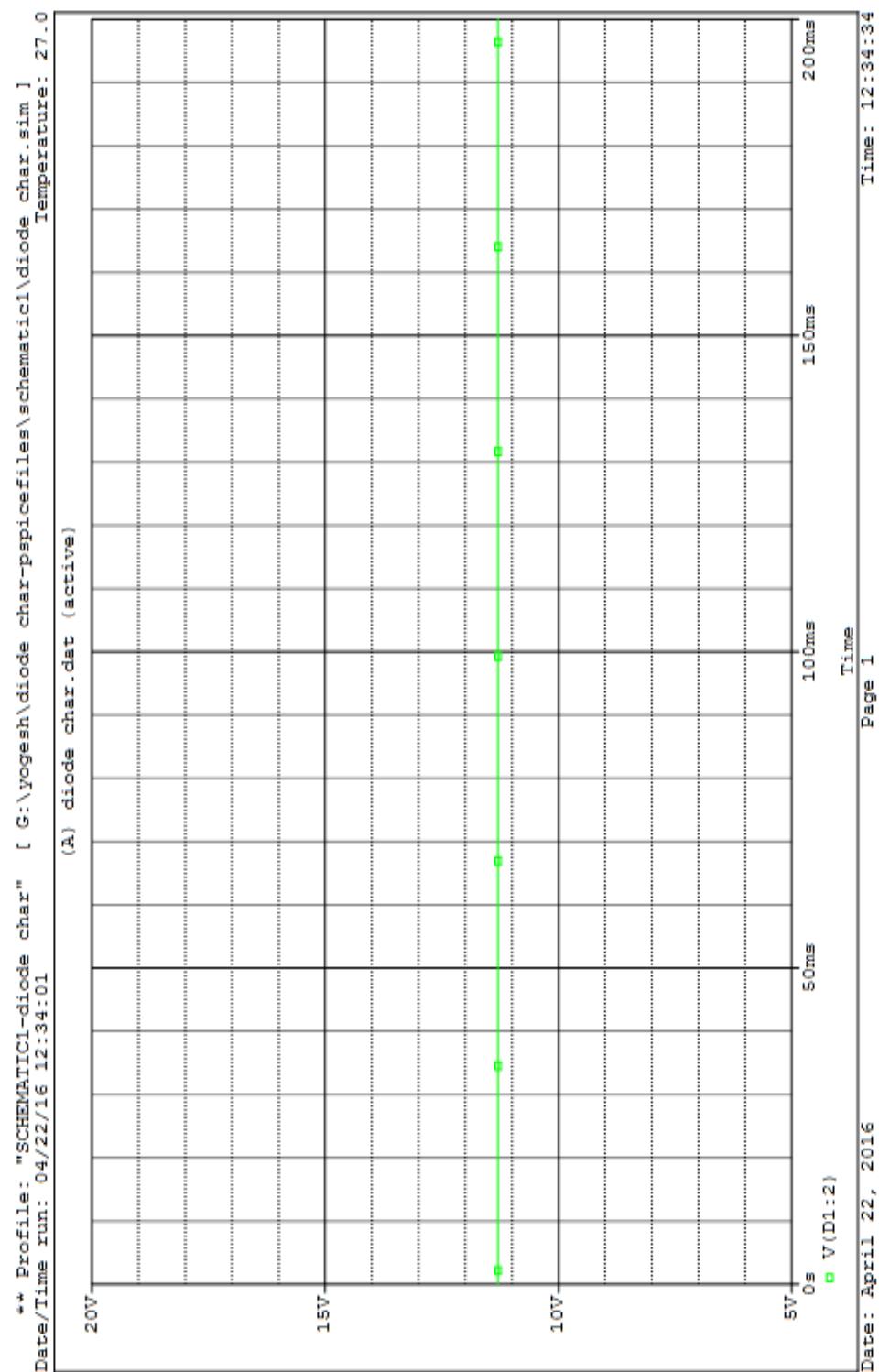
D\_D1 N00142 N00149 D1N4002

R\_R1 0 N00149 1k

V\_V2 N00142 0 12



## OUTPUT



## **CONCLUSION:**

By performing this practical, we conclude that diode characteristics can be implemented and also stimulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect stimulation we can implement it in the practical world.

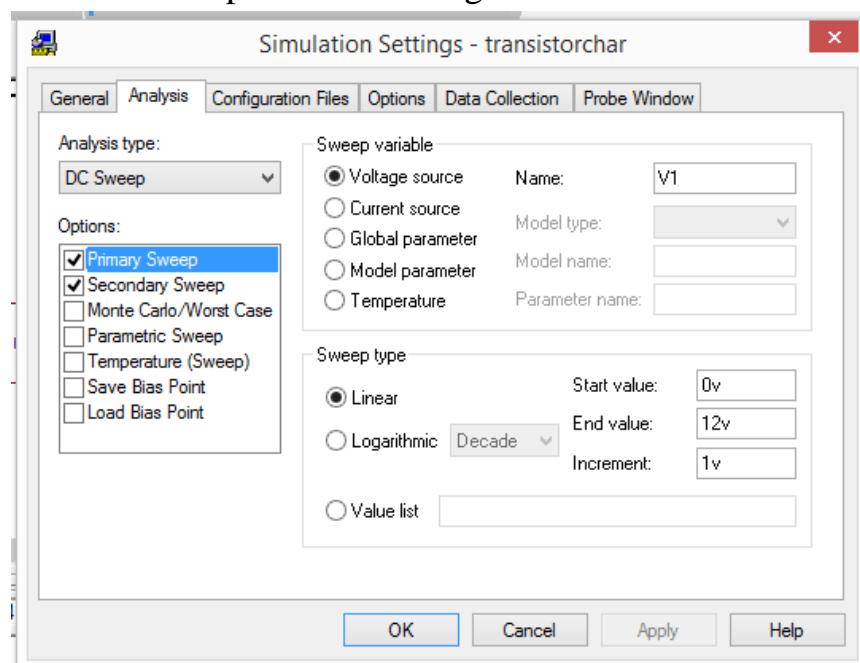
## PRACTICAL: 9

**AIM:** To study the characteristic of transistor.

**SOFTWARE:** OrCAD capture CIS 16.3

### PROCEDURE:

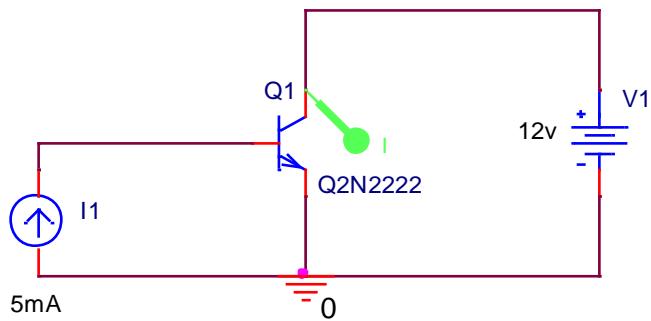
- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.



- Now, stimulate the circuit by clicking the run button.

- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Now run the project using Pspice and note the characteristic.
- Then save the practical.

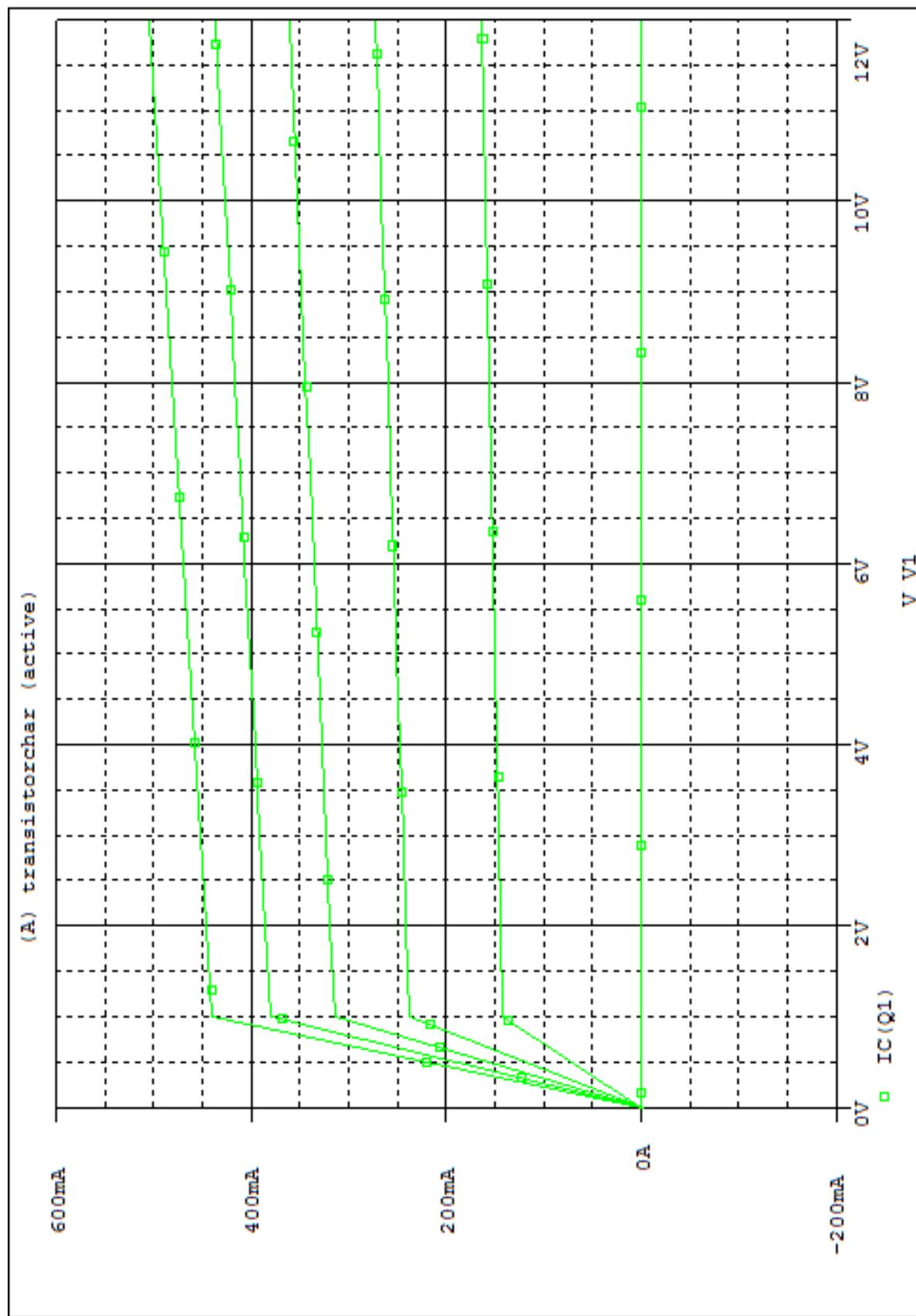
## CIRCUIT:



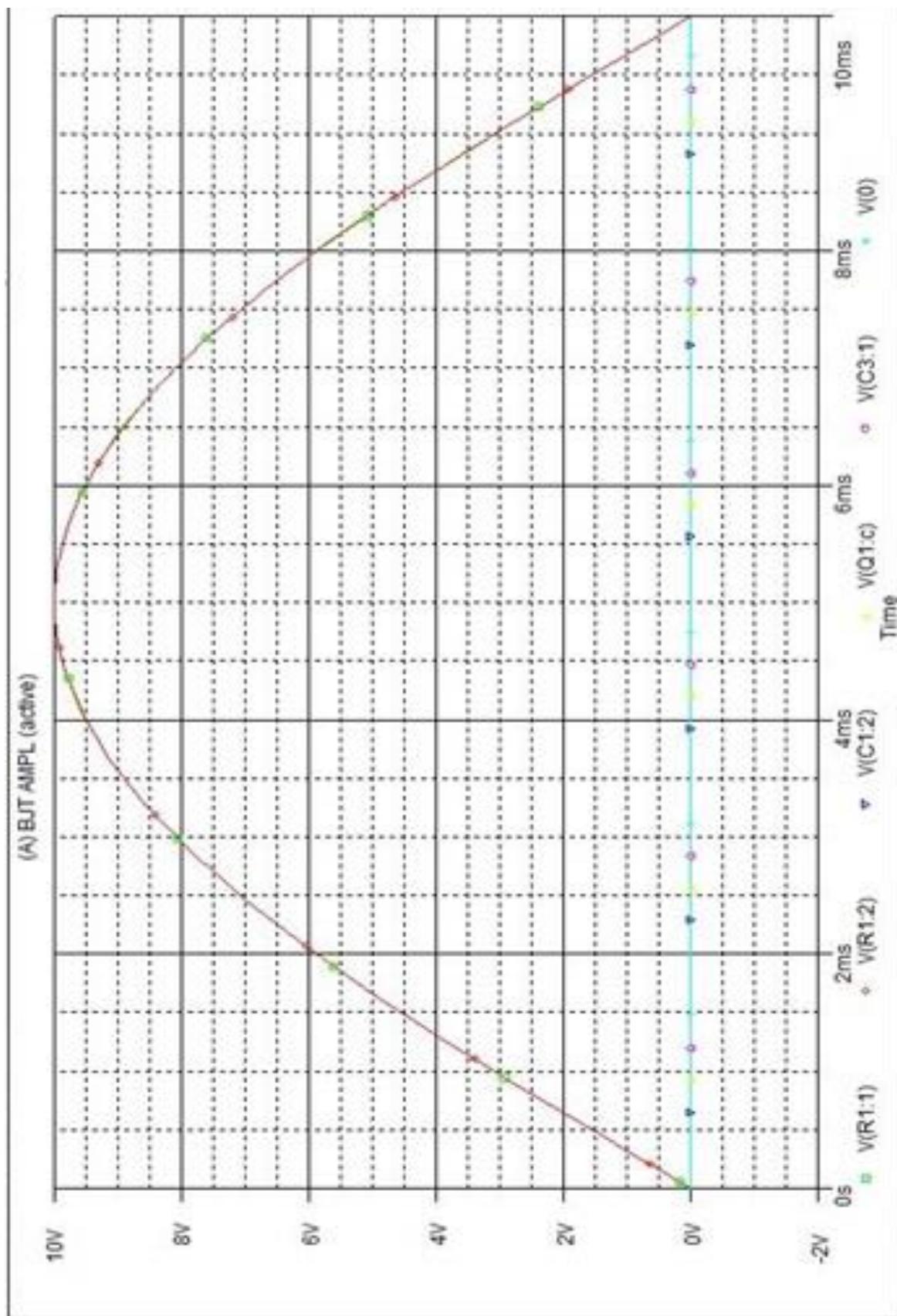
## NET LIST

```
* source TRANSISTORCHAR
Q_Q1      N00169 N00162 0 Q2N2222A
I_I1      0 N00162 DC 5mA
V_V1      N00169 0 12V
```

## GRAPH



## GRAPH



## **CONCLUSION:**

Thus from this practical we got to know about the transistor characteristics.

## PRACTICAL: 10

**AIM:** To study about Transient Frequency response of BJT Amplifier.

**Software used:** ‘Cadence OrCAD Capture 16.3’.

### Procedure:

- For any type of simulation design we got to have above software i.e. ‘Cadence OrCAD Capture 16.3’ installed in your pc where we do the required simulation using Pspice.
- So first open the required software by following the steps below:  
Start> all programs> cadence> release 16.3> OrCAD capture
- Then Create a new project by following the steps below:
  - Enter project name
  - Select analog or mixed A/D
  - File> new> project>
  - Create a folder for the project
  - Click ok
- After you have completed your project (i.e. made a required circuit), you got to simulate it using ‘Pspice’. For that follow the steps below:
- **Creating a new simulation profile and setting up** Select “Create a blank project” and Click OK

You should now see a blank schematic diagram

### 1. The simulation:

1. Pspice> new simulation profile.
2. Give a name to the ‘New simulation profile’ and then click to ‘Create’.
3. Then on the ‘Analysis Tab’ in the window named ‘Simulation Settings’, make ‘Analysis Type’ to ‘AC sweep/noise’.>
4. Then under ac sweep type> logarithmic>decade.

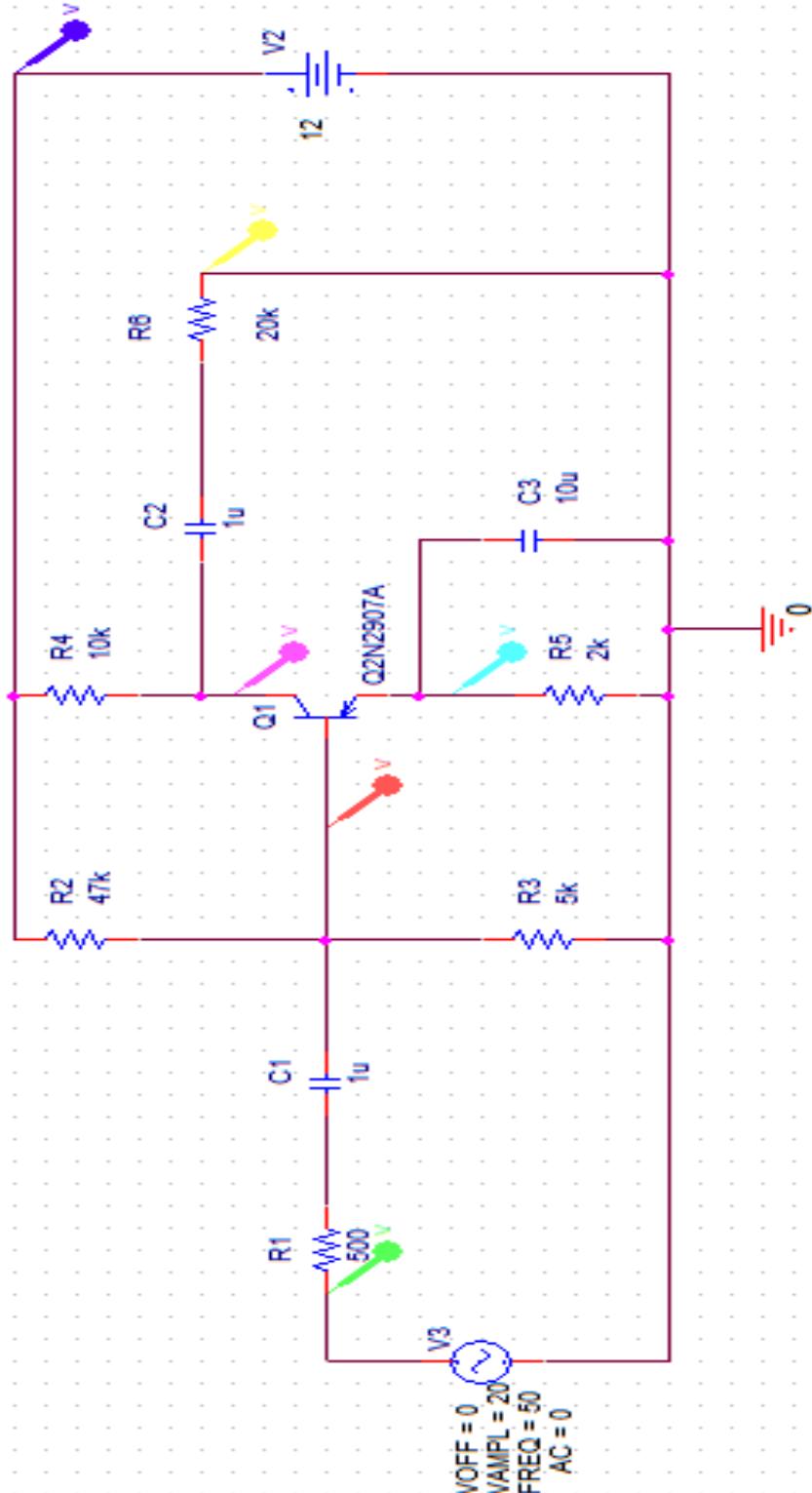
5. Keep start frequency as '1' and end frequency as '1000k' and points/decade as '101'.
6. Click on apply and ok.

## **2. Simulating the circuit and observing the simulation results:**

1. Click on the 'Run' icon to start the simulation.
2. Observe the 'Simulation Graph' carefully.
3. You can even print if it required.

**\*(note that here we are using the 'PNP-Transistor' named 'Q2N2907A')**

## CIRCUIT DIAGRAM :

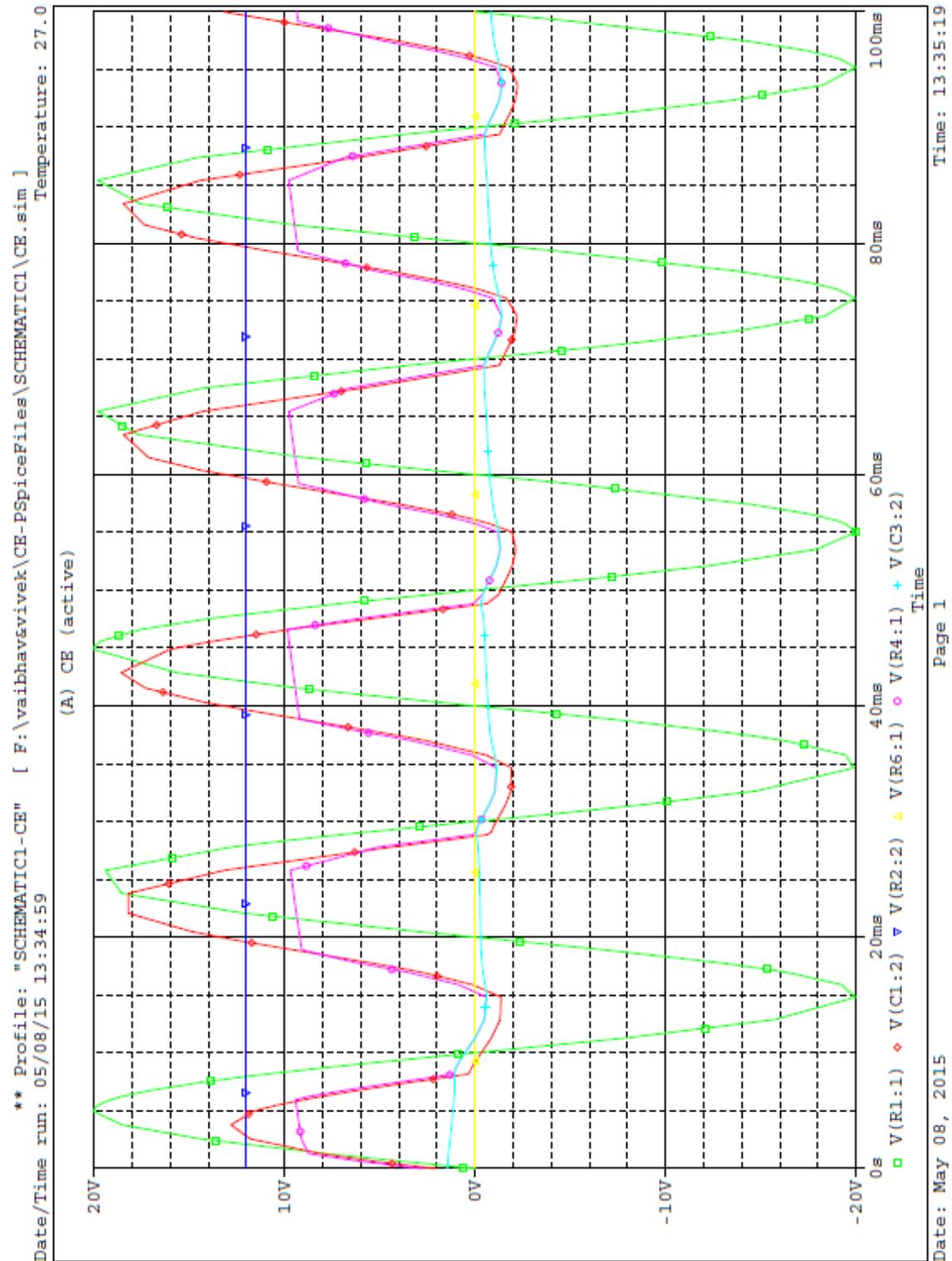


## **NETLIST:**

\* source CE

R\_R4 N01310 N01348 10k TC=0,0  
C\_C2 N01310 N01342 1u TC=0,0  
C\_C1 N01432 N01400 1u TC=0,0  
R\_R1 N01422 N01432 500 TC=0,0  
R\_R2 N01400 N01348 47k TC=0,0  
C\_C3 0 N01314 10u TC=0,0  
R\_R3 0 N01400 5k TC=0,0  
Q\_Q1 N01310 N01400 N01314 Q2N2907A  
R\_R6 0 N01342 20k TC=0,0  
V\_V2 N01348 0 12  
R\_R5 0 N01314 2k TC=0,0  
V\_V3 N01422 0 AC 0  
+SIN 0 20 50 0 0 0  
C\_C3 0 N00280 10u TC=0,0  
V\_V2 N00325 0 15Vdc  
Q\_Q1 N00275 N00484 N00280 Q2N2907A

## GRAPH:



## **CONCLUSION :**

After performing this practical, we conclude that we can study ‘Transient Frequency response of BJT Amplifier.

