ICA & HDL SIMULATION

LABORATORY MANUAL

III – I SEMESTER

Prepared By

Mr. K.Murali Krishna, **Associate Professor**

Mrs. S.Rajani, **Assistant Professor**

DEPARTMENT OF ELECTRONICS AND COMMUNICATIONS ENGG

MALLA REDDY COLLEGE OF ENGINEERING AND TECHNOLOGY

(Sponsored by CMR Educational Society)

(Affiliated to JNTU, Hyderabad) Secunderabad-14.

CONTENTS

CYCLE-I

S.No.	Experiment Name	Page No.
1	INTRODUCTION - STUDY OF IC741, IC555 & IC565	3
2	OP-AMP APPLICATIONS – ADDER, SUBTRACTOR, COMPARATOR	8
3	INTEGRATOR AND DIFFERENTIATOR USING IC741 OP-AMP	13
4	ACTIVE FILTER APPLICATIONS – LPF AND HPF(1^{ST} ORDER)	18
5	IC741 WAVEFORM GENERATORS – SINE, SQAURE AND TRIANGULAR WAVES	26
6	IC555 TIMER – MONOSTABLE AND ASTABLE MULTIVIBRATORS	31
7	SCHMITT TRIGGER CIRCUIT USING IC 741	38
8	IC 565 – PLL APPLICATIONS	41
9	VOLTAGE REGULATORS USING IC 723, THREE TERMINAL VOLTAGE REGULATORS – 7805, 7809 AND 7912	44

CYCLE-II

S.No.	Experiment Name	Page No.
1	INTRODUCTION - Xilinx	52
2	HDL CODE TO REALIZE ALL LOGIC GATES	61
3	DESIGN OF 2-TO-4 ENCODER	63
4	DESIGN OF 8-TO-3 ENCODER	65
5	DESIGN OF 8-TO-1 MULTIPLEXER AND 1X8 DEMULTIPLEXER	67

6	DESIGN OF 4 BIT BINARY TO GRAY CODE CONVERTER	70
7	DESIGN OF 4-BIT COMPARATOR	72
8	DESIGN OF FULL ADDER USING THREE MODELLING STYLES	74
9	DESIGN OF FLIP FLOPS (SR,JK,D,T)	77
10	DESIGN OF 4 BIT BINARY ,BCD COUNTERS(SYNCHRONUS AND ASYNCHRONUS RESET)	83
11	FINITE STATE MACHINE DESIGN	86

VISION

To evolve into a center of excellence in Engineering Technology through creative and innovative practices in teaching-learning, promoting academic achievement & research excellence to produce internationally accepted competitive and world class professionals.

MISSION

To provide high quality academic programmes, training activities, research facilities and opportunities supported by continuous industry institute interaction aimed at employability, entrepreneurship, leadership and research aptitude among students.

QUALITY POLICY

- Impart up-to-date knowledge to the students in Electronics & Communication area to make them quality engineers.
- Make the students experience the applications on quality equipment and tools.
- Provide systems, resources and training opportunities to achieve continuous improvement.
- Maintain global standards in education, training and services.

PROGRAMME EDUCATIONAL OBJECTIVES

PEO1: PROFESSIONALISM & CITIZENSHIP To create and sustain a community of learning in which students acquire knowledge and learn to apply it professionally with due consideration for ethical, ecological and economic issues.
PEO2: TECHNICAL ACCOMPLISHMENTS To provide knowledge based services to satisfy the needs of society and the industry by providing hands on experience in various technologies in core field.
PEO3: INVENTION, INNOVATION AND CREATIVITY To make the students to design, experiment, analyze, interpret in the core field with the help of other multi disciplinary concepts wherever applicable.
PEO4: PROFESSIONAL DEVELOPMENT To educate the students to disseminate research findings with good soft skills and become a successful entrepreneur.
PEO5: HUMAN RESOURCE DEVELOPMENT To graduate the students in building national capabilities in technology, education and research.

CODE OF CONDUCT FOR THE LABORATORIES

- All students must observe the Dress Code while in the laboratory.
- Sandals or open-toed shoes are NOT allowed.
- Foods, drinks and smoking are NOT allowed.
- All bags must be left at the indicated place.
- The lab timetable must be strictly followed.
- Be PUNCTUAL for your laboratory session.
- Program must be executed within the given time.
- Noise must be kept to a minimum.
- Workspace must be kept clean and tidy at all time.
- Handle the systems and interfacing kits with care.
- All students are liable for any damage to the accessories due to their own negligence.
- All interfacing kits connecting cables must be RETURNED if you taken from the lab supervisor.
- Students are strictly PROHIBITED from taking out any items from the laboratory.
- Students are NOT allowed to work alone in the laboratory without the Lab Supervisor
- USB Ports have been disabled if you want to use USB drive consult lab supervisor.
- Report immediately to the Lab Supervisor if any malfunction of the accessories, is there.

Before leaving the lab

- Place the chairs properly.
- Turn off the system properly
- Turn off the monitor.
- Please check the laboratory notice board regularly for updates.

CYCLE - I

INTRODUCTION

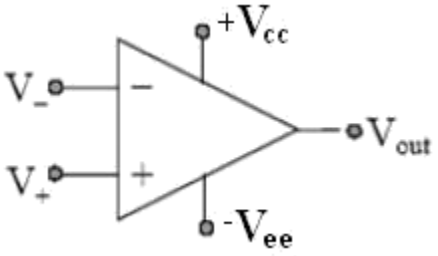
STUDY OF IC741, IC555 & IC565

AIM: To study pin details, specifications, applications and features of IC741 (Op-Amp) IC555 (Timer) & IC565.

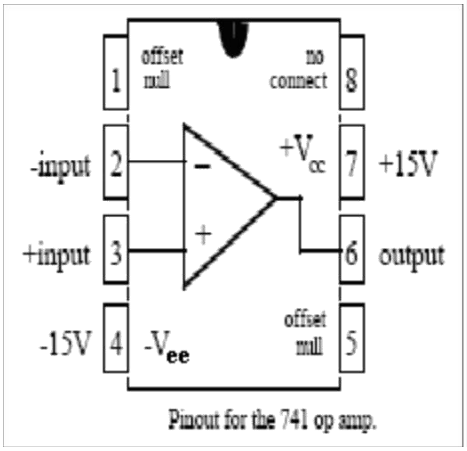
COMPONENTS: IC741, IC555 & IC565

IC741: (OPERATIONAL AMPLIFIER)

Symbol:



PIN CONFIGURATION:



SPECIFICATIONS:

Supply Voltage	**± 18V**
Internal Power Dissipation	310mw
Differential input voltage	± 30V
Input Voltage	**± 15V**
Operating temperature range	0ºC to 70ºC

APPLICATIONS:

Non-