## PRACTICAL: 11

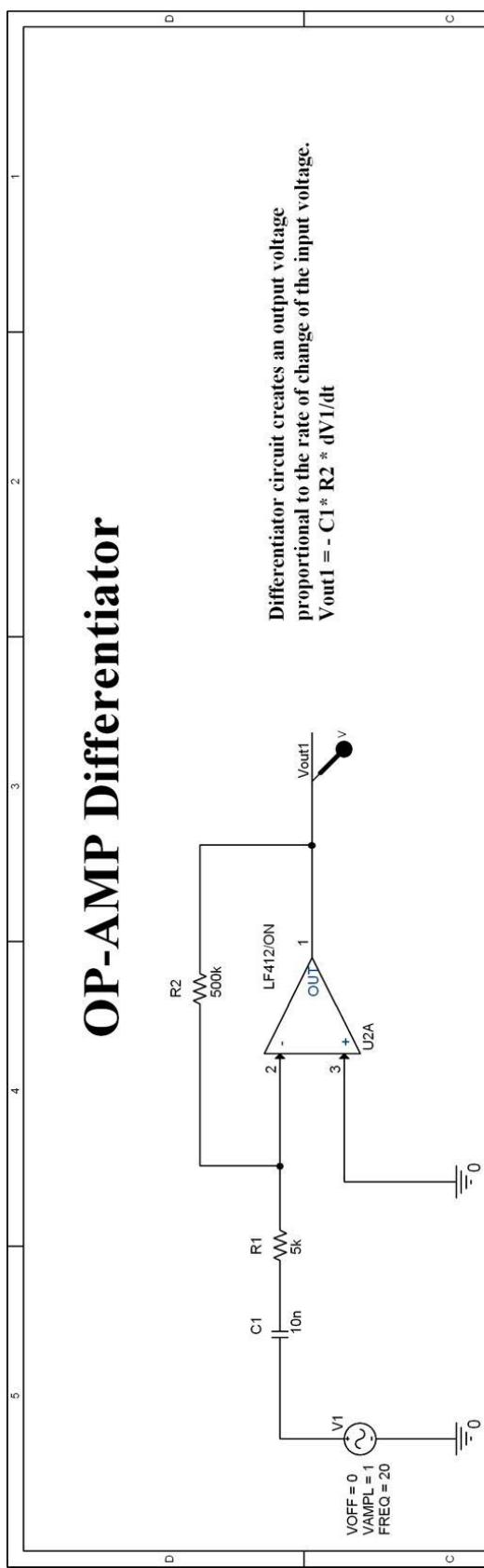
**AIM:** To Simulate OPAMP as Differentiator & Integrator using OrCAD Capture 16.3

**SOFTWARE:** Orcad capture CIS 16.3

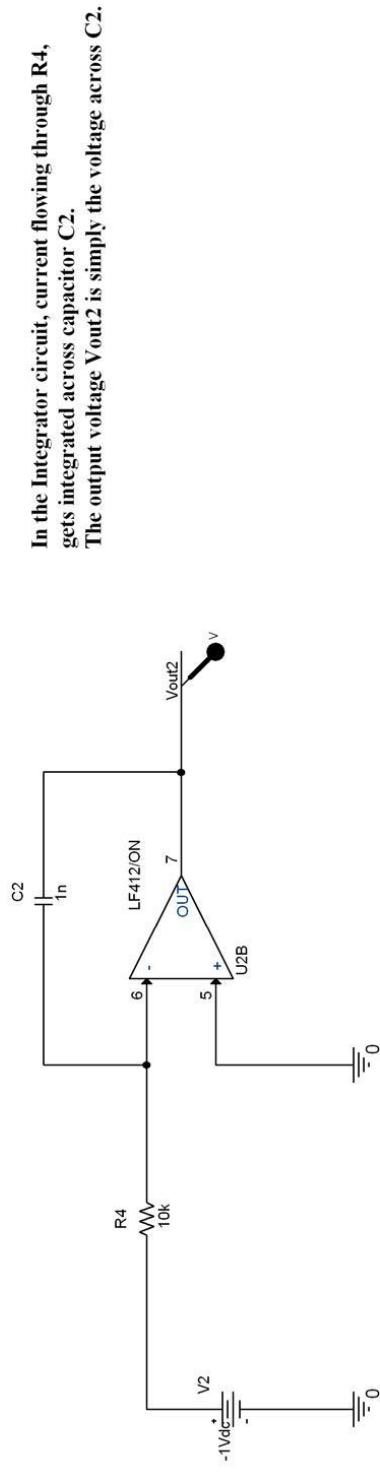
### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Now run the project using Pspice.
- Then save the practical.

## OP-AMP Differentiator



## OP-AMP Integrator

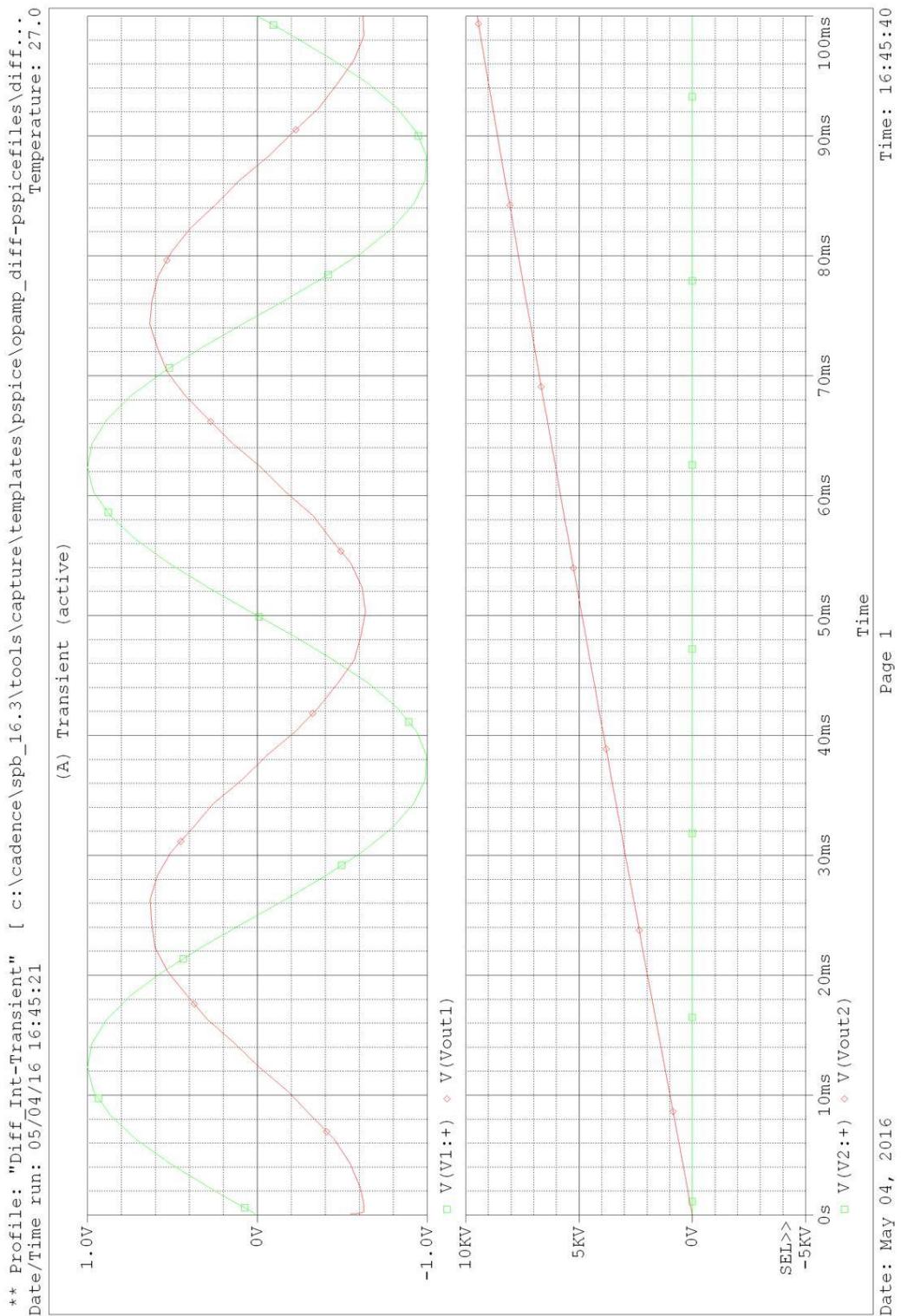


**CIRCUIT :**

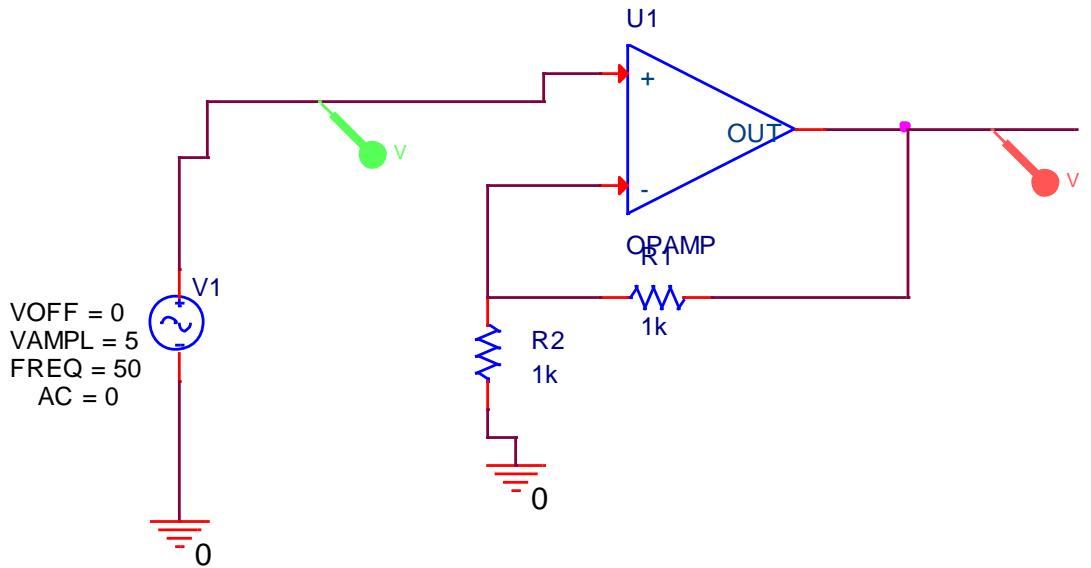
## **NET LIST**

```
* source OPAMP_DIFF
R_R1      N00100 N00096 5k TC=0,0
C_C1      N00143 N00100 10n TC=0,0
V_V1      N00143 0
+SIN 0 1 20 0 0 0
R_R2      N00096 VOUT1 500k TC=0,0
X_U2A      0 N00096 VOUT1 OPAMP1
X_U2B      0 N06633 VOUT2 OPAMP2
R_R4      N06677 N06633 10k TC=0,0
C_C2      N06633 VOUT2 1n IC=0v TC=0,0
V_V2      N06677 0 -1Vdc
```

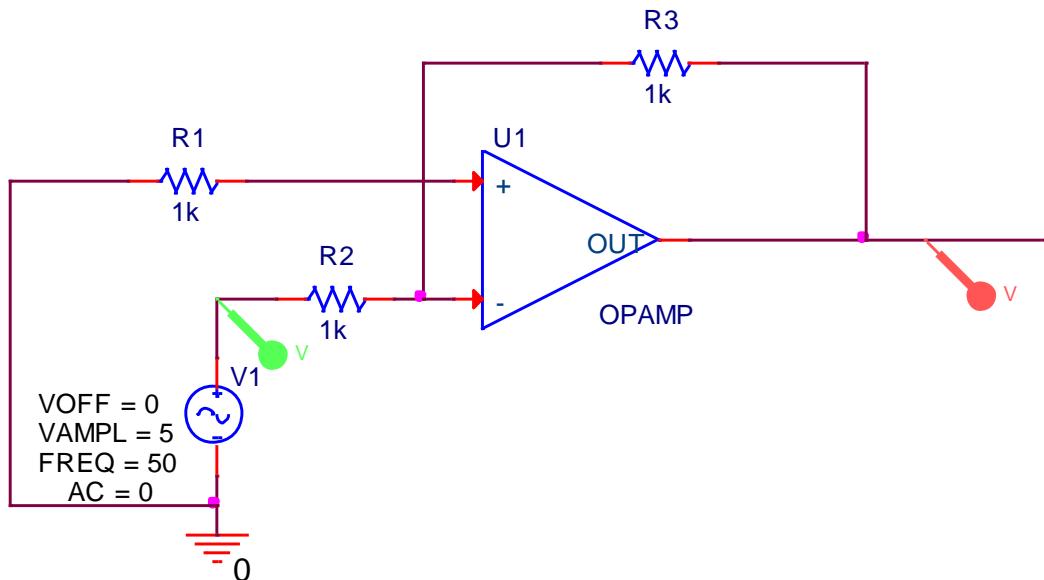
# OUTPUT



## CIRCUIT :



## NON INVERTING MODE



## INVERTING MODE

## **NET LIST**

### **// NON INVERTING MODE**

\* source TRANS AS AMP

V\_V1 N00130 0 AC 0

+SIN 0 5 50 0 0 0

E\_U1 N00160 0 VALUE {LIMIT(V(N00130,N00656)\*1E6,-15V,+15V)}

R\_R1 N00656 N00160 1k TC=0,0

R\_R2 0 N00656 1k TC=0,0

### **//INVERTING MODE**

\* source INV AMP

E\_U1 N00185 0 VALUE {LIMIT(V(N00155,N00159)\*1E6,-15V,+15V)}

R\_R1 0 N00155 1k TC=0,0

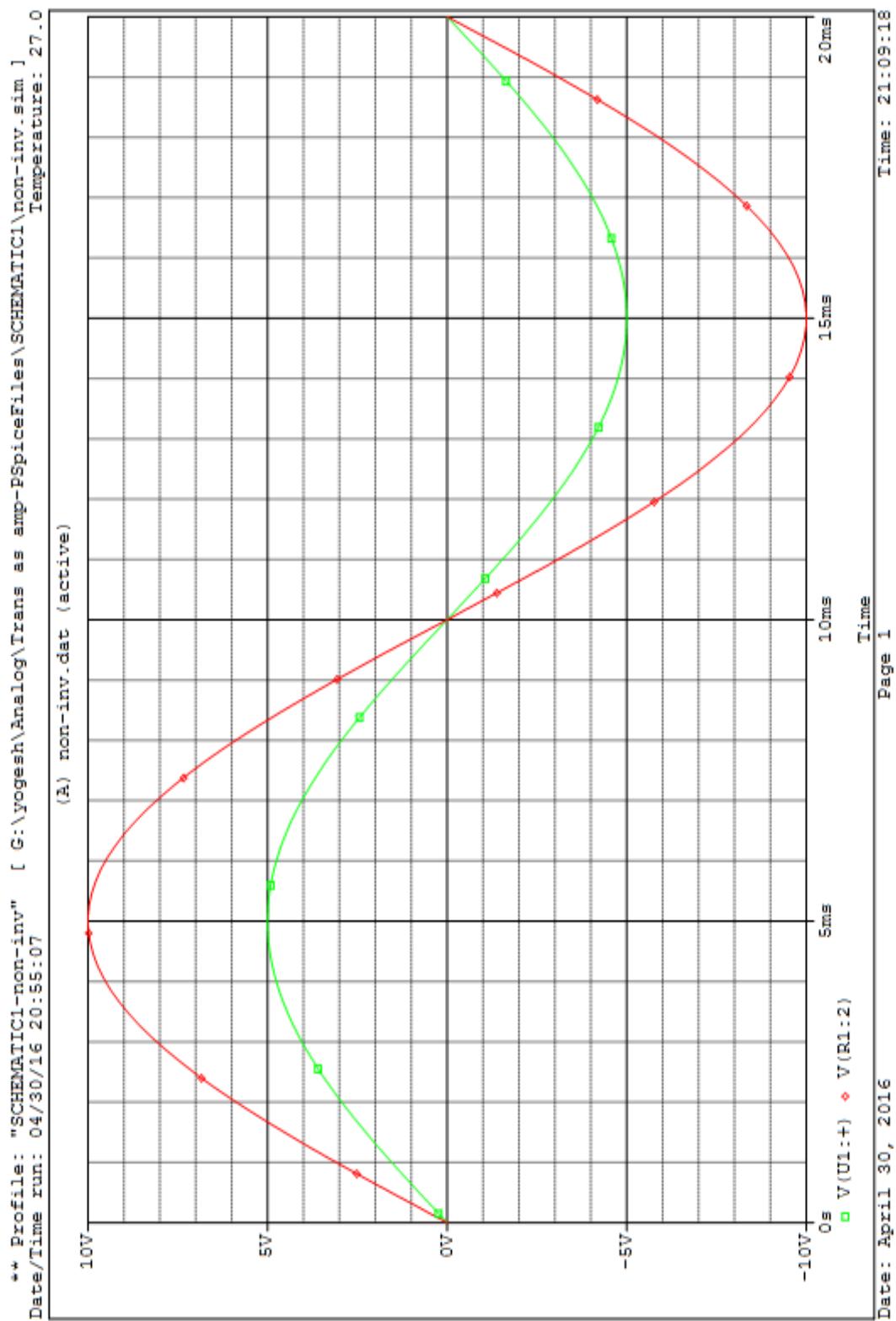
R\_R2 N00163 N00159 1k TC=0,0

R\_R3 N00159 N00185 1k TC=0,0

V\_V1 N00163 0 AC 0

+SIN 0 5 50 0 0 0

## GRAPH:



## **CONCLUSION :**

Thus from this experiment we got to know about op amp and its characteristics in inverting and non inverting mode.

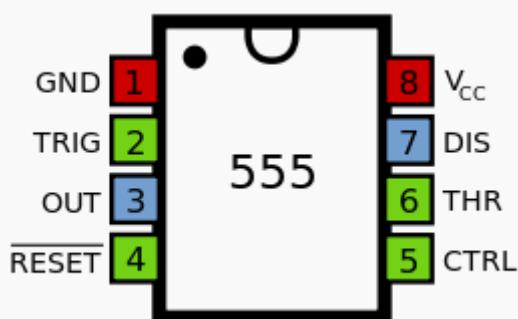
## PRACTICAL: 12

**AIM:** To Simulate IC 555 Timer Circuit using Or-Cad Capture 16.3

**SOFTWARE:** Orcad capture CIS 16.3

### THEORY:

1. The **555 timer IC** is an integrated circuit (chip) used in a variety of timer, pulse generation, and oscillator applications. The 555 can be used to provide time delays, as an oscillator, and as a flip-flop element. Derivatives provide up to four timing circuits in one package.
2. Introduced in 1971 by American company Signetics, the 555 is still in widespread use due to its low price, ease of use, and stability. It is now made by many companies in the original bipolar and also in low-power CMOS types. As of 2003, it was estimated that 1 billion units are manufactured every year.

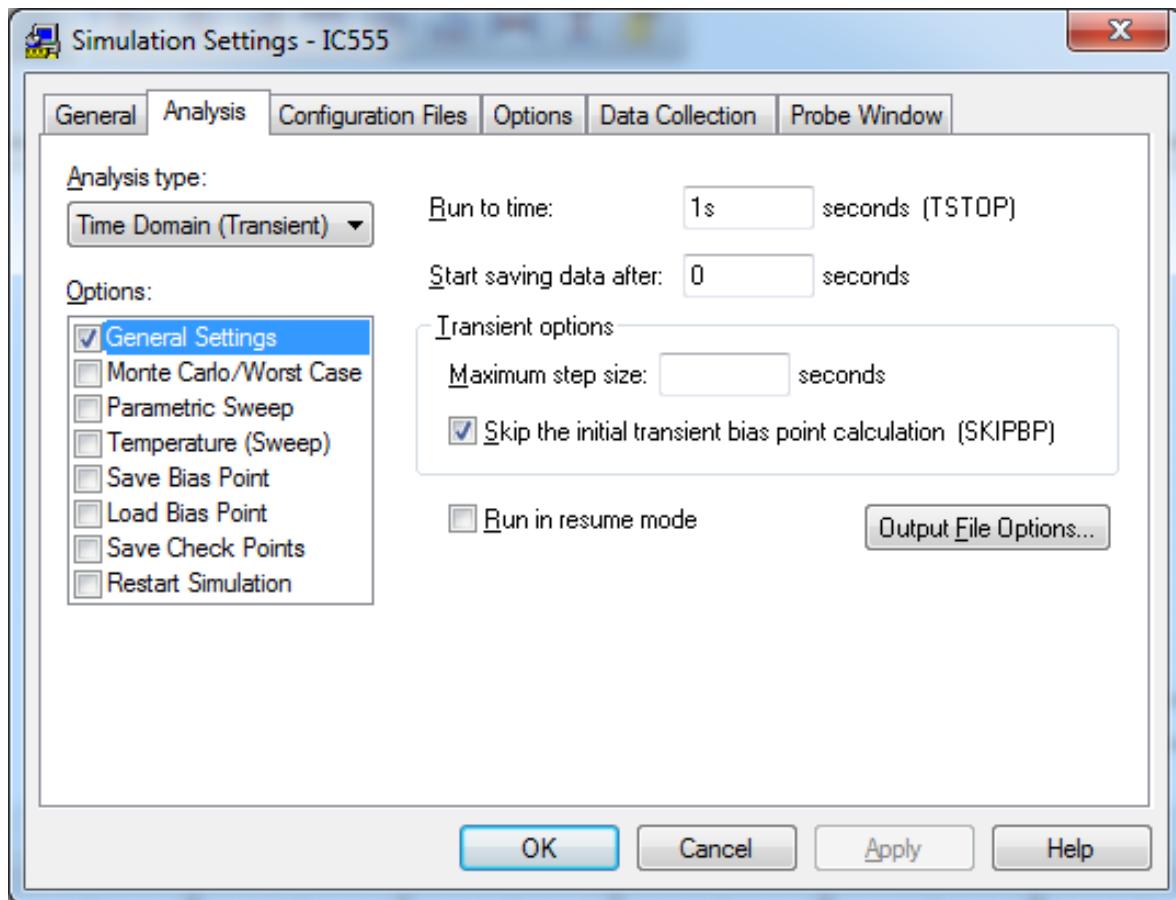


Pinout diagram

3. The connection of the pins for a DIP package is as follows:
4. Pin 5 is also sometimes called the CONTROL VOLTAGE pin. By applying a voltage to the CONTROL VOLTAGE input one can alter the timing characteristics of the device. In most applications, the CONTROL VOLTAGE input is not used. It is usual to connect a 10 nF capacitor between pin 5 and 0 V to prevent interference. The CONTROL VOLTAGE input can be used to build an astable multivibrator with a frequency-modulated output.

## **PROCEDURE:**

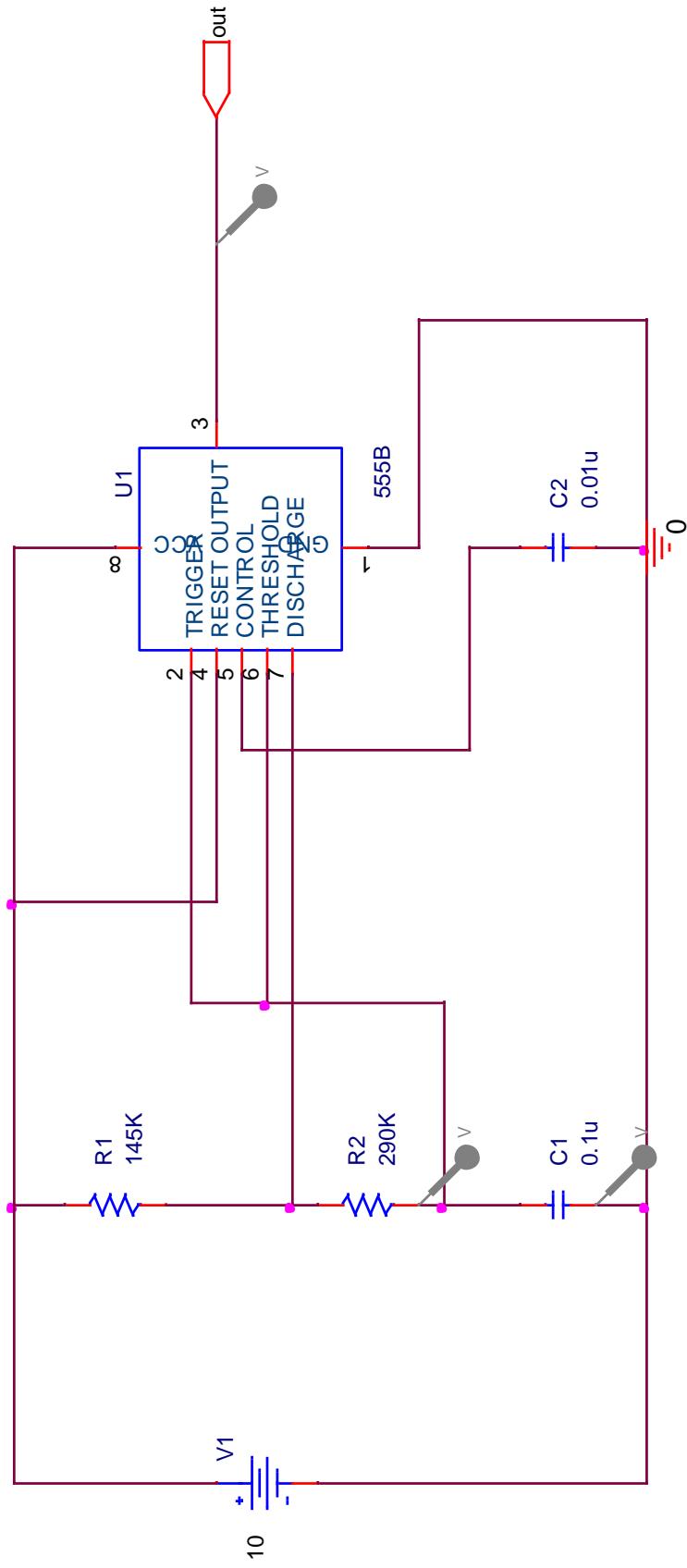
- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Now run the project using Pspice.
- Then save the practical.



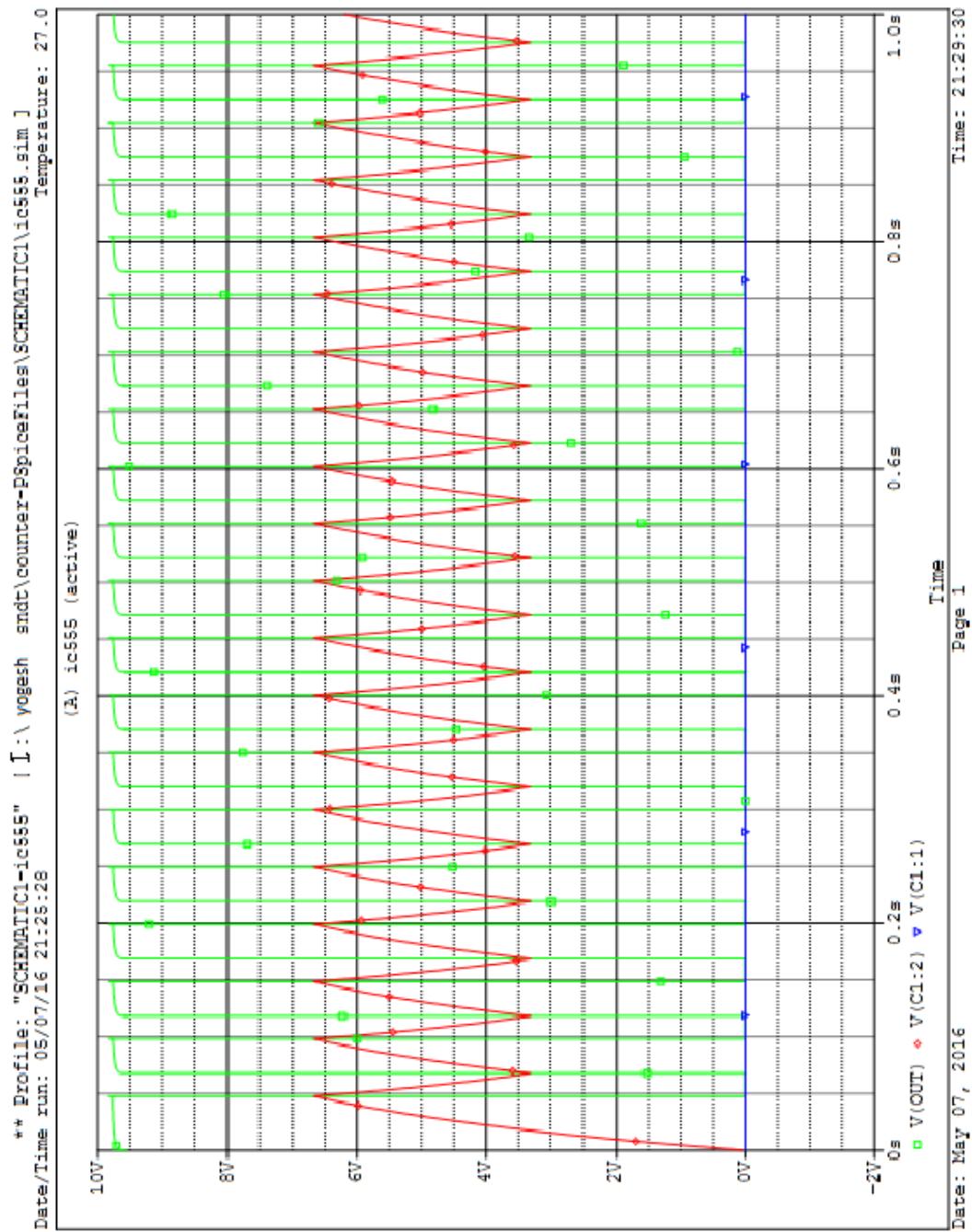
## NETLIST

```
* source IC 555
.EXTERNAL OUTPUT out
X_U1      0 N00191 OUT N00207 N00300 N00191 N00187 N00207 555B
V_V1      N00207 0 10
R_R1      N00187 N00207 145K TC=0,0
R_R2      N00191 N00187 290K TC=0,0
C_C1      0 N00191 0.1u TC=0,0
C_C2      0 N00300 0.01u TC=0,0
```

## CIRCUIT:



## OUTPUT



## CONCLUSION

Thus we got to know about the 555 timer ic .

## PRACTICAL: 13

**AIM:** To study all the basic digital logic gates.

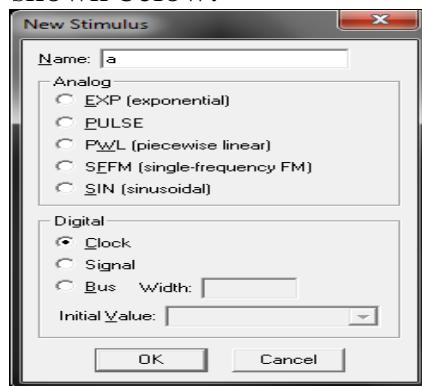
**SOFTWARE:** OrCAD capture CIS 16.3

### PROCEDURE:

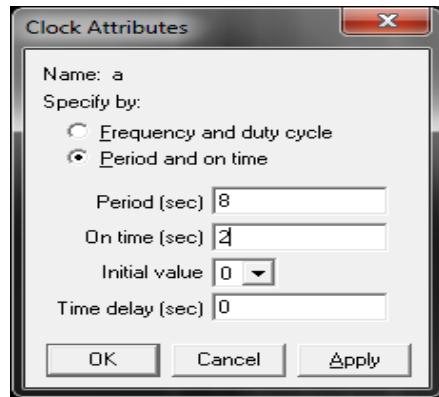
- Firstly go to start menu & open program & then go to cadence & from that open OrCAD capture.
- In OrCAD go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- Now, here we have to implement the logic gates, so we have to give two inputs, for that we have to go to stimulus editor & the path is given below:

○ Start menu → all programs → Cadence → Release 16.3 → Pspice accessories → Stimulus editor

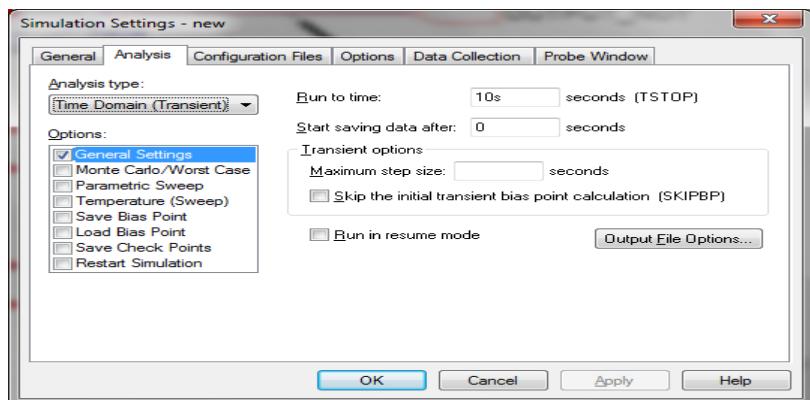
- For inputs ,you have to select new file(stimulus → new), a dialog will appear as shown below:



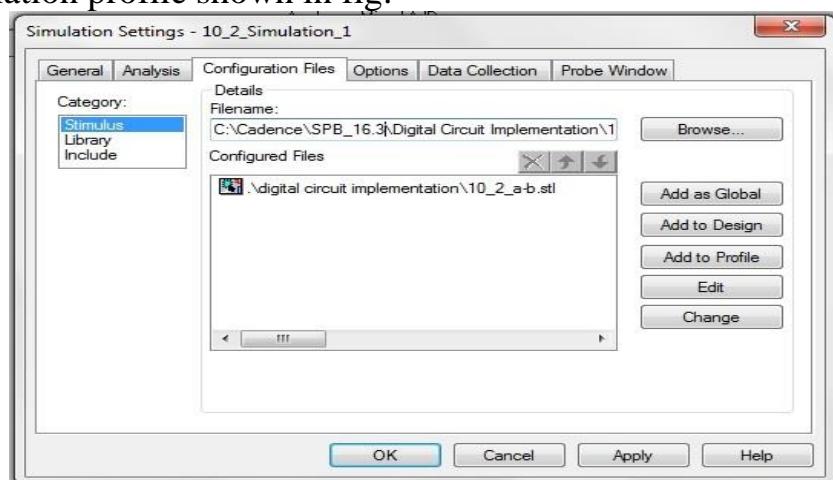
- Press ok & now again another dialog box will appear as shown in fig:



- Now, same do for b, & save this.
- On completing the schematic diagram of the circuit, move cursor to the 'Pspice' option given in the toolbar and select the 'New stimulation profile'(For implementation of the circuit)



- We have to add inputs in project. How to add that shown in stimulation profile shown in fig.

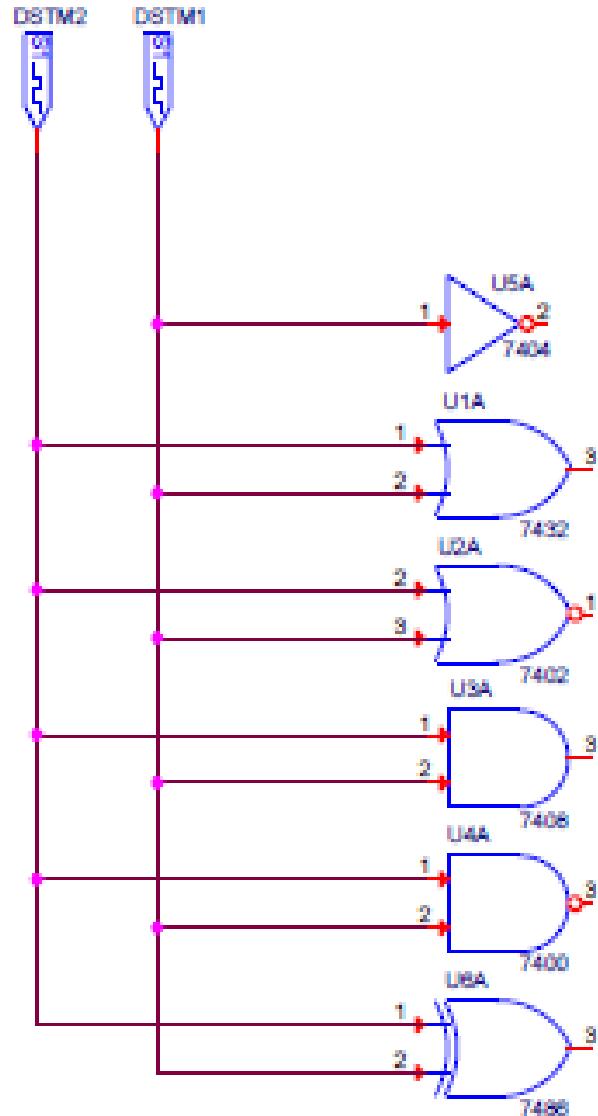


- It will display the dialog box for editing the stimulus circuit, complete the box & click ok.
- Now, stimulate the circuit by selecting run button.

- It will display the window showing the output graph of the respective simulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Now run the project using Pspice and note the waveforms for all logic gates.
- Then, save the practicals.

## CIRCUIT

---

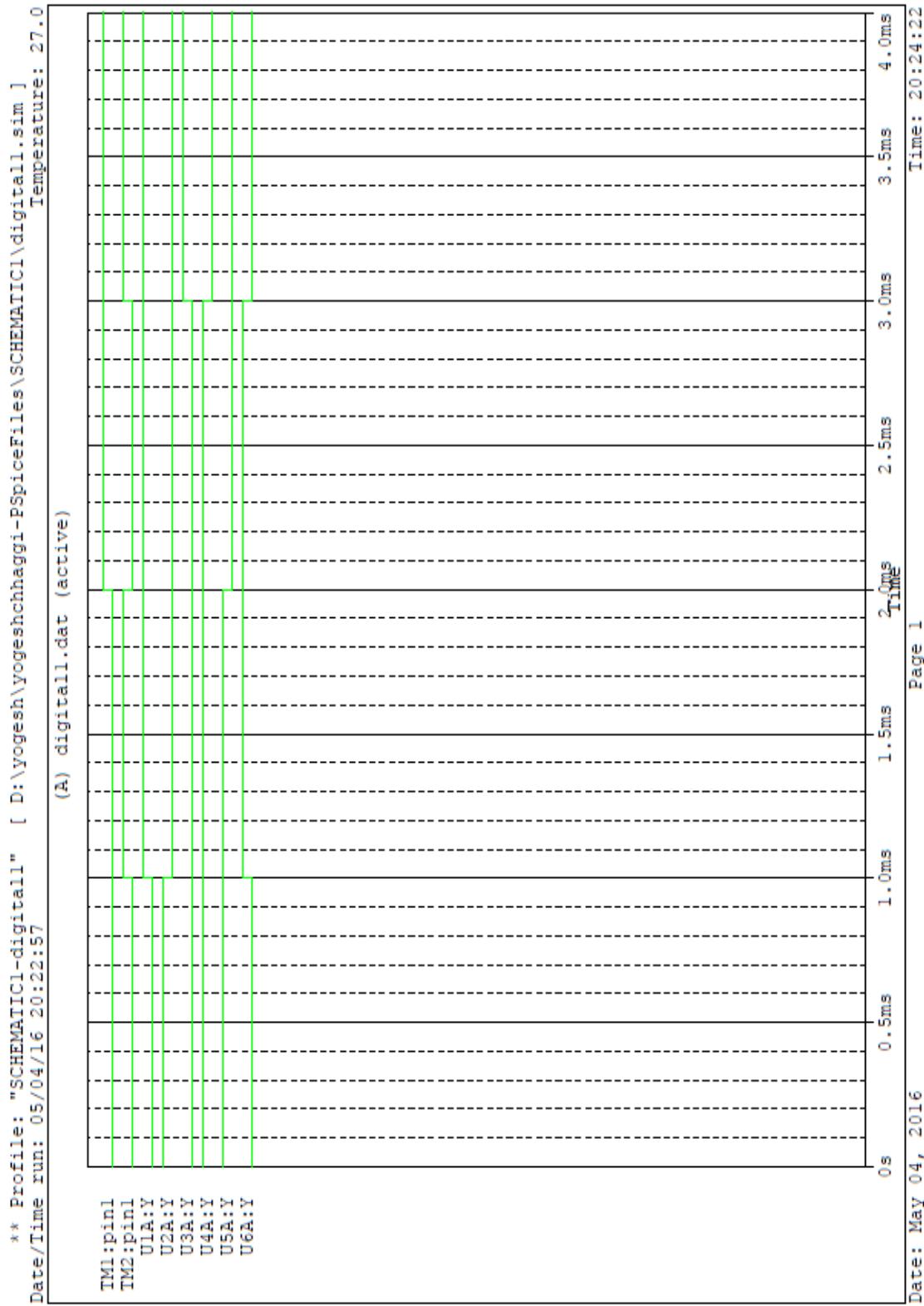


## LOGIC GATES

## NETLIST :

```
* source LOGIC GATES
.EXTERNAL OUTPUT AND
.EXTERNAL OUTPUT OR
.EXTERNAL OUTPUT NOT
.EXTERNAL OUTPUT NAND
.EXTERNAL OUTPUT NOR
.EXTERNAL OUTPUT XOR
.EXTERNAL OUTPUT XNOR
X_U1A      N00523 N00625 AND $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2A      N00523 N00625 OR $G_DPWR $G_DGND 7432 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3A      N00523 NOT $G_DPWR $G_DGND 7404 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U4A      N00523 N00625 NAND $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U5A      N00523 N00625 NOR $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U6A      N00523 N00625 XOR $G_DPWR $G_DGND 7486 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U7A      N00523 N00625 XNOR $G_DPWR $G_DGND 74LS266
PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1    STIM(1,0) $G_DPWR $G_DGND N00523 IO_STM
STIMULUS=A
U_DSTM2    STIM(1,0) $G_DPWR $G_DGND N00625 IO_STM
STIMULUS=B
+ IO_LEVEL=0 MNTYMXDLY=0
```

# OUTPUT



## **CONCLUSION :**

After completion of this practical I able to understand the working of various logic gates using OrCAD software.

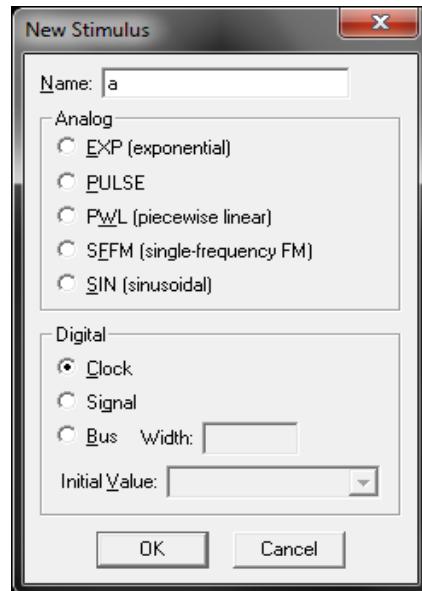
## PRACTICAL: 14

**AIM:** To study about NAND as universal logic gate.

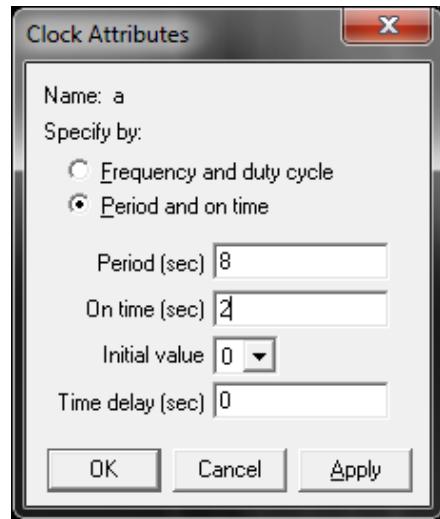
**SOFTWARE:** OrCAD capture CIS 16.3

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open OrCAD capture.
- In OrCAD go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- Now, here we have to implement the logic gates, so we have to give two inputs, for that we have to go to stimulus editor & the path is given below:
  - a. Start menu → all programs → Cadence → Release 16.3 → Pspice accessories → Stimulus editor
- For inputs ,you have to select new file(stimulus → new), a dialog will appear as shown below:

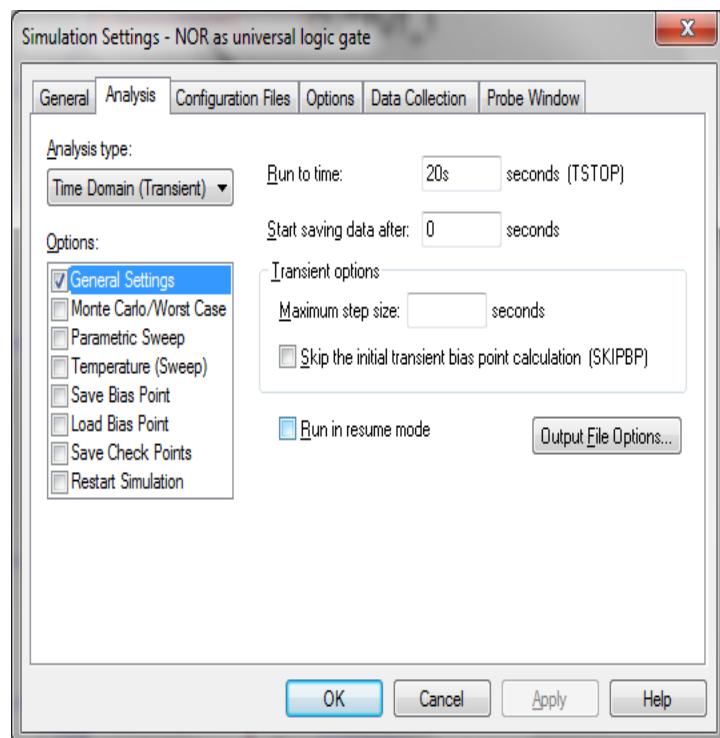


- Press ok & now again another dialog box will appear as shown in fig:



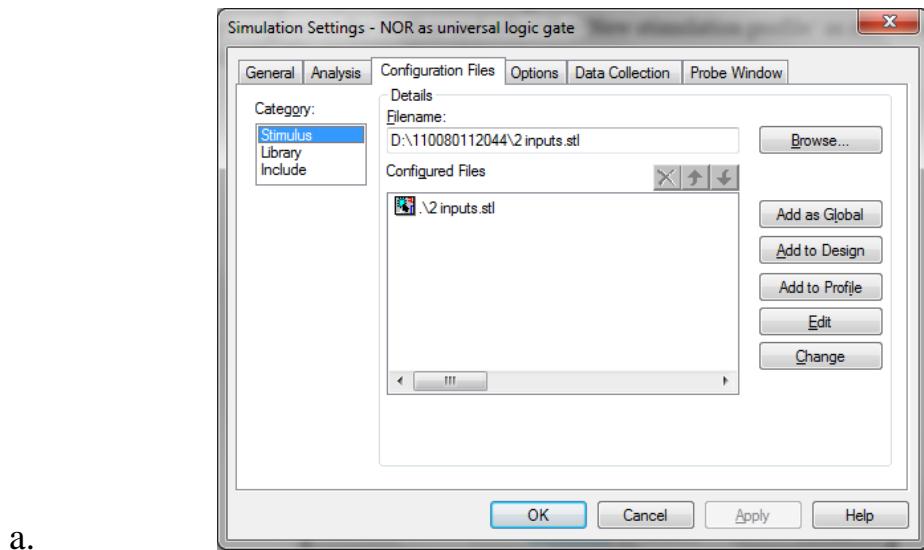
a.

- Now, same do for b, & save this.
- On completing the schematic diagram of the circuit, move cursor to the 'Pspice' option given in the toolbar and select the 'New stimulation profile'.(For implementation of the circuit)



a.

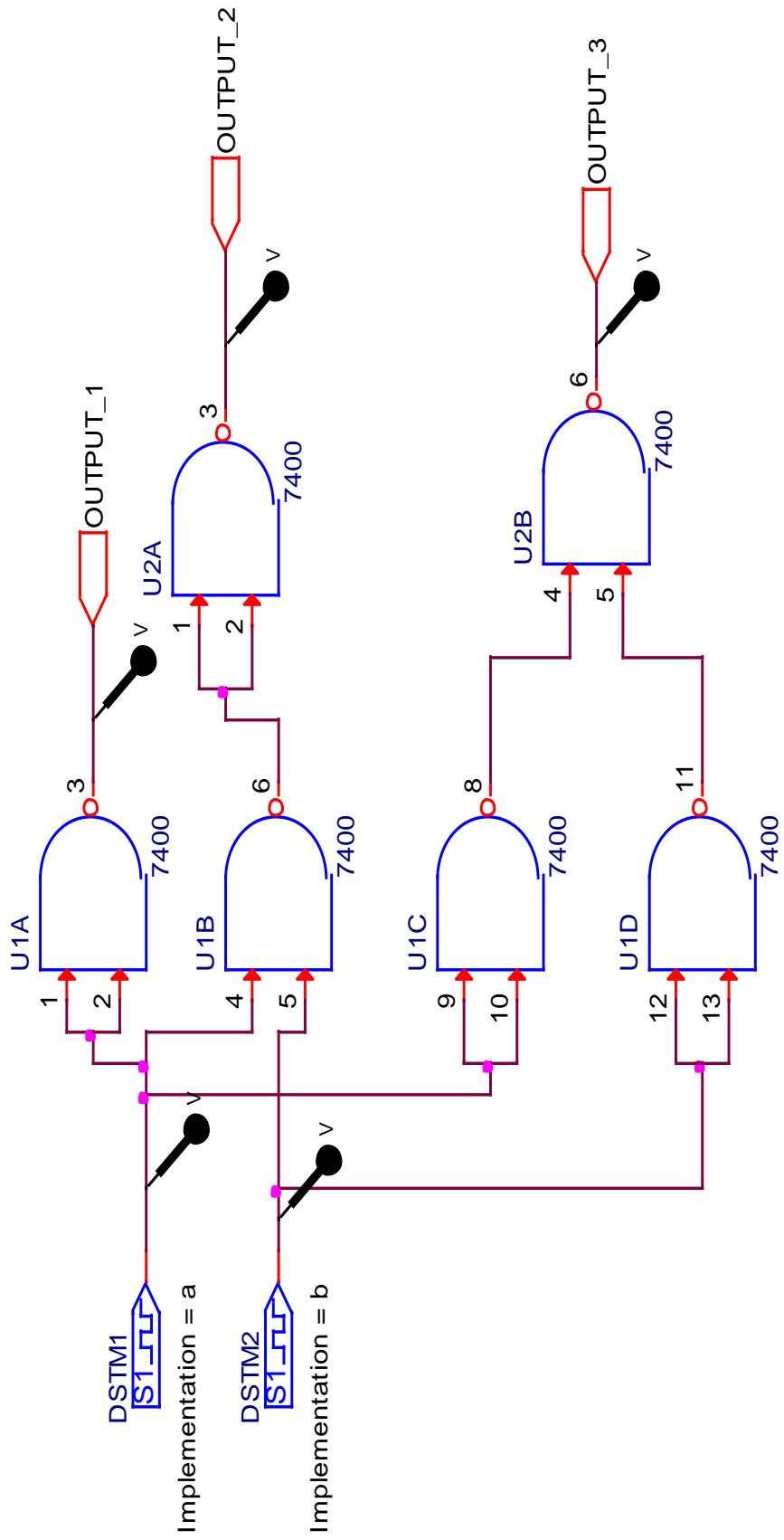
- We have to add inputs in project. How to add that shown in stimulation profile shown in fig.



a.

- It will display the dialog box for editing the stimulus circuit, complete the box & click ok.
- Now, stimulate the circuit by selecting run button.
- It will display the window showing the output graph of the respective simulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Now run the project using Pspice and note the waveforms for all logic gates.
- Then, save the practicals.

# CIRCUIT

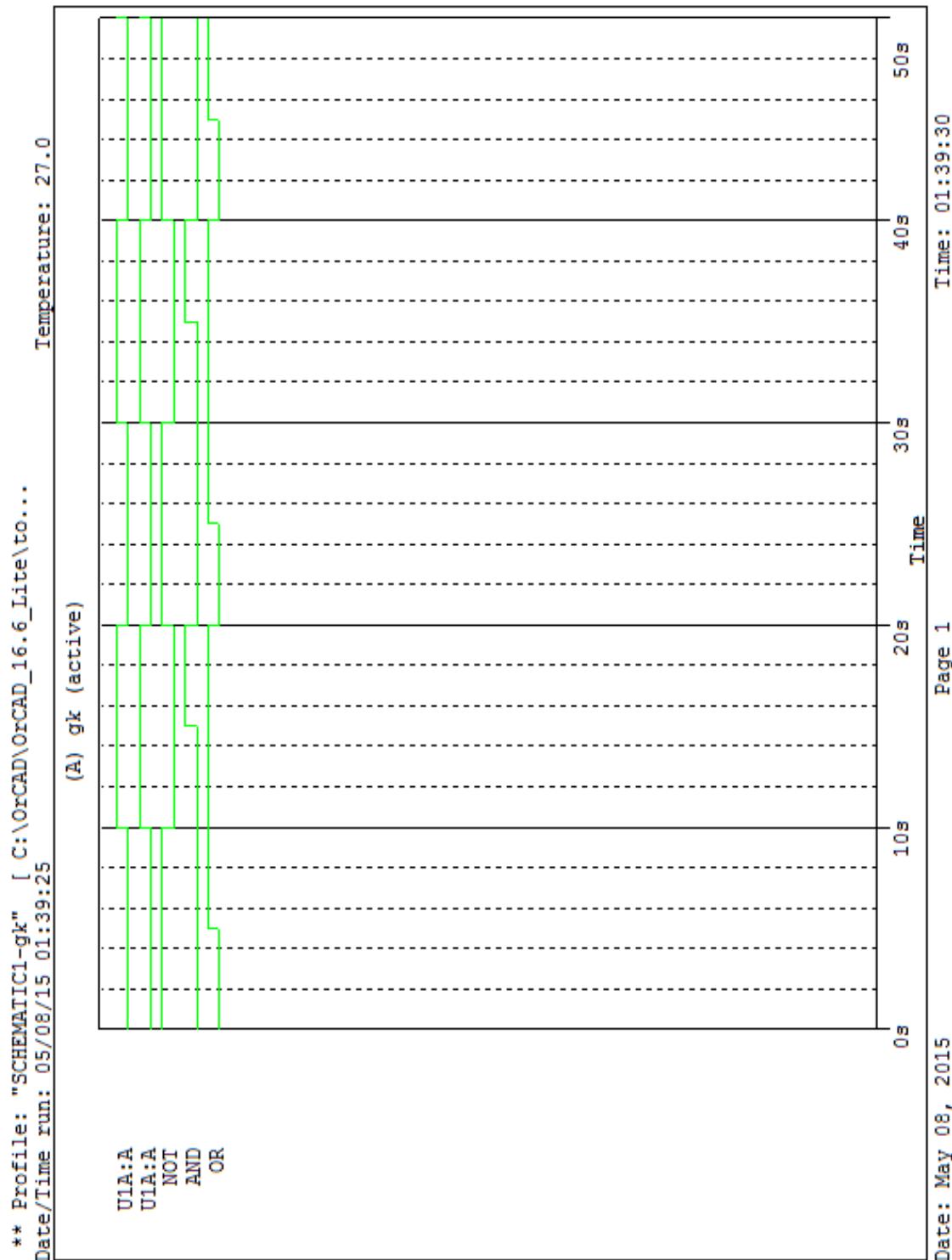


**NAND as universal gate**

## NET LIST

```
* source NAND
.EXTERNAL OUTPUT not
.EXTERNAL OUTPUT and
.EXTERNAL OUTPUT or
X_U1A    N00203 N00203 NOT $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2B    N00643 N00643 AND $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U1B    N00203 N00629 N00643 $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2C    N00203 N00203 N00908 $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3C    N00908 N00918 OR $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3B    N00629 N00629 N00918 $G_DPWR $G_DGND 7400 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1   STIM(1,0) $G_DPWR $G_DGND N00203 IO_STM
STIMULUS=a
U_DSTM2   STIM(1,0) $G_DPWR $G_DGND N00629 IO_STM
STIMULUS=b
=0 MNTYMXDLY=0
```

# OUTPUT



## **CONCLUSION:**

By performing this practical, we concluded we can study about the NOR gate and NAND gate which can be used as AND, OR and NOT gates with the help of OrCAD software.

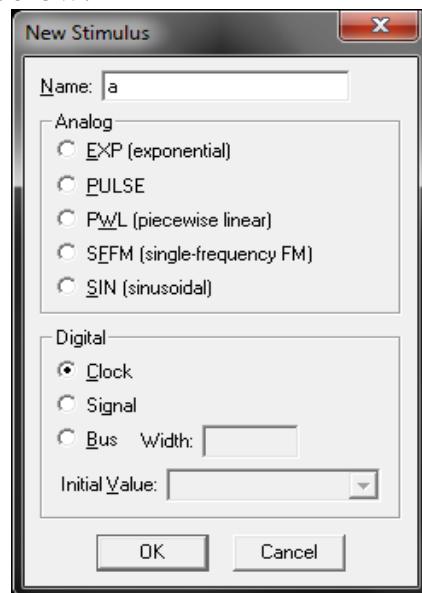
## PRACTICAL: 15

**AIM:** To study about NOR as universal logic gate.

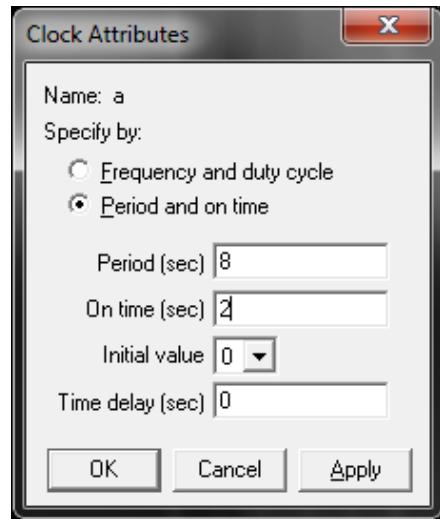
**SOFTWARE:** OrCAD capture CIS 16.3

### PROCEDURE:

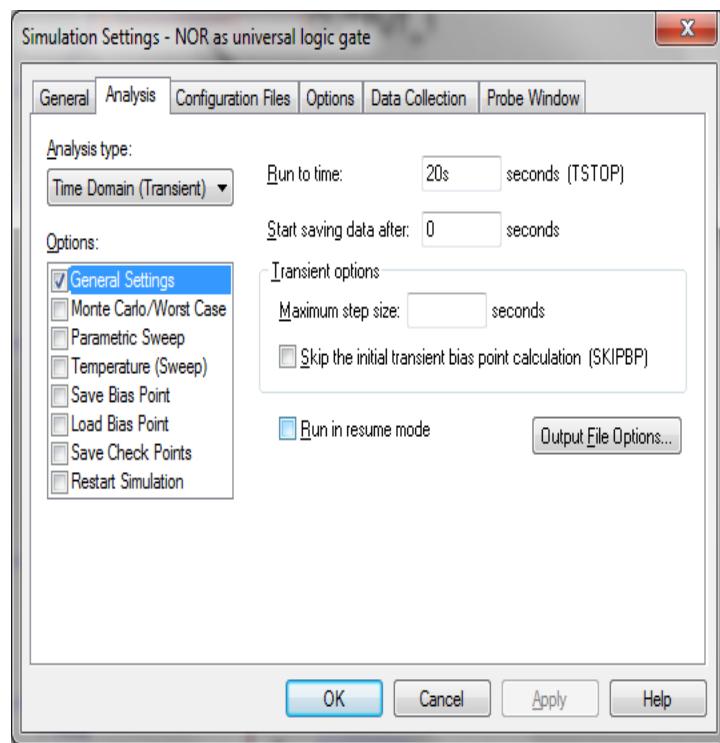
- Firstly go to start menu & open program & then go to cadence & from that open OrCAD capture.
- In OrCAD go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- Now, here we have to implement the logic gates, so we have to give two inputs, for that we have to go to stimulus editor & the path is given below:
  - Start menu → all programs → Cadence → Release 16.3 → Pspice accessories → Stimulus editor
- For inputs ,you have to select new file(stimulus → new), a dialog will appear as shown below:



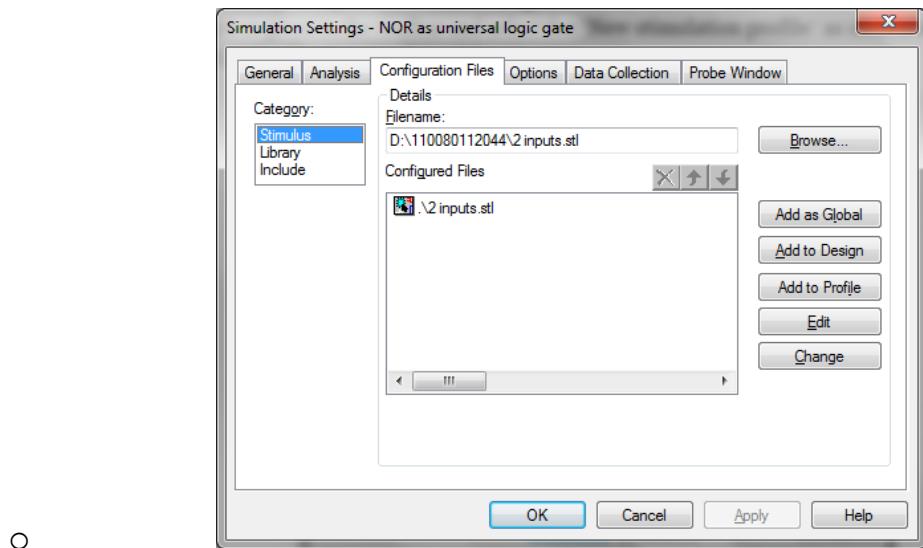
- Press ok & now again another dialog box will appear as shown in fig:



- Now, same do for b, & save this.
- On completing the schematic diagram of the circuit, move cursor to the 'Pspice' option given in the toolbar and select the 'New stimulation profile'.(For implementation of the circuit)

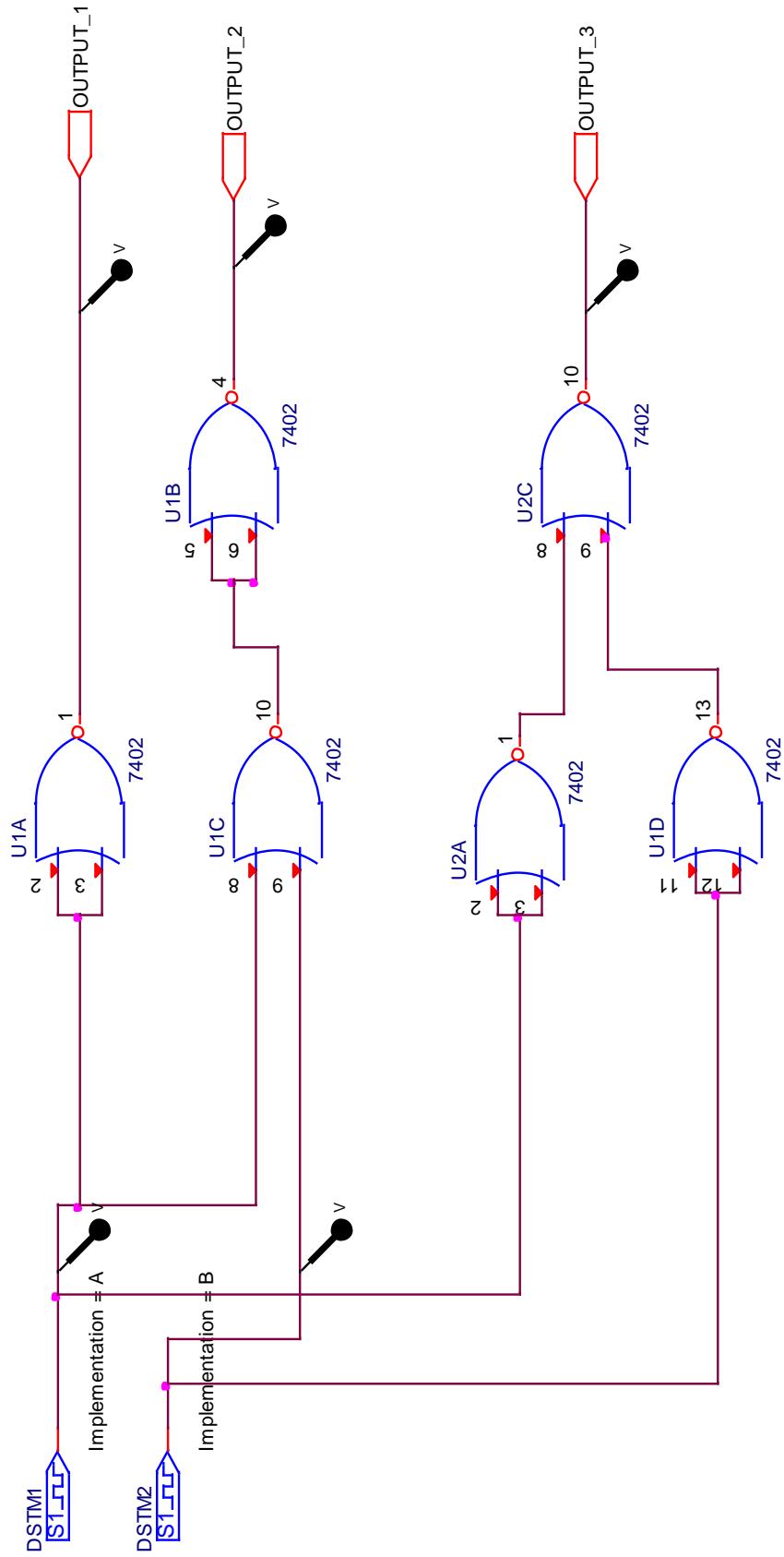


- We have to add inputs in project. How to add that shown in stimulation profile shown in fig.



- It will display the dialog box for editing the stimulus circuit, complete the box & click ok.
- Now, stimulate the circuit by selecting run button.
- It will display the window showing the output graph of the respective simulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Now run the project using Pspice and note the waveforms for all logic gates.
- Then, save the practicals.

# CIRCUIT

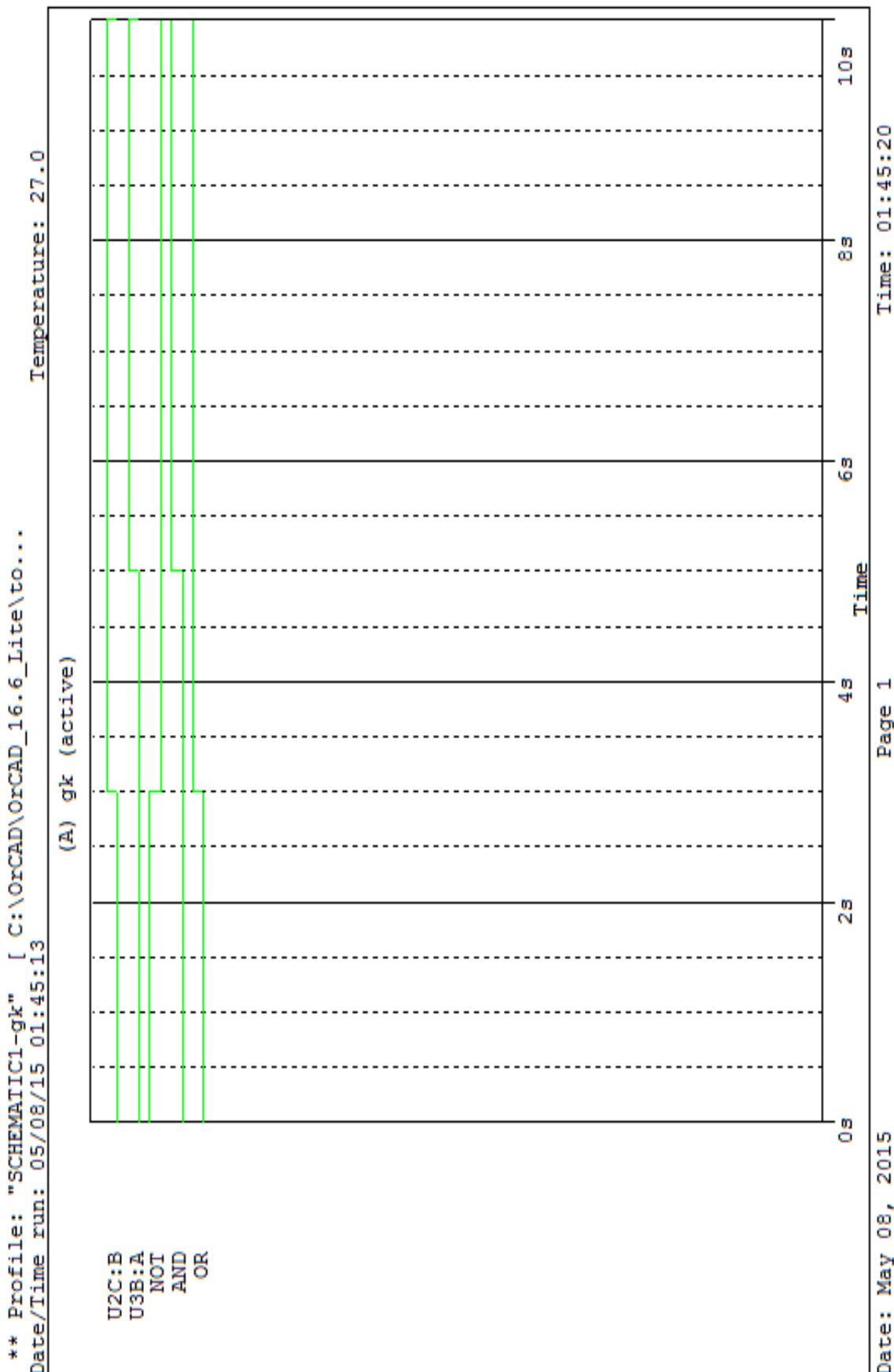


**NOR as universal gate**

## NET LIST

```
* source NOR
.EXTERNAL OUTPUT NOT
.EXTERNAL OUTPUT and
.EXTERNAL OUTPUT OR
X_U1A    N00197 N00197 NOT $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2C    N00432 N00464 AND $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2B    N00486 N00486 N00464 $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U1B    N00197 N00197 N00432 $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3B    N00707 N00707 OR $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2D    N00197 N00486 N00707 $G_DPWR $G_DGND 7402 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1   STIM(1,0) $G_DPWR $G_DGND N00197 IO_STM
STIMULUS=a
U_DSTM2   STIM(1,0) $G_DPWR $G_DGND N00486 IO_STM
STIMULUS=b
```

## OUTPUT



## **CONCLUSION:**

By performing this practical, we concluded we can study about the NOR gate and NAND gate which can be used as AND, OR and NOT gates with the help of OrCAD software.

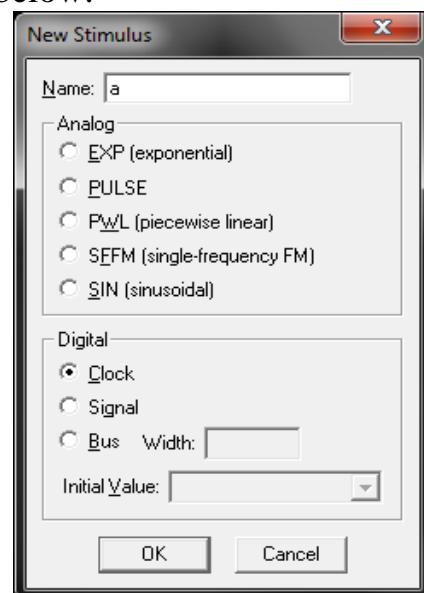
## PRACTICAL: 16

**AIM:** To study about Half adder

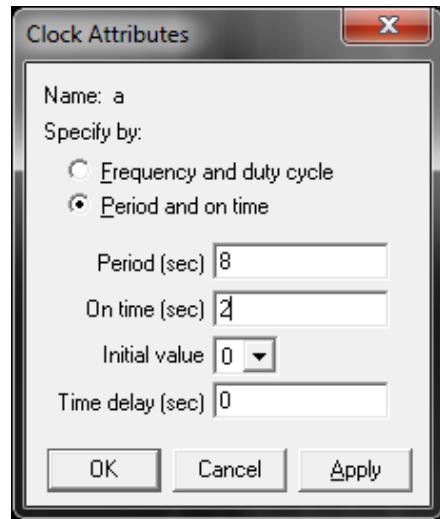
**SOFTWARE:** OrCAD capture CIS 16.3

### PROCEDURE:

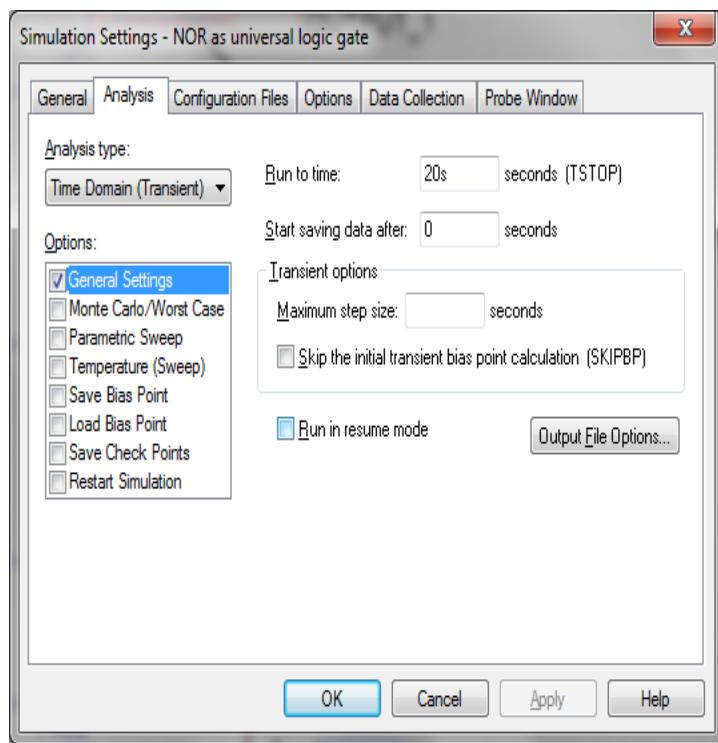
- Firstly go to start menu & open program & then go to cadence & from that open OrCAD capture.
- In OrCAD go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- Now, here we have to implement the logic gates, so we have to give two inputs, for that we have to go to stimulus editor & the path is given below:
  - Start menu → all programs → Cadence → Release 16.3 → Pspice accessories → Stimulus editor
- For inputs ,you have to select new file(stimulus → new), a dialog will appear as shown below:



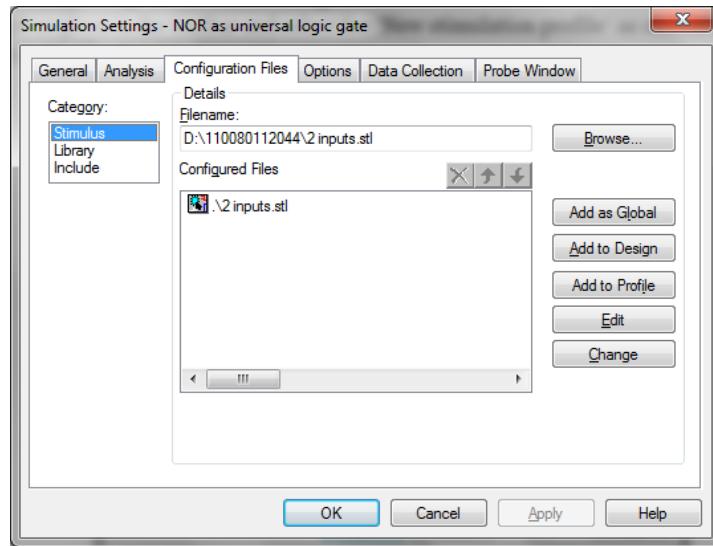
- Press ok & now again another dialog box will appear as shown in fig:



- Now, same do for b and c, & save this.
- On completing the schematic diagram of the circuit, move cursor to the 'Pspice' option given in the toolbar and select the 'New stimulation profile'.(For implementation of the circuit)

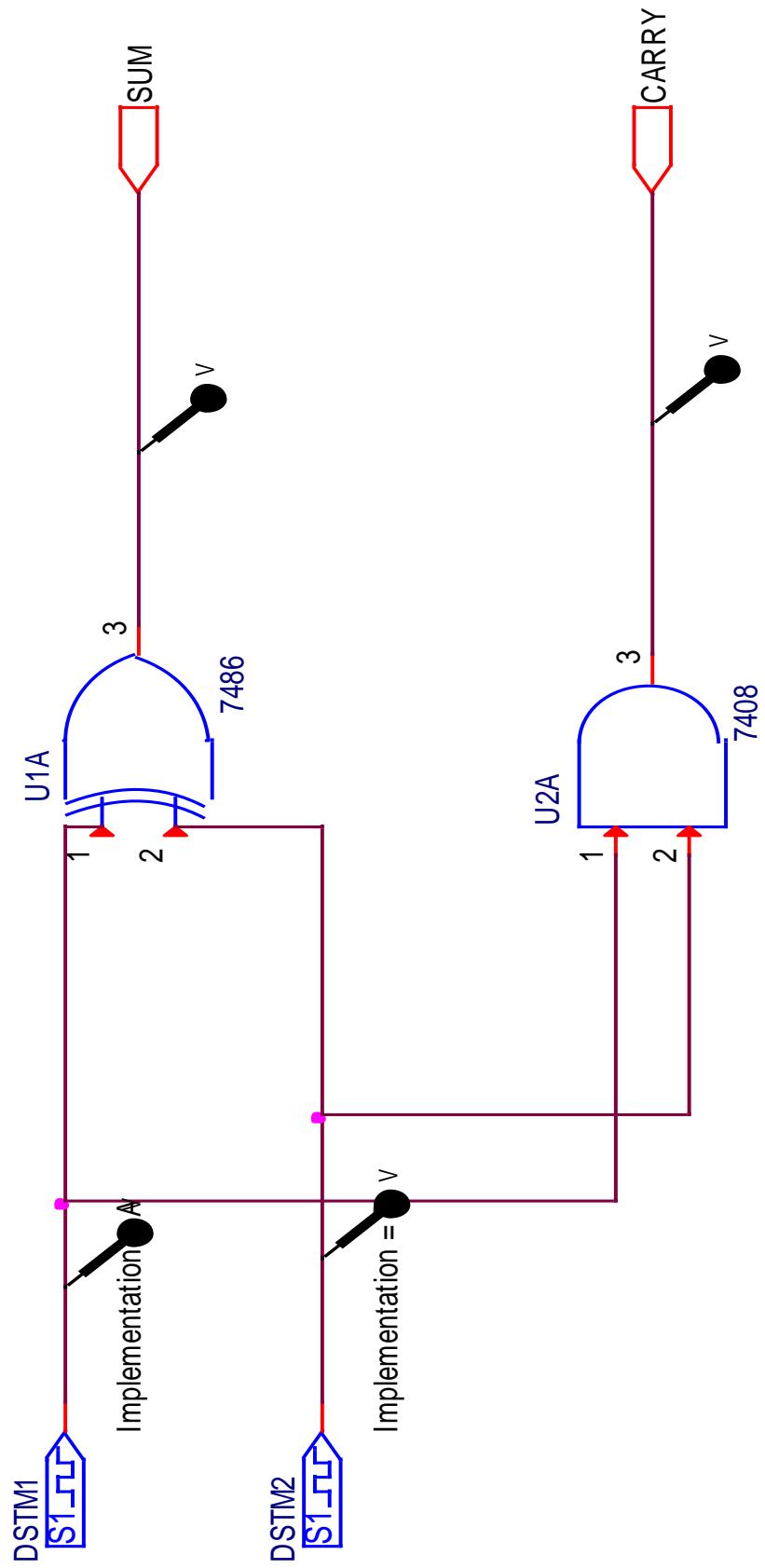


- We have to add inputs in project. How to add that shown in stimulation profile shown in fig.



- It will display the dialog box for editing the stimulus circuit, complete the box & click ok.
- Now, stimulate the circuit by selecting run button.
- It will display the window showing the output graph of the respective simulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Now run the project using Pspice and note the waveforms for all logic gates.
- Then, save the practicals.

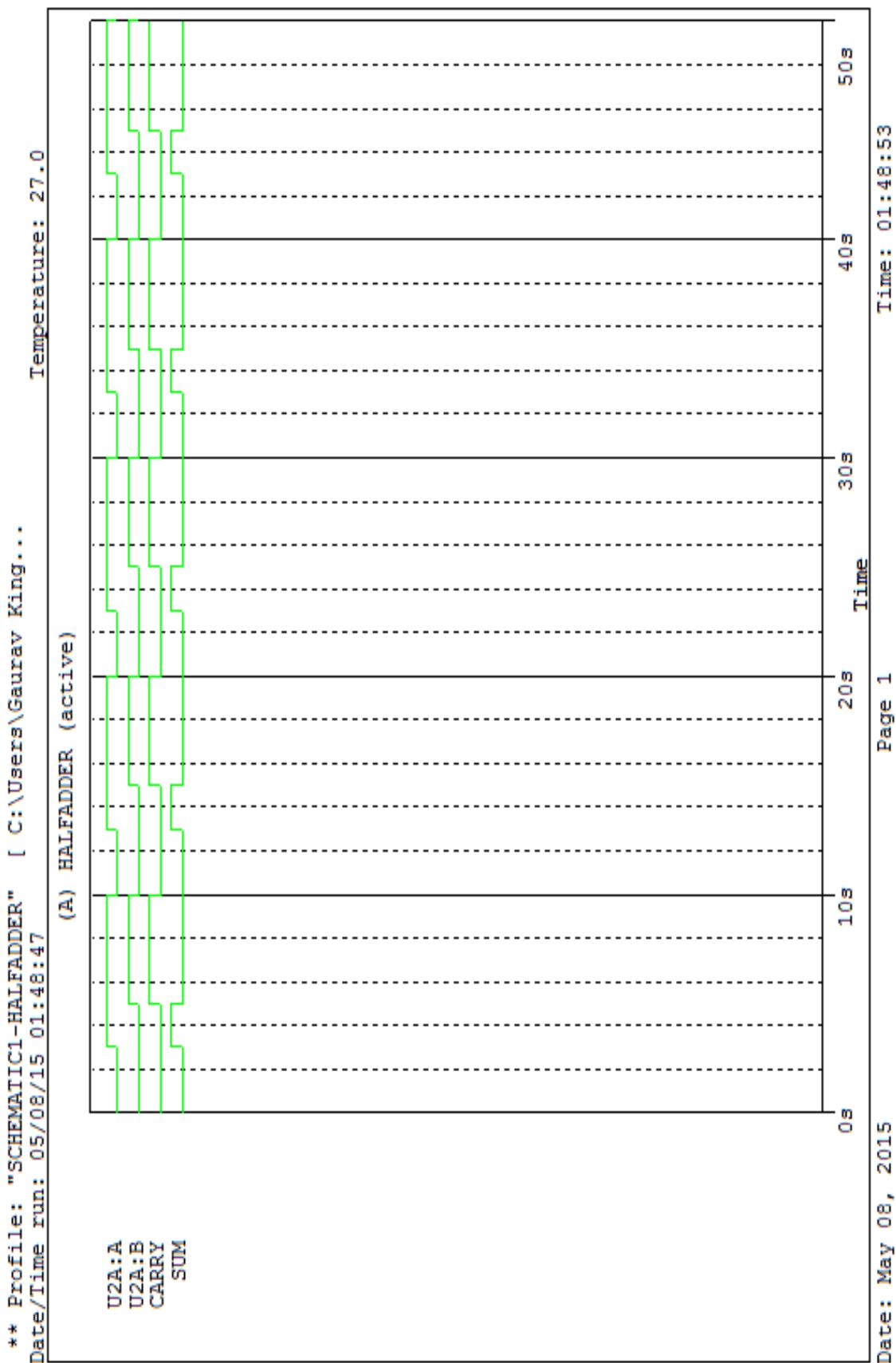
# CIRCUIT



## NET LIST

```
* source HALF ADDER
.EXTERNAL OUTPUT CARRY
.EXTERNAL OUTPUT SUM
X_U1A      N00196 N00206 CARRY $G_DPWR $G_DGND 7408
PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2A      N00196 N00206 SUM $G_DPWR $G_DGND 7486 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1    STIM(1,0) $G_DPWR $G_DGND N00196 IO_STM
STIMULUS=a
U_DSTM2    STIM(1,0) $G_DPWR $G_DGND N00206 IO_STM
STIMULUS=b
```

## OUTPUT



## **CONCLUSION:**

By performing this practical, we conclude our study of half adder using OrCAD software.

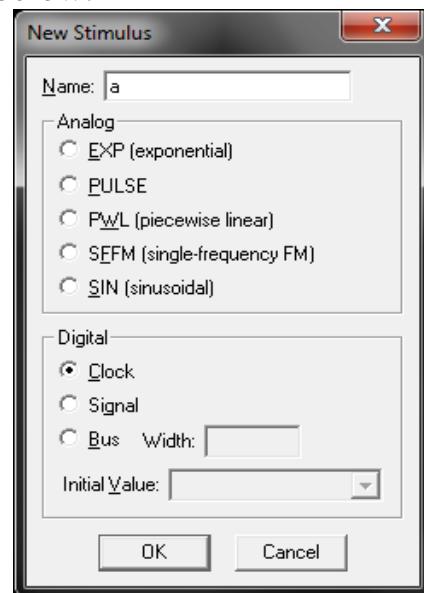
## PRACTICAL: 17

**AIM:** To study about Full adder.

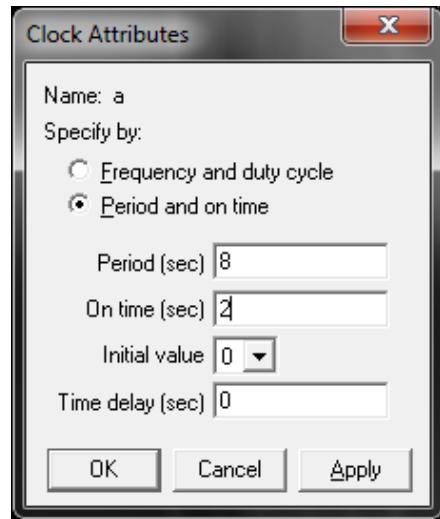
**SOFTWARE:** OrCAD capture CIS 16.3

### PROCEDURE:

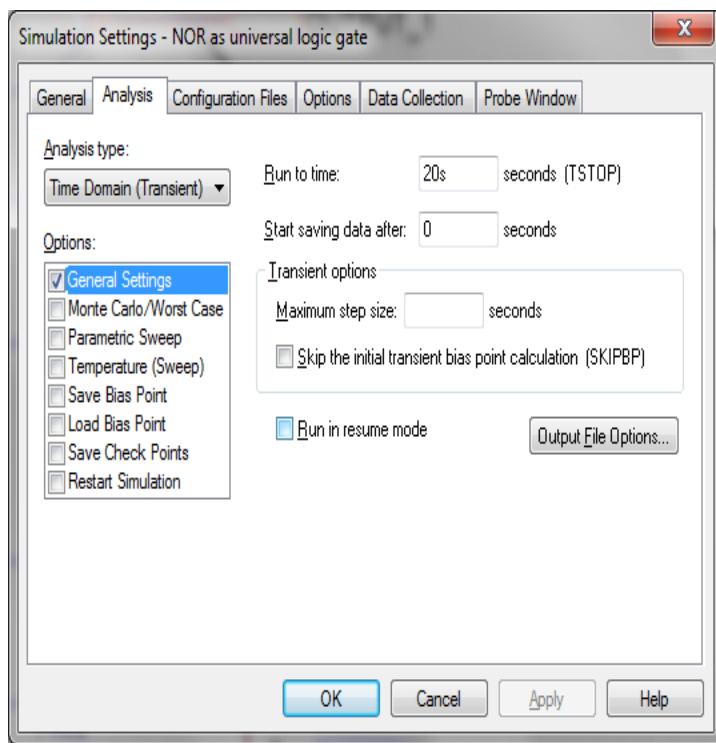
- Firstly go to start menu & open program & then go to cadence & from that open OrCAD capture.
- In OrCAD go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- Now, here we have to implement the logic gates, so we have to give two inputs, for that we have to go to stimulus editor & the path is given below:
  - Start menu → all programs → Cadence → Release 16.3 → Pspice accessories → Stimulus editor
- For inputs ,you have to select new file(stimulus → new), a dialog will appear as shown below:



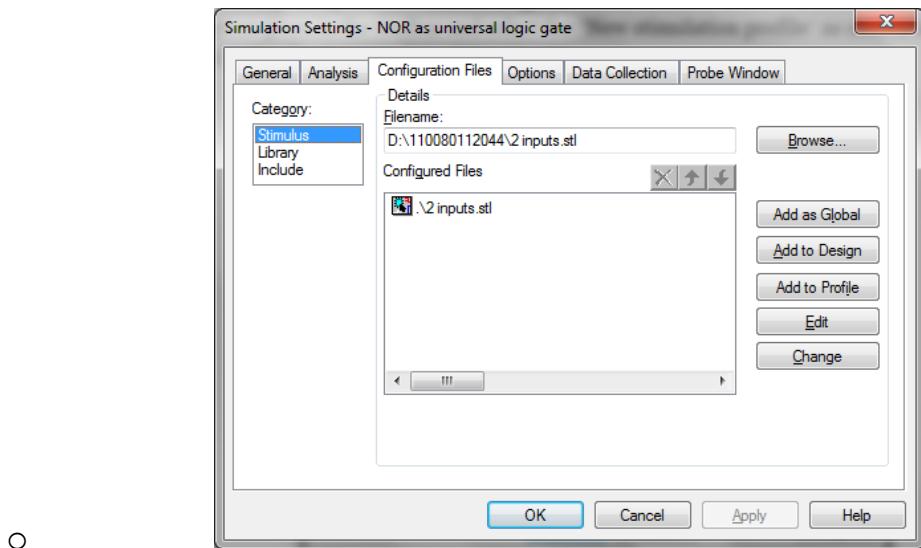
- Press ok & now again another dialog box will appear as shown in fig:



- Now, same do for b and c, & save this.
- On completing the schematic diagram of the circuit, move cursor to the 'Pspice' option given in the toolbar and select the 'New stimulation profile'.(For implementation of the circuit)

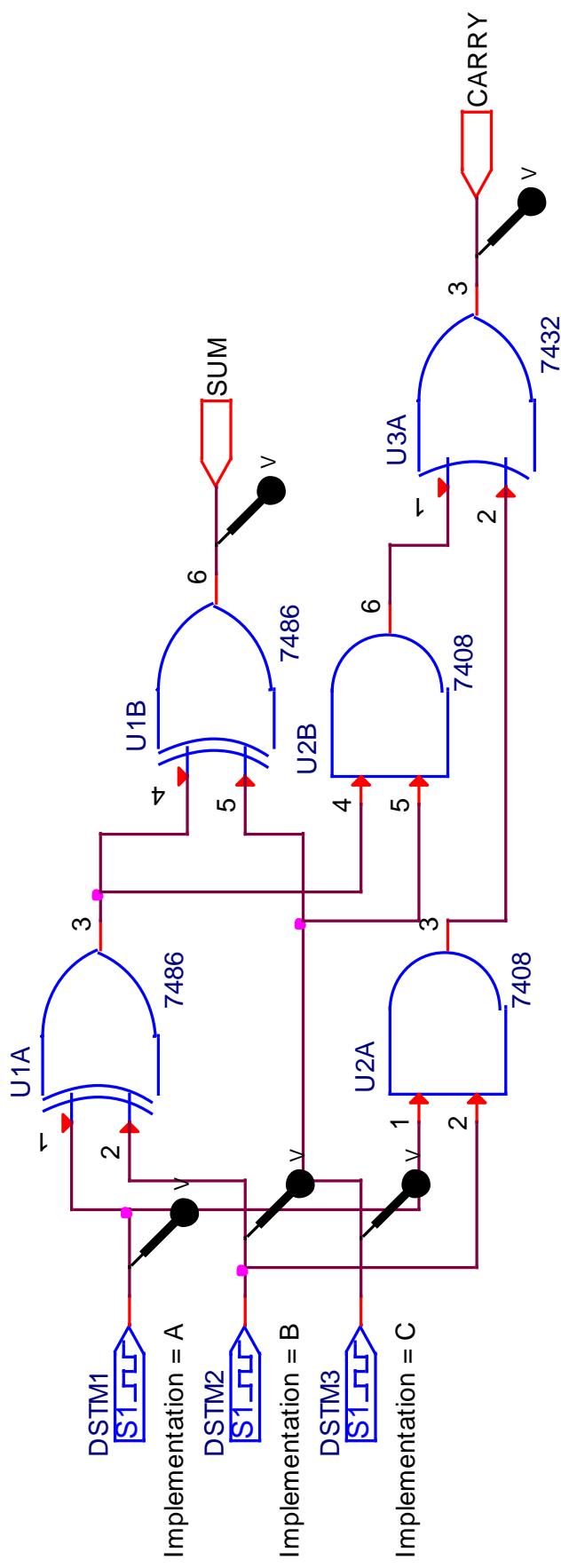


- We have to add inputs in project. How to add that shown in stimulation profile shown in fig.



- It will display the dialog box for editing the stimulus circuit, complete the box & click ok.
- Now, stimulate the circuit by selecting run button.
- It will display the window showing the output graph of the respective simulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Now run the project using Pspice and note the waveforms for all logic gates.
- Then, save the practicals.

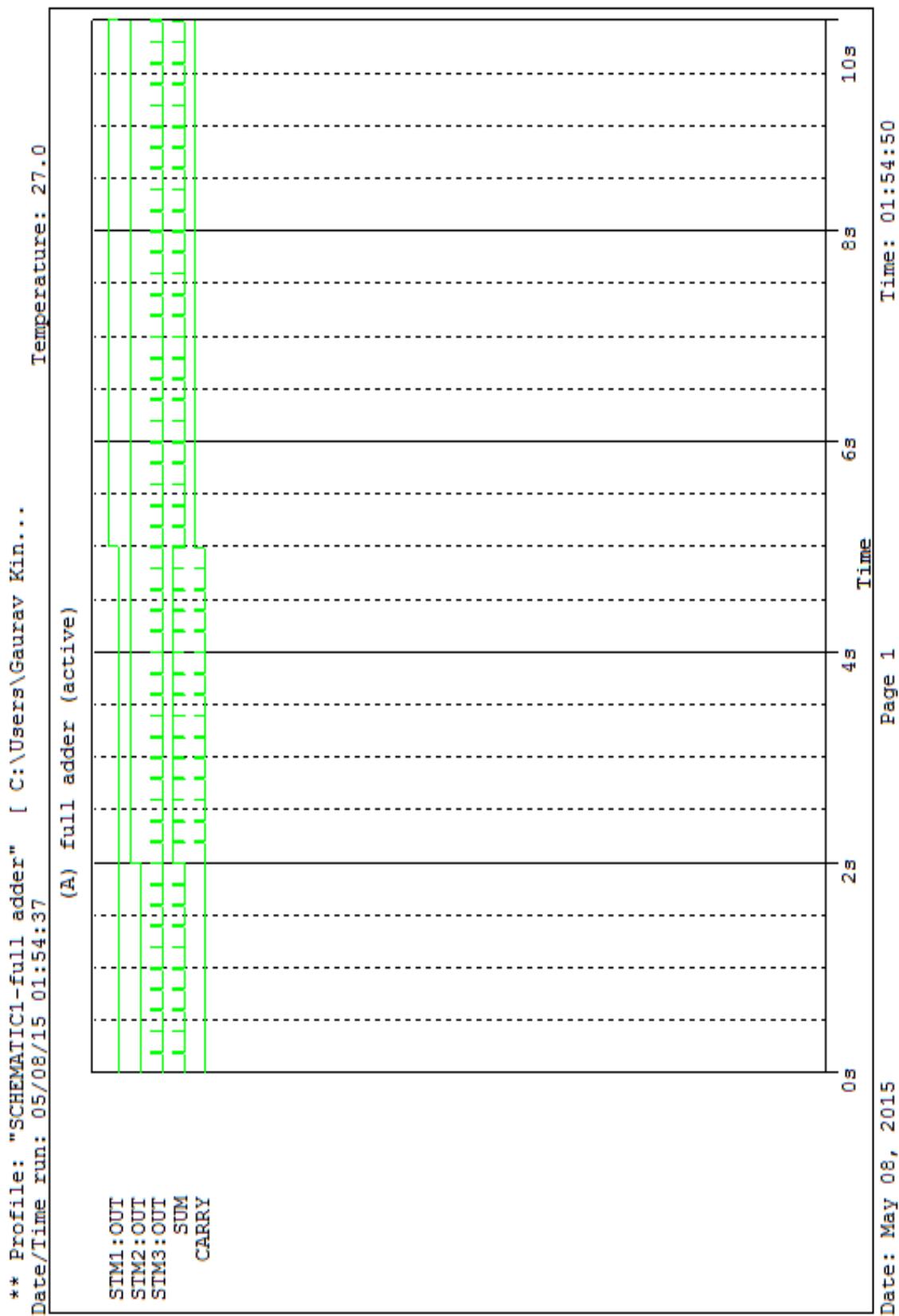
## CIRCUIT



## NET LIST

```
* source FULL ADDER
.EXTERNAL OUTPUT sum
.EXTERNAL OUTPUT carry
X_U1A      N00169 N00173 N00180 $G_DPWR $G_DGND 7486 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U1B      N00180 N00192 SUM $G_DPWR $G_DGND 7486 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1    STIM(1,0) $G_DPWR $G_DGND N00169 IO_STM
STIMULUS=A
U_DSTM2    STIM(1,0) $G_DPWR $G_DGND N00173 IO_STM
STIMULUS=B
U_DSTM3    STIM(1,0) $G_DPWR $G_DGND N00192 IO_STM
STIMULUS=Cin
X_U2A      N00169 N00173 N00393 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2B      N00180 N00192 N00405 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3A      N00393 N00405 CARRY $G_DPWR $G_DGND 7432
PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
```

## OUTPUT



## **CONCLUSION:**

By performing this practical, we conclude that we can study half adder and full adder using OrCAD software.

## PRACTICAL: 18

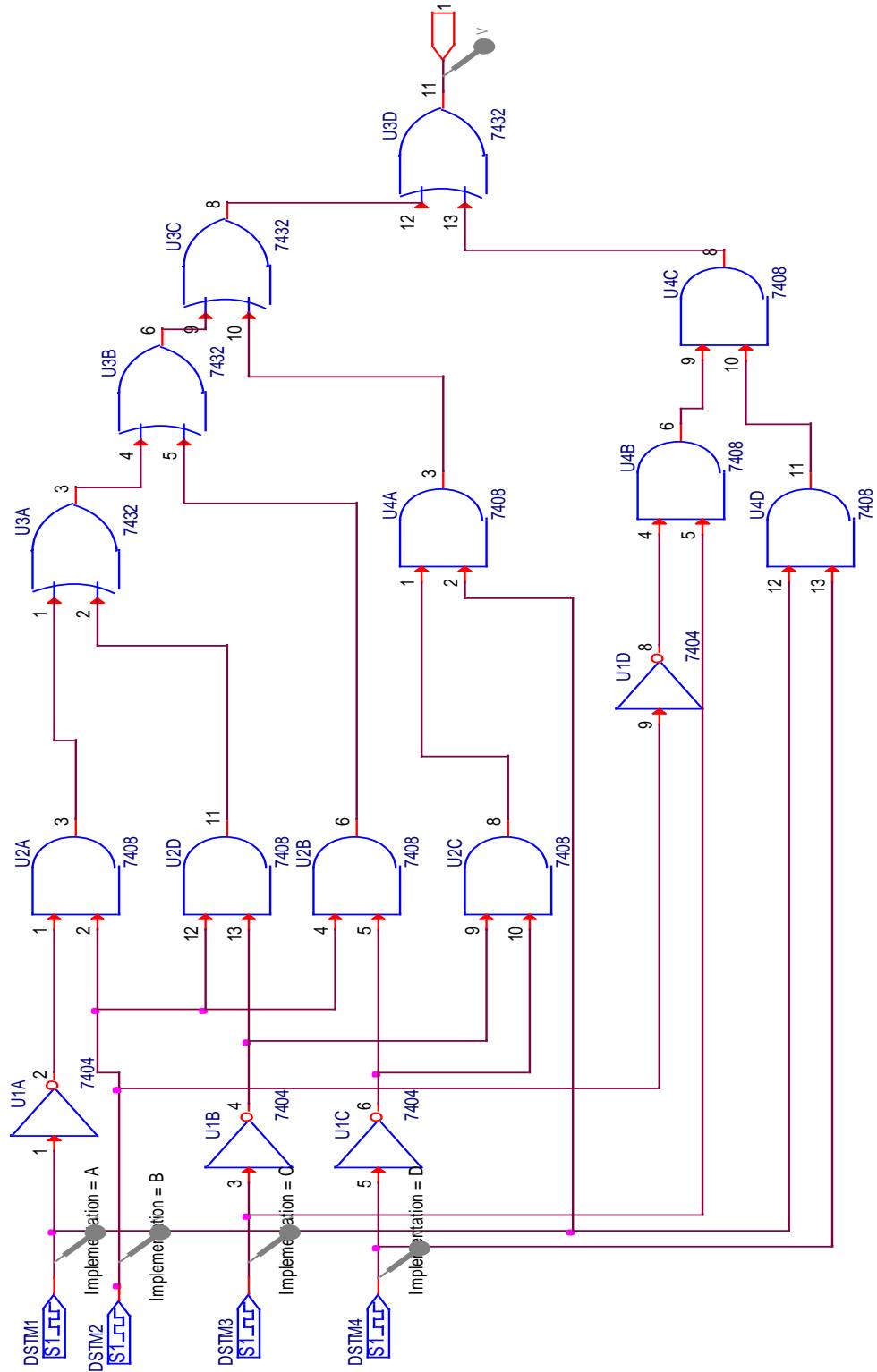
**AIM:** To implement a Boolean function :  $A(B+C)+BC'$

**SOFTWARE USED:** Orcad capture CIS 16.2

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Then save the practical.

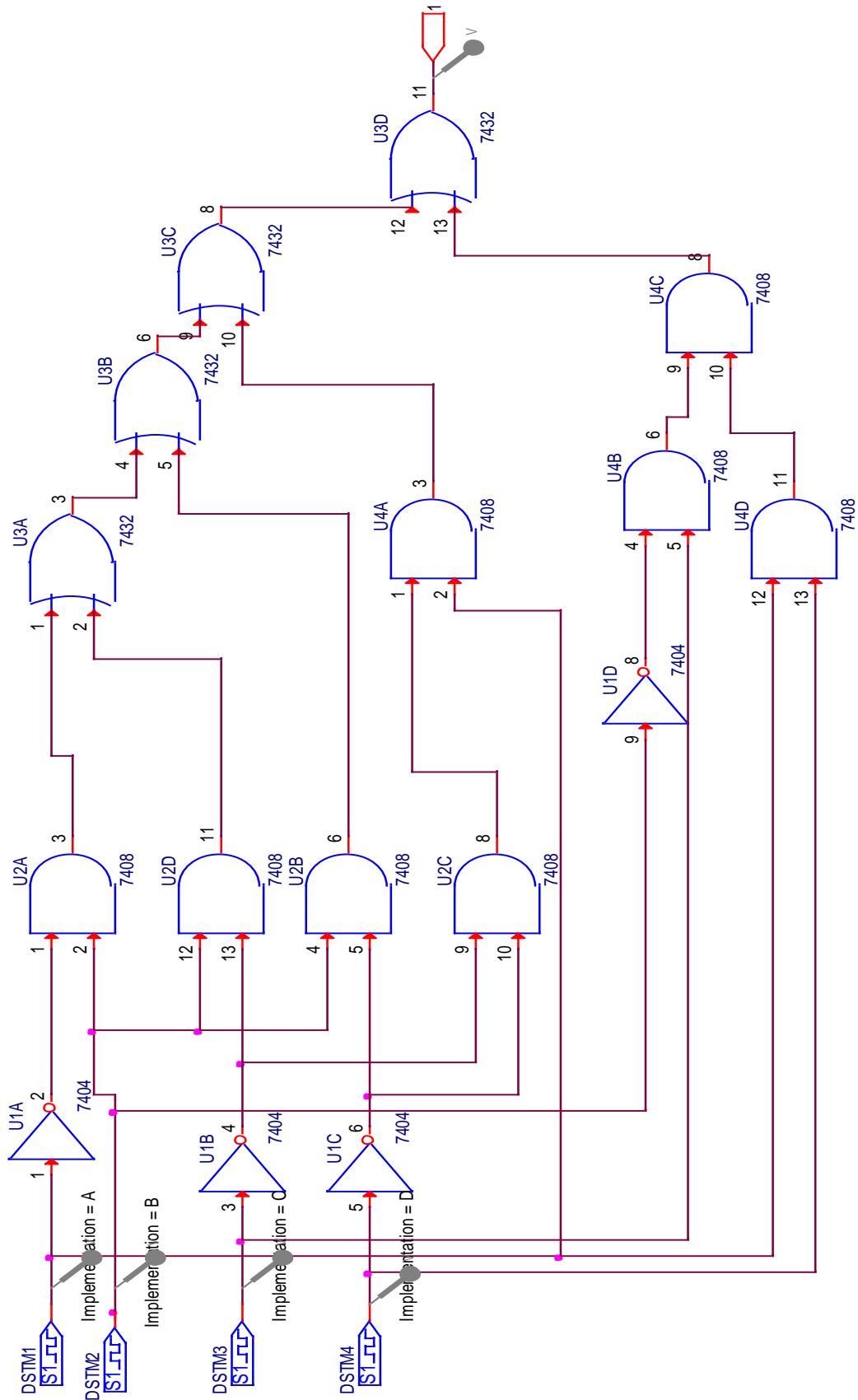
## CIRCUIT :



## NET LIST:

```
* source BOOLEAN
.EXTERNAL OUTPUT 1
X_U1A      N15369 N14700 $G_DPWR $G_DGND 7404 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U1B      N15397 N14748 $G_DPWR $G_DGND 7404 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U1C      N15405 N14760 $G_DPWR $G_DGND 7404 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U1D      N15139 N14994 $G_DPWR $G_DGND 7404 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2A      N14700 N15139 N14685 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2B      N15139 N14760 N15413 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2C      N14748 N14760 N15219 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2D      N15139 N14748 N14737 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3A      N14685 N14737 N14712 $G_DPWR $G_DGND 7432 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3B      N14712 N15413 N14719 $G_DPWR $G_DGND 7432 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3C      N14719 N15207 N15536 $G_DPWR $G_DGND 7432 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3D      N15536 N14972 1 $G_DPWR $G_DGND 7432 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U4A      N15219 N15369 N15207 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U4B      N14994 N15397 N15613 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U4C      N15613 N14987 N14972 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U4D      N15369 N15405 N14987 $G_DPWR $G_DGND 7408 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1    STIM(1,0) $G_DPWR $G_DGND N15369 IO_STM
STIMULUS=A
U_DSTM2    STIM(1,0) $G_DPWR $G_DGND N15139 IO_STM
STIMULUS=B
U_DSTM3    STIM(1,0) $G_DPWR $G_DGND N15397 IO_STM
STIMULUS=C
U_DSTM4    STIM(1,0) $G_DPWR $G_DGND N15405 IO_STM
STIMULUS=D
```

## OUTPUT



## **CONCLUSION :**

By performing this practical, we concluded that the given function can be implemented and also can be simulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect simulation we can implement it in the practical world.

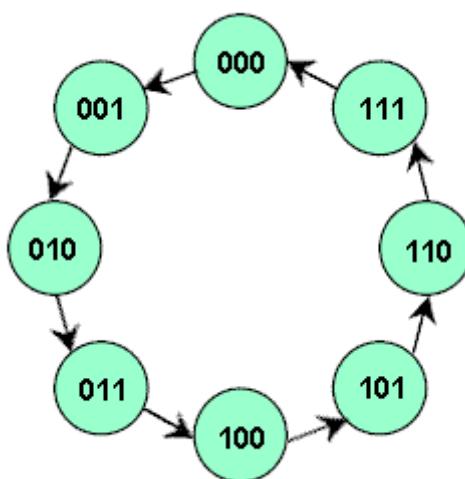
## PRACTICAL: 19

**AIM:** To implement and Simulate 3 –bit Counter Circuit using Or-Cad Capture 16.3

**SOFTWARE:** Orcad capture CIS 16.3

### THEORY

1. A sequential circuit that goes through a prescribed sequence of states upon the application of input pulses is called a counter. The input pulses, called count pulses, may be clock pulses. In a counter, the sequence of states may follow a binary count or any other sequence of states. Counters are found in almost all equipment containing digital logic. They are used for counting the number of occurrences of an event and are useful for generating timing sequences to control operations in a digital system.
2. Of the various sequences a counter may follow, the straight binary sequence is the simplest and most straightforward. A counter that follows the binary sequence is called a binary counter. An n-bit binary counter consists of n flip-flops and can count in binary from 0 to  $2^n - 1$ .
3. A counter is first described by a state diagram, which shows the sequence of states through which the counter advances when it is clocked. Figure 18 shows a state diagram of a 3-bit binary counter.



**State diagram of a 3-bit binary counter.**

4. The circuit has no inputs other than the clock pulse and no outputs other than its internal state (outputs are taken off each flip-flop in the counter).

The next state of the counter depends entirely on its present state, and the state transition occurs every time the clock pulse occurs.

5. Once the sequential circuit is defined by the state diagram, the next step is to obtain the next-state table, which is derived from the state diagram in Figure 18 and is shown in Table 15.

**State table**

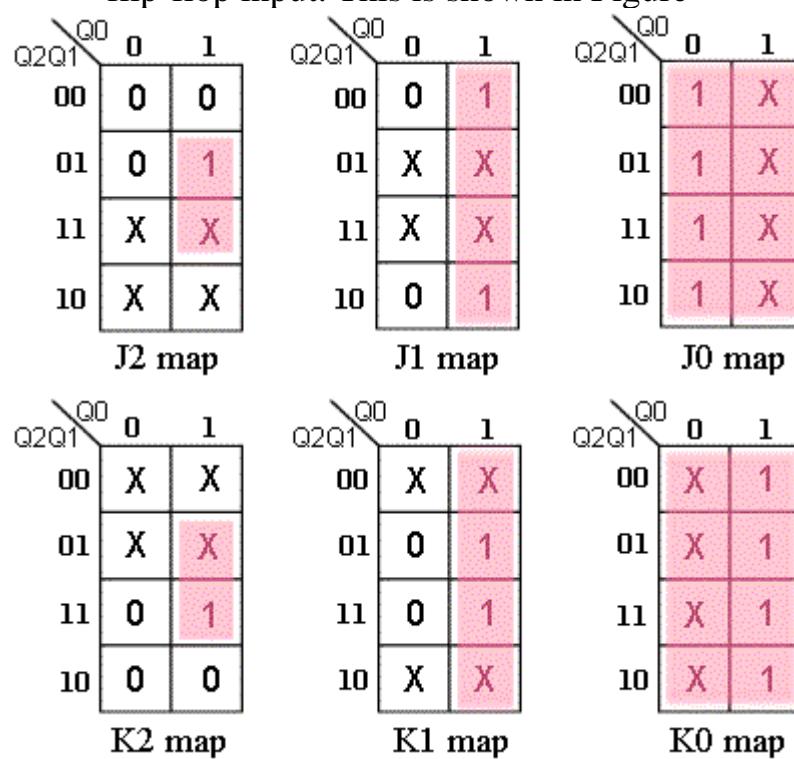
Present State			Next State		
Q2	Q1	Q0	Q2	Q1	Q0
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

6. Since there are eight states, the number of flip-flops required would be three. Now we want to implement the counter design using JK flip-flops. Next step is to develop an excitation table from the state table, which is shown in Table 16.

### Excitation table

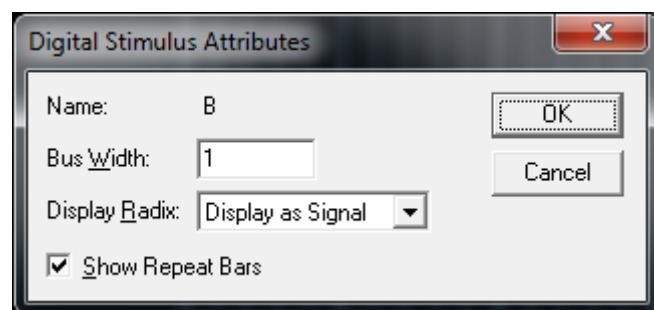
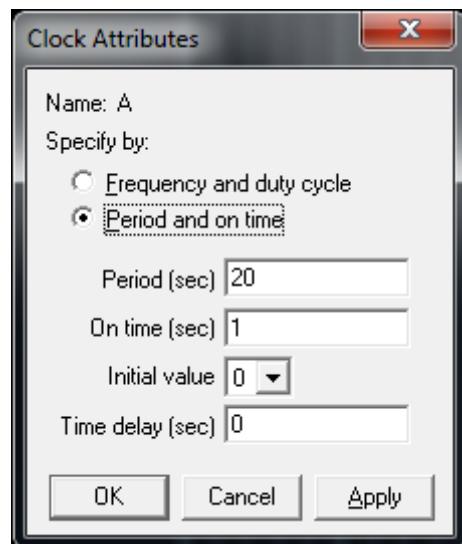
Output State Transitions			Flip-flop inputs								
Present State			Next State								
Q2 Q1 Q0			Q2 Q1 Q0			J2 K2 J1 K1 J0 K0					
0 0 0			0 0 1			J2	X	0 X	1 X		
0 0 1			0 1 0			K2	X	1 X	X 1		
0 1 0			0 1 1			J1	X	X 0	1 X		
0 1 1			1 0 0			K1	X	X 1	X 1		
1 0 0			1 0 1			J0	X	0 X	1 X		
1 0 1			1 1 0			K0	X	1 X	X 1		
1 1 0			1 1 1				X	0	X 0	1 X	
1 1 1			0 0 0				X	1	X 1	X 1	

7. Now transfer the JK states of the flip-flop inputs from the excitation table to Karnaugh maps to derive a simplified Boolean expression for each flip-flop input. This is shown in Figure

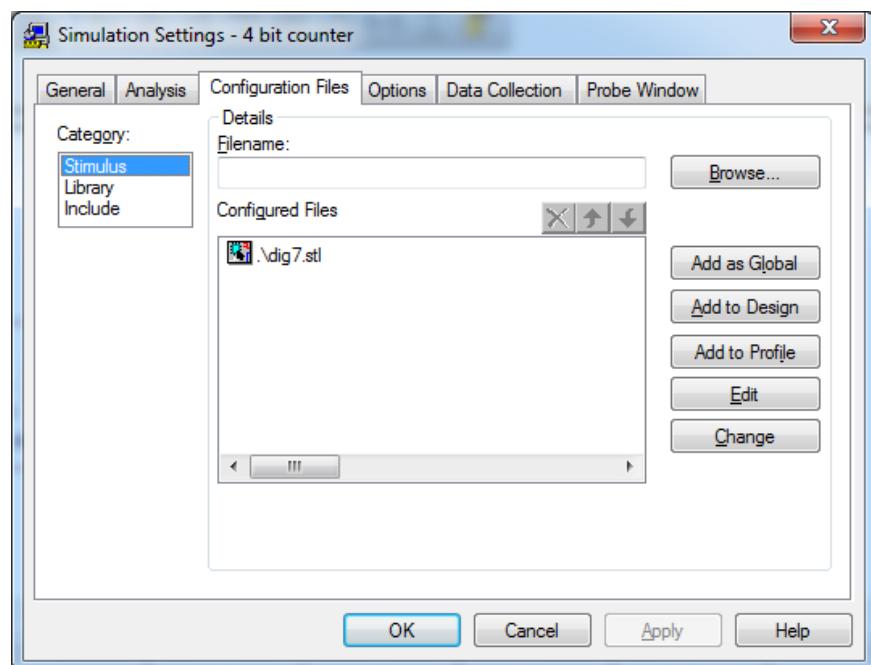
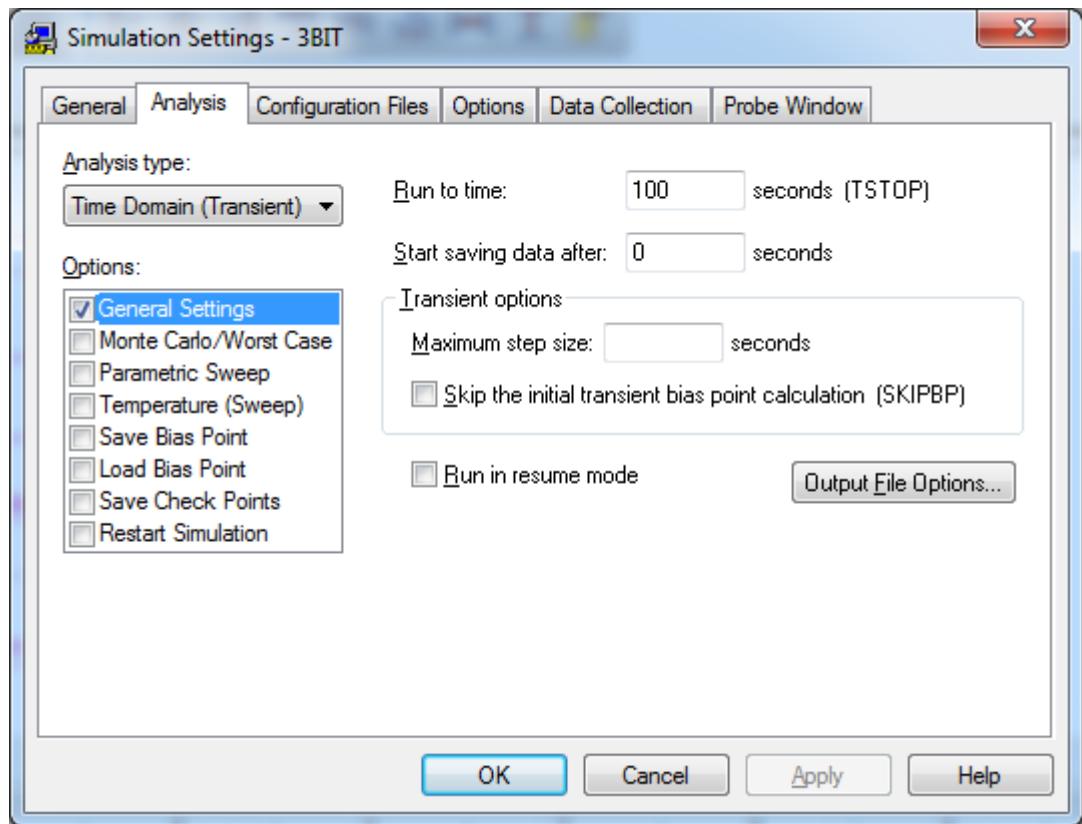


## **PROCEDURE:**

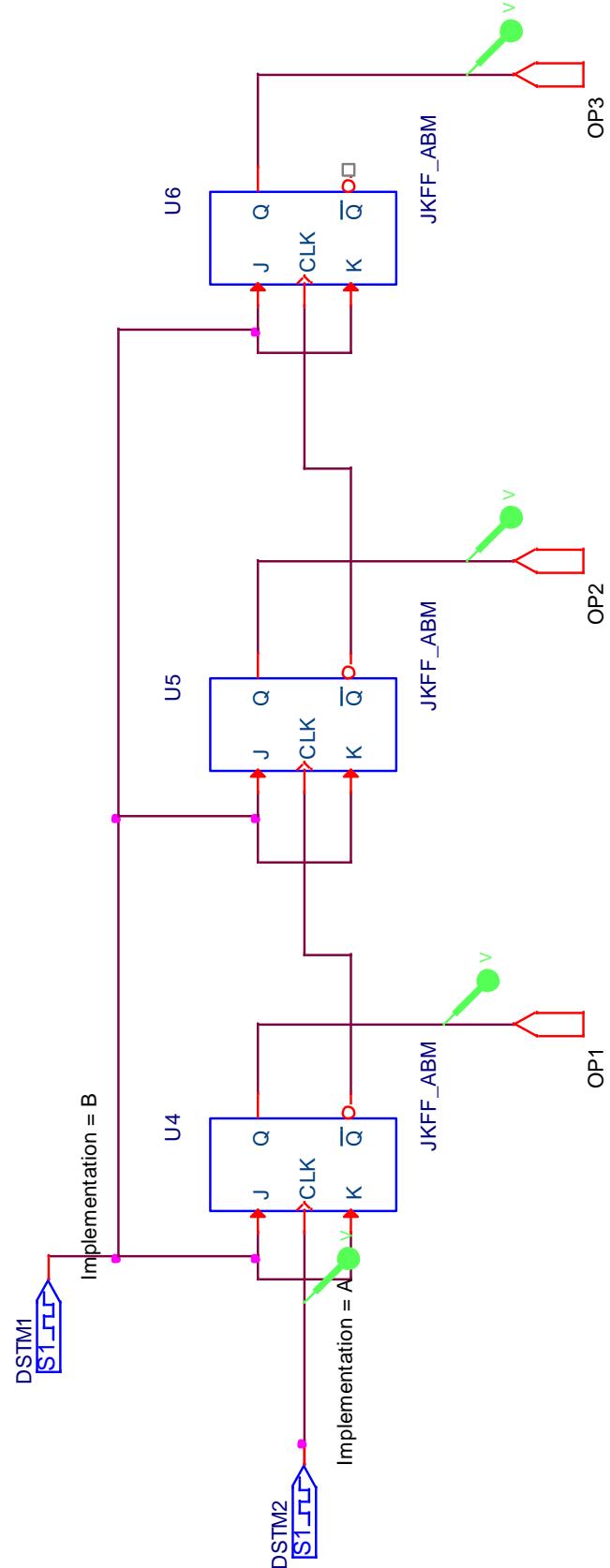
Input attributes for the JK FF



**IN stimulus editor**



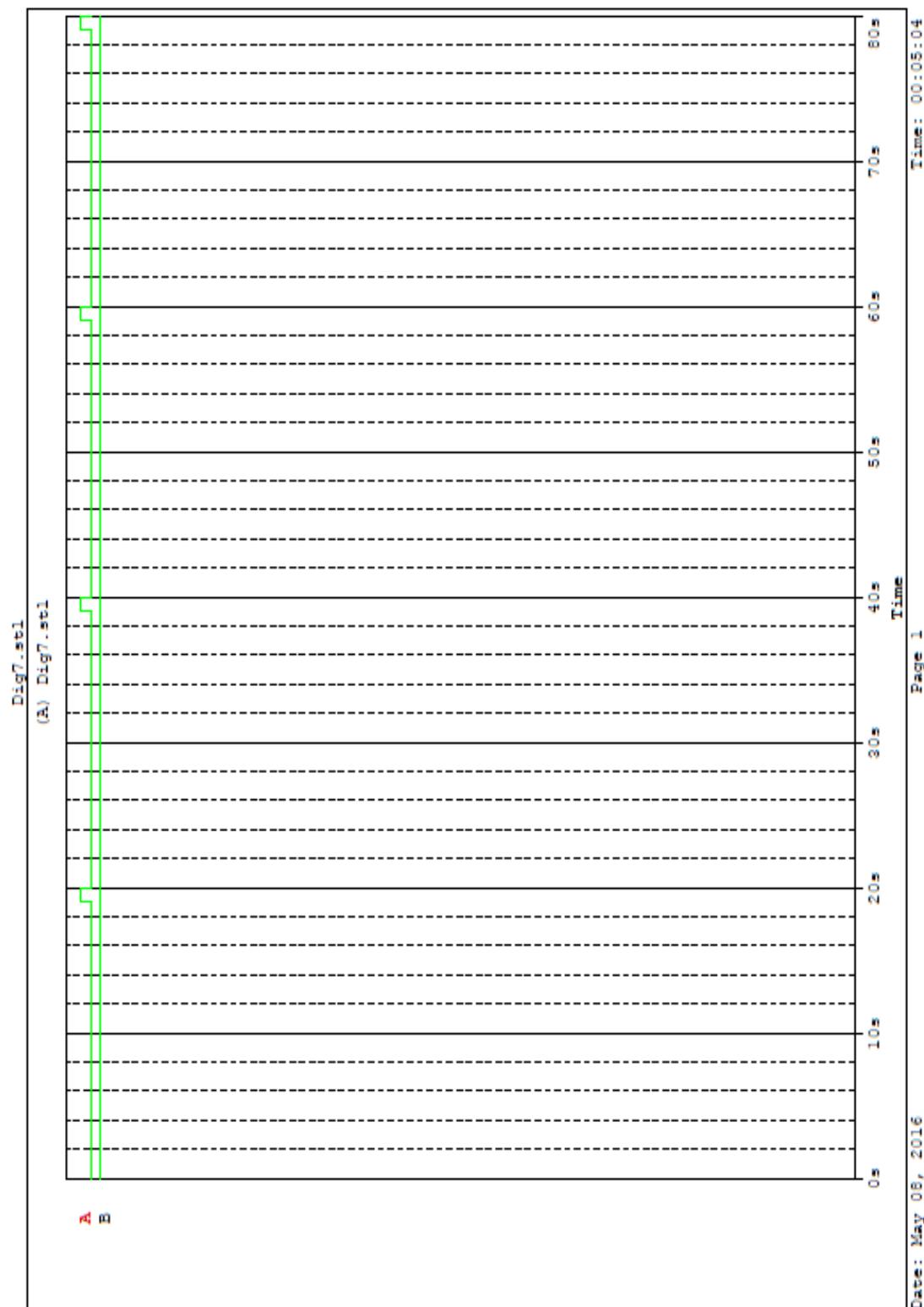
## CIRCUIT



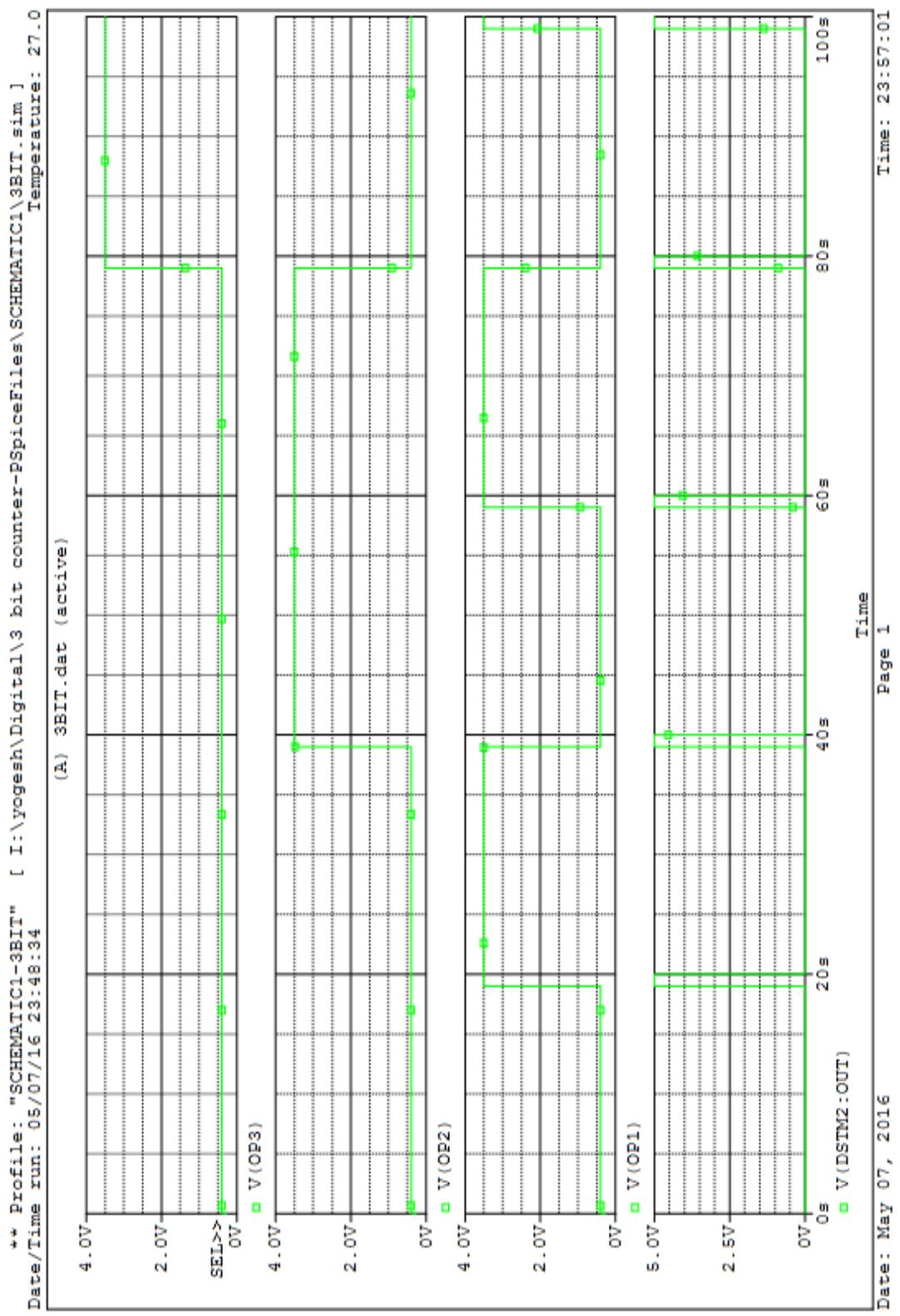
## NETLIST

```
* source 3 BIT COUNTER
.EXTERNAL OUTPUT OP1
.EXTERNAL OUTPUT OP2
.EXTERNAL OUTPUT OP3
U_DSTM1      STIM(1,0) $G_DPWR $G_DGND N00187 IO_STM
STIMULUS=B
U_DSTM2      STIM(1,0) $G_DPWR $G_DGND N00183 IO_STM
STIMULUS=A
X_U4        N00187 N00183 N00187 OP1 N00126 JKFF_ABM PARAMS:
VIH=0.8 VOH=3.5
+ VOL=0.4
X_U5        N00187 N00126 N00187 OP2 N00141 JKFF_ABM PARAMS:
VIH=0.8 VOH=3.5
+ VOL=0.4
X_U6        N00187 N00141 N00187 OP3 M_UN0001 JKFF_ABM PARAMS:
VIH=0.8
+ VOH=3.5 VOL=0.4
```

## OUTPUT:



## OUTPUT



## **CONCLUSION**

Thus we have simulated the 3 Bit counter Using ORCAD Capture CIS 16.3

## PRACTICAL: 20

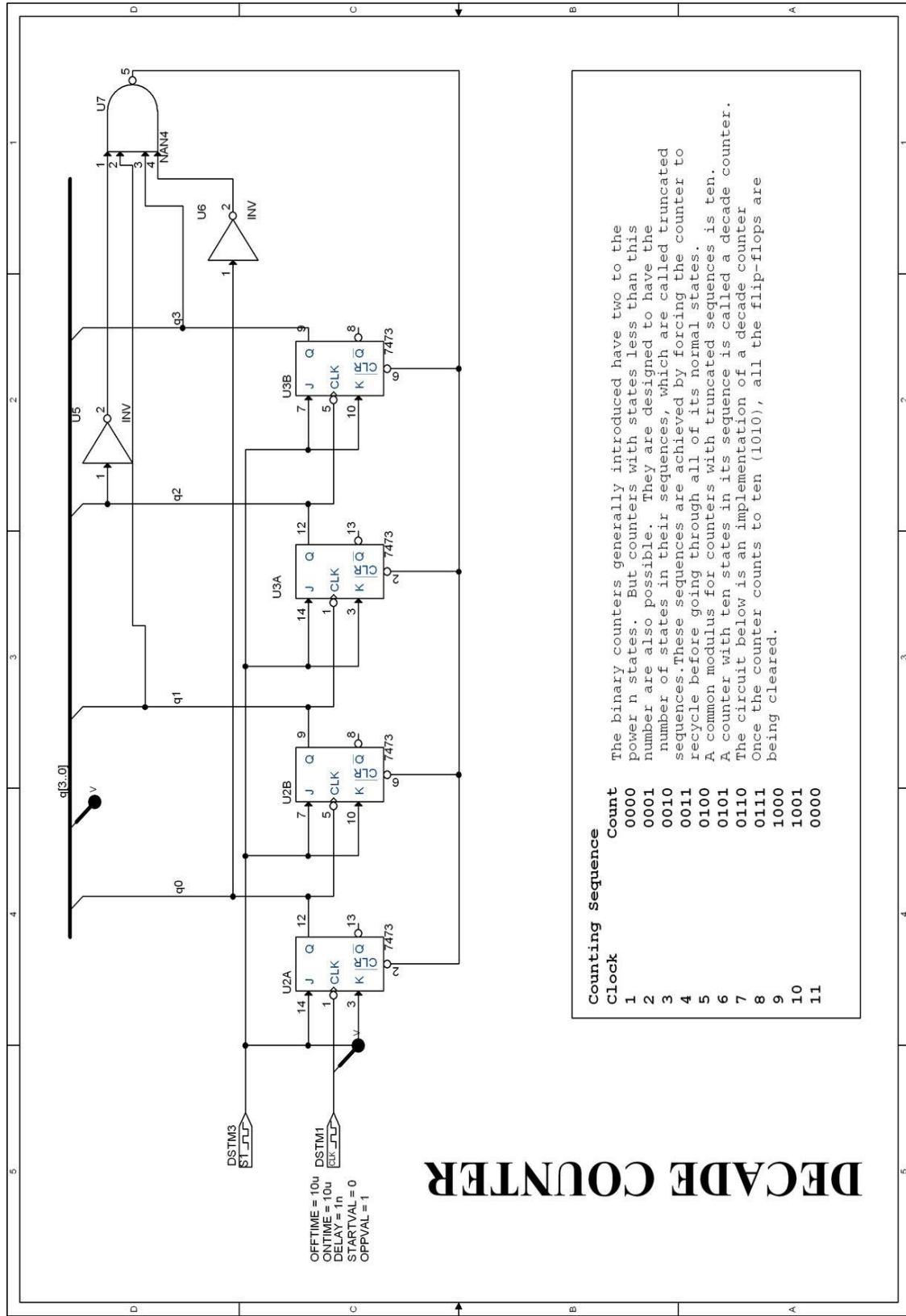
**AIM:** To implement and Simulate Decade Counter using OrCAD Capture 16.3

**SOFTWARE USED:** OrCAD Capture CIS 16.2

### PROCEDURE:

- Firstly go to start menu & open program & then go to cadence & from that open orcad capture.
- In orcad go to file menu, open new & in that new project, to create a blank project. In that the dialog box will appear, in that write your practical name & below write your roll no.
- It will display another dialog box. Select ‘Create a blank project’.
- Now a schematic profile will appear where you have to, make the circuit.
- Now you have to make a circuit, so for that in tool bar select place part & write your component’s name & double click on that & keep the components in their specific place in schematic profile & then join them by wires & place earthing.
- On completing the schematic diagram of the circuit, move cursor to the ‘Pspice’ option given in the toolbar and select the ‘New stimulation profile’. (For implementation of the circuit).
- It will display dialog box for editing the stimulation of the circuit. On completion of editing click ok.
- Now, stimulate the circuit by clicking the run button.
- It will display the window showing the output graph of the respective stimulated circuit. And if error is there, then it will show the ‘Net list’ in which detail info. For the error location is given.
- Then keep voltage marker in circuit to measure the circuit, then press run & the graph generate according to circuit.
- Then save the practical.

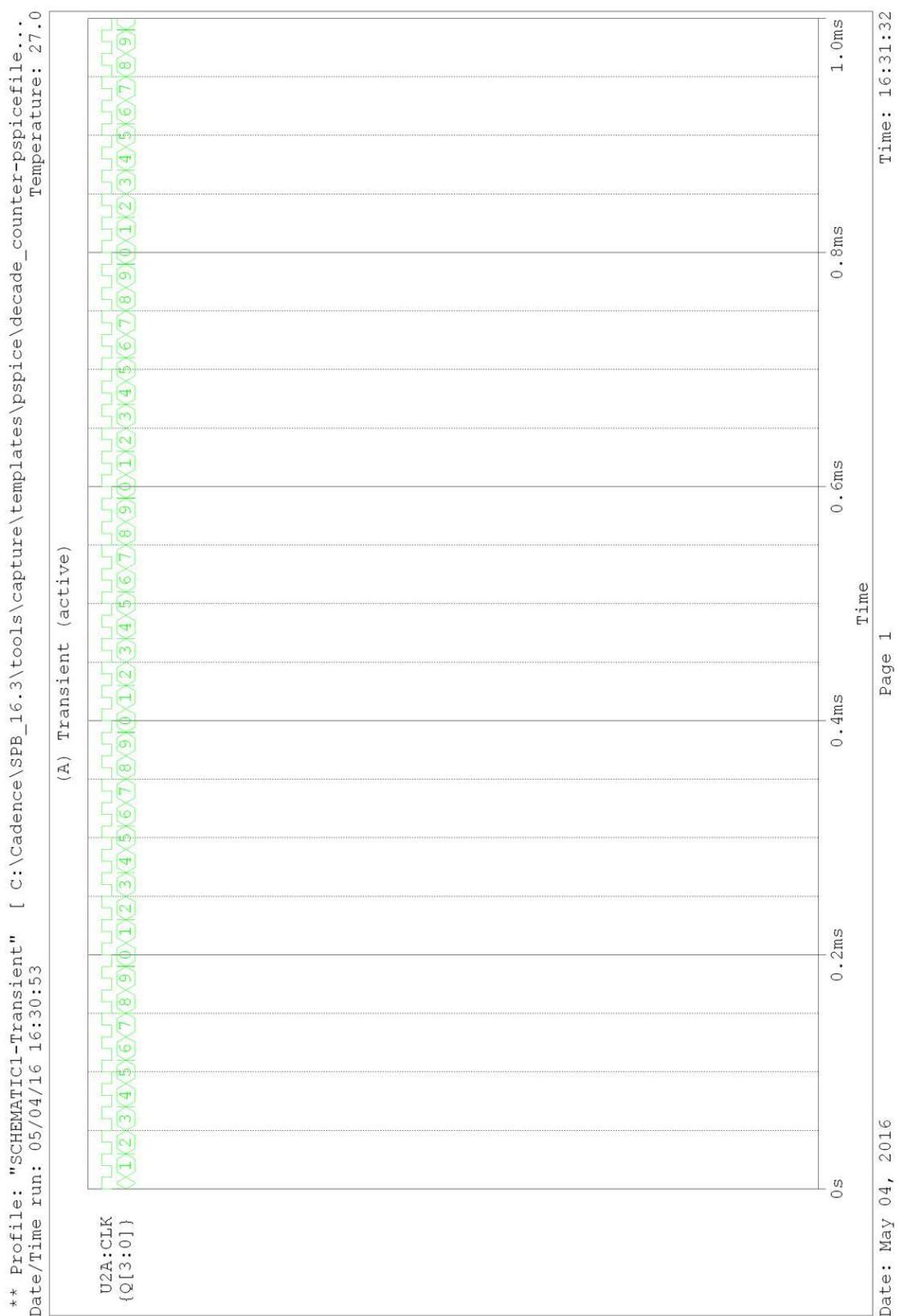
## CIRCUIT :



## NET LIST:

```
* source DECADE_COUNTER
X_U2A      N18935 N18978 N14547 N14547 Q0 M_UN0001 $G_DPWR
$G_DGND 7473
+ PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U2B      Q0 N18978 N14547 N14547 Q1 M_UN0002 $G_DPWR
$G_DGND 7473 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3A      Q1 N18978 N14547 N14547 Q2 M_UN0003 $G_DPWR
$G_DGND 7473 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
X_U3B      Q2 N18978 N14547 N14547 Q3 M_UN0004 $G_DPWR
$G_DGND 7473 PARAMS:
+ IO_LEVEL=0 MNTYMXDLY=0
U_DSTM1    STIM(1,1) $G_DPWR $G_DGND N18935 IO_STM
IO_LEVEL=0
+ 0 0
+ +1n 1
+REPEAT FOREVER
+ +10u 0
+ +10u 1
+ ENDREPEAT
U_DSTM3    STIM(1,1)
+ $G_DPWR $G_DGND
+ N14547
+ IO_STM
+ IO_LEVEL=0
+ 0s 1
X_U7       N19630 Q1 Q3 N19692 N18978 $G_DPWR $G_DGND NAN4
X_U6       Q0 N19692 $G_DPWR $G_DGND INV
X_U5       Q2 N19630 $G_DPWR $G_DGND INV
```

## OUTPUT:



## **CONCLUSION :**

By performing this practical, we concluded that the given function can be implemented and also can be simulated by using this software. Any error in the implementation of the circuit can be corrected in the software and by perfect simulation we can implement it in the practical world.

## PRACTICAL: 21

**AIM:** Introduction to MATLAB.

**SOFTWARE USED:** MATLAB v12.0

### **THEORY:**

#### **Introduction**

MATLAB stands for “*MATrix LABoratory*”. It is a technical computing environment for high performance numeric computation and visualisation. It integrates numerical analysis, matrix computation, signal processing and graphics in an easy-to-use environment, where problems and solutions are expressed just as they are written mathematically, without traditional programming. MATLAB allows us to express the entire algorithm in a few dozen lines, to compute the solution with great accuracy in a few minutes on a computer, and to readily manipulate a three-dimensional display of the result in colour.

MATLAB is an interactive system whose basic data element is a matrix that does not require dimensioning. It enables us to solve many numerical problems in a fraction of the time that it would take to write a program and execute in a language such as FORTRAN, BASIC, or C. It also features a family of application specific solutions, called **toolboxes**.

Areas in which toolboxes are available include signal processing, image processing, control systems design, dynamic systems simulation, systems identification, neural networks, wavelength communication and others. It can handle linear, non-linear, continuous-time, discrete-time, multivariable and multirate systems.

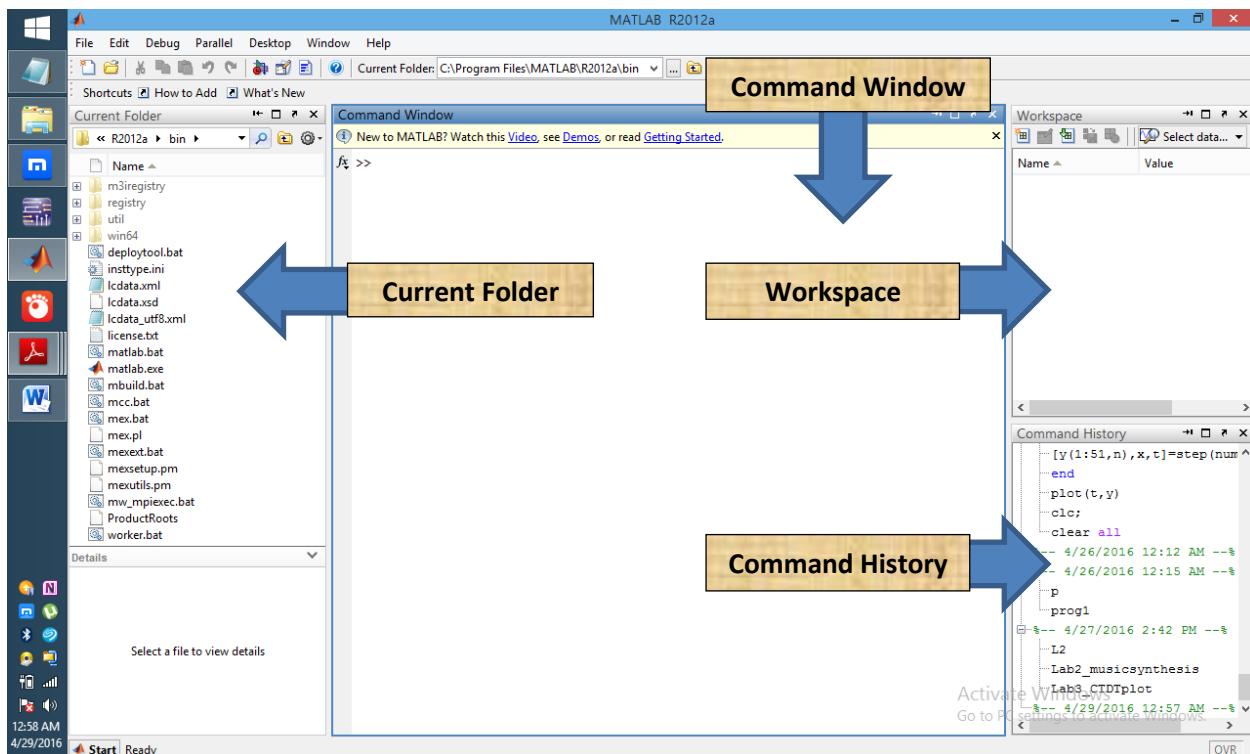
MATLAB is used in every facet of computational mathematics. Following are some

commonly used mathematical calculations where it is used most commonly:

1. Dealing with Matrices and Arrays
2. 2-D and 3-D Plotting and graphics
3. Linear Algebra
4. Algebraic Equations
5. Non-linear Functions
6. Statistics
7. Data Analysis
8. Calculus and Differential Equations

9. Numerical Calculations
10. Integration
11. Transforms
12. Curve Fitting
13. Various other special functions

## Understanding the MATLAB Environment



The desktop has the following panels:

1. **Current Folder** - This panel allows you to access the project folders and files.
2. **Command Window** - This is the main area where commands can be entered at the command line. It is indicated by the command prompt (>>).
3. **Workspace** - The workspace shows all the variables created and/or imported from files.
4. **Command History** - This panel shows or rerun commands that are entered at the command line.

## Working In The Command Window

1. As you work in MATLAB, you can enter individual statements in the Command Window. For example, create a variable named “a” by typing this statement at the command line:

a = 1

MATLAB immediately adds variable a to the workspace and displays the result in the Command Window.

a =

1

2. When you do not specify an output variable, MATLAB uses the variable ans, short for *answer*, to store the results of your calculation.

sin(a)

ans =

0.8415

The value of ans changes with every command that returns an output value that is not assigned to a variable.

3. If you end a statement with a semicolon, MATLAB performs the computation, but suppresses the display of output in the Command Window.

b = 2;

To enter multiple statements on multiple lines before running any of the statements, use **Shift+Enter** between statements. This action is unnecessary when you enter a paired keyword statement on multiple lines, such as for and end.

4. You also can enter more than one statement on the same line by separating statements. To distinguish between commands, end each one with a comma or semicolon.

Commands that end with a comma display their results, while commands that end with a semicolon do not.

For example, enter the following three statements at the command line:

A = magic(5), B = ones(5) \* 4.7; C = A./B

A =

```

17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9

```

C =

```

3.6170 5.1064 0.2128 1.7021 3.1915
4.8936 1.0638 1.4894 2.9787 3.4043
0.8511 1.2766 2.7660 4.2553 4.6809
2.1277 2.5532 4.0426 4.4681 0.6383
2.3404 3.8298 5.3191 0.4255 1.9149

```

MATLAB displays only the values of A and C in the Command Window.

5. To recall previous lines in the Command Window, press the up- and down-arrow keys,  $\uparrow$  and  $\downarrow$ . Press the arrow keys either at an empty command line or after you type the first few characters of a command. For example, to recall the command  $b = 2$ , type b, and then press the up-arrow key.
6. To clear a command from the Command Window without executing it, press the Escape (**Esc**) key.
7. You can evaluate any statement already in the Command Window. Select the statement, right-click, and then select **Evaluate Selection**.
8. In the Command Window, you also can execute only a portion of the code currently at the command prompt. To evaluate a portion of the entered code, select the code, and then press **Enter**.  
For example, select a portion of the following code:

The screenshot shows the MATLAB Command Window. The text 'disp('hello'), disp('world')' is typed into the window. The word 'hello' is highlighted with a yellow rectangular selection box. The MATLAB logo and the text '>>' are visible to the left of the code.

## Commonly Used Characters

Operator	Purpose
+	Plus; addition operator.
-	Minus; subtraction operator.
*	Scalar and matrix multiplication operator.
.*	Array multiplication operator.
.^	Scalar and matrix exponentiation operator.
.^	Array exponentiation operator.
\	Left-division operator.
/	Right-division operator.
.\	Array left-division operator.
./	Array right-division operator.
:	Colon; generates regularly spaced elements and represents an entire row or column.
( )	Parentheses; encloses function arguments and array indices; overrides precedence.
[ ]	Brackets; enclosures array elements.
.	Decimal point.
...	Ellipsis; line-continuation operator
,	Comma; separates statements and elements in a row
;	Semicolon; separates columns and suppresses display.
%	Percent sign; designates a comment and specifies formatting.
-	Quote sign and transpose operator.
.-	Non-conjugated transpose operator.
=	Assignment operator.

## Commonly Used Constants

Name	Meaning
ans	Most recent answer.
eps	Accuracy of floating-point precision.
i,j	The imaginary unit $\sqrt{-1}$ .
Inf	Infinity.
NaN	Undefined numerical result (not a number).
pi	The number $\pi$

## Commands

### Session Management

Command	Purpose
clc	Clears command window.
clear	Removes variables from memory.
exist	Checks for existence of file or variable.
global	Declares variables to be global.
help	Searches for a help topic.
lookfor	Searches help entries for a keyword.
quit	Stops MATLAB.
who	Lists current variables.
whos	Lists current variables (long display).

### System Operation

Command	Purpose
cd	Changes current directory.
date	Displays current date.
delete	Deletes a file.
diary	Switches on/off diary file recording.
dir	Lists all files in current directory.
load	Loads workspace variables from a file.
path	Displays search path.
pwd	Displays current directory.
save	Saves workspace variables in a file.
type	Displays contents of a file.
what	Lists all MATLAB files in the current directory.
wk1read	Reads .wk1 spreadsheet file.

### Session Management

Command	Purpose
disp	Displays contents of an array or string.
fscanf	Read formatted data from a file.
format	Controls screen-display format.
fprintf	Performs formatted writes to screen or file.
input	Displays prompts and waits for input.
;	Suppresses screen printing.

The **fscanf** and **fprintf** commands behave like C's **scanf** and **printf** functions.

## Vector, Matrix & Array

Command	Purpose
<b>cat</b>	Concatenates arrays.
<b>find</b>	Finds indices of nonzero elements.
<b>length</b>	Computes number of elements.
<b>linspace</b>	Creates regularly spaced vector.
<b>logspace</b>	Creates logarithmically spaced vector.
<b>max</b>	Returns largest element.
<b>min</b>	Returns smallest element.
<b>prod</b>	Product of each column.
<b>reshape</b>	Changes size.
<b>size</b>	Computes array size.
<b>sort</b>	Sorts each column.
<b>sum</b>	Sums each column.
<b>eye</b>	Creates an identity matrix.
<b>ones</b>	Creates an array of ones.
<b>zeros</b>	Creates an array of zeros.
<b>cross</b>	Computes matrix cross products.
<b>dot</b>	Computes matrix dot products.
<b>det</b>	Computes determinant of an array.
<b>inv</b>	Computes inverse of a matrix.
<b>pinv</b>	Computes pseudoinverse of a matrix.
<b>rank</b>	Computes rank of a matrix.
<b>rref</b>	Computes reduced row echelon form.
<b>cell</b>	Creates cell array.
<b>celldisp</b>	Displays cell array.
<b>cellplot</b>	Displays graphical representation of cell array.
<b>num2cell</b>	Converts numeric array to cell array.
<b>deal</b>	Matches input and output lists.
<b>iscell</b>	Identifies cell array.

## Plots

Command	Purpose
<b>axis</b>	Sets axis limits.
<b>fplot</b>	Intelligent plotting of functions.
<b>grid</b>	Displays gridlines.
<b>plot</b>	Generates xy plot.
<b>print</b>	Prints plot or saves plot to a file.
<b>title</b>	Puts text at top of plot.
<b>xlabel</b>	Adds text label to x-axis.
<b>ylabel</b>	Adds text label to y-axis.
<b>axes</b>	Creates axes objects.
<b>close</b>	Closes the current plot.
<b>close all</b>	Closes all plots.
<b>figure</b>	Opens a new figure window.
<b>gtext</b>	Enables label placement by mouse.
<b>hold</b>	Freezes current plot.
<b>legend</b>	Legend placement by mouse.
<b>refresh</b>	Redraws current figure window.
<b>set</b>	Specifies properties of objects such as axes.
<b>subplot</b>	Creates plots in sub windows.
<b>text</b>	Places string in figure.
<b>bar</b>	Creates bar chart.
<b>loglog</b>	Creates log-log plot.
<b>polar</b>	Creates polar plot.
<b>semilogx</b>	Creates semi log plot. (logarithmic abscissa).
<b>semilogy</b>	Creates semi log plot. (logarithmic ordinate).
<b>stairs</b>	Creates stairs plot.
<b>stem</b>	Creates stem plot.

## Syntax Of Inbuilt Functions

### **plot:**

#### **Syntax:**

```
plot(Y)
plot(X1,Y1,...,Xn,Yn) plot(X1,Y1,LineSpec,...,Xn,Yn,LineSpec)
plot(...,'PropertyName',PropertyValue,...) plot(axes_handle,...)
h = plot(...)
```

#### **Description:**

plot(Y) plots the columns of Y versus the index of each value when Y is a real number. For complex Y, plot(Y) is equivalent to plot(real(Y),imag(Y)).

### **stem:**

#### **Syntax:**

```
stem(Y) stem(X,Y) stem(...,'fill')
stem(...,LineSpec) stem(...,'PropertyName',PropertyValue,...) stem(axes_handle,...)
h = stem(...)
```

#### **Description:**

A 2D stem plot displays data as lines extending from a baseline along the x-axis. A circle (the default) or other marker whose y-position represents the data value terminates each stem.

### **figure:**

#### **Syntax:**

```
figure
figure('PropertyName',PropertyValue,...) figure(h)
h = figure(...)
```

#### **Description:**

figure creates figure graphics objects. Figure objects are the individual windows on the screen in which the MATLAB software displays graphical output.

### **axis:**

#### **Syntax:**

```
axis([xmin xmax ymin ymax])
axis([xmin xmax ymin ymax zmin zmax cmin cmax])
v = axis
axis auto
axis manual
axis tight
axis fill
axis ij
axis xy
axis equal
axis image
axis square
axis vis3d
axis normal
axis off
axis on
axis(axes_handles,...)
[mode,visibility,direction] = axis('state')
```

#### **Description:**

axis manipulates commonly used axes properties. (See Algorithm section.)

axis([xmin xmax ymin ymax]) sets the limits for the x- and y-axis of the current axes.

axis([xmin xmax ymin ymax zmin zmax cmin cmax]) sets the x-, y-, and z-axis limits and the color scaling limits (see caxis) of the current axes.

v = axis returns a row vector containing scaling factors for the x-, y-, and z-axis. v has four or six components depending on whether the current axes is 2-D or 3-D, respectively. The returned values are the current axes XLim, Ylim, and ZLim properties.

axis auto sets MATLAB default behavior to compute the current axes limits automatically, based on the minimum and maximum values of x, y, and z data. You can restrict this automatic behavior to a specific axis. For example, axis 'auto x' computes only the x-axis limits automatically; axis 'auto yz' computes the y- and z-axis limits automatically.

axis manual and axis(axis) freezes the scaling at the current limits, so that if hold is on, subsequent plots use the same limits. This sets the XLimMode, YLimMode, and ZLimMode properties to manual.

axis tight sets the axis limits to the range of the data.

**xlabel:****Syntax:**

```
xlabel('string') xlabel(fname)  
xlabel(...,'PropertyName',PropertyValue,...) xlabel(axes_handle,...)  
h = xlabel(...)
```

**Description:**

Every axes graphics object can have one label for the x-axis. The label appears beneath its respective axis in a two-dimensional plot and to the side or beneath the axis in a 3D plot.

**ylabel:****Syntax:**

```
ylabel('string') ylabel(fname)  
ylabel(...,'PropertyName',PropertyValue,...) ylabel(axes_handle,...)  
h = ylabel(...)
```

**Description:**

Every axes graphics object can have one label for the y-axis. The label appears beneath its respective axis in a two-dimensional plot and to the side or beneath the axis in a 3D plot.

**grid:****Syntax:**

```
grid on  
grid off  
grid  
grid(axes_handle,...) grid minor
```

**Description:**

The grid function turns the current axes' grid lines on and off.

**title:****Syntax:**

```
title('string') title(fname)  
title(...,'PropertyName',PropertyValue,...) title(axes_handle,...)  
h = title(...)
```

**Description:**

Each axes graphics object can have one title. The title is located at the top and in the center of the axes.

## **subplot:**

### **Syntax:**

```

h = subplot(m,n,p) or subplot(mnp) subplot(m,n,p,'replace') subplot(m,n,P)
subplot(h)
subplot('Position',[left bottom width height])
subplot(..., prop1, value1, prop2, value2, ...)
h = subplot(...)

```

### **Description:**

subplot divides the current figure into rectangular panes that are numbered rowwise. Each pane contains an axes object which you can manipulate using Axes Properties. Subsequent plots are output to the current pane.

## **conv:**

### **Syntax:**

```

w = conv(u,v)
w = conv(...,'shape')

```

### **Description:**

w = conv(u,v) convolves vectors u and v. Algebraically, convolution is the same operation as multiplying the polynomials whose coefficients are the elements of u and v.

## **fft:**

### **Syntax:**

```

Y = fft(x)
Y = fft(X,n)
Y = fft(X,[],dim)
Y = fft(X,n,dim)

```

### **Description:**

Y = fft(x) returns the discrete Fourier transform (DFT) of vector x, computed with a fast Fourier transform (FFT) algorithm.

**zeros:****Syntax:**

```
B = zeros(n)
B = zeros(m,n)
B = zeros([m n])
B = zeros(m,n,p,...)
B = zeros([m n p ...])
B = zeros(size(A))
Y = zeros
zeros(m, n,...,classname)
zeros([m,n,...],classname)
```

**Description:**

`B = zeros(n)` returns an  $n$ -by- $n$  matrix of zeros. An error message appears if  $n$  is not a scalar.

**sound:****Syntax:**

```
sound(y,Fs)
sound(y,Fs,bits)
```

**Description:**

`sound(y,Fs)` sends audio signal  $y$  to the speaker at sample rate  $Fs$ . If you do not specify a sample rate, `sound` plays at 8192 Hz. For single-channel (mono) audio,  $y$  is an  $m$ -by-1 column vector, where  $m$  is the number of audio samples. If your system supports stereo playback,  $y$  can be an  $m$ -by-2 matrix, where the first column corresponds to the left channel, and the second column corresponds to the right channel. The `sound` function assumes that  $y$  contains floating-point numbers between -1 and 1, and clips values outside that range.

**sound:****Syntax:**

```
sound(y,Fs)
sound(y,Fs,bits)
```

**Description:**

`sound(y,Fs)` sends audio signal  $y$  to the speaker at sample rate  $Fs$ . If you do not specify a sample rate, `sound` plays at 8192 Hz. For single-channel (mono) audio,  $y$  is an  $m$ -by-1 column vector, where  $m$  is the number of audio samples. If your system supports stereo playback,  $y$  can be an  $m$ -by-2 matrix, where the first column corresponds to the left channel, and the second column corresponds to the right channel. The `sound` function assumes that  $y$  contains floating-point numbers between -1 and 1, and clips values outside that range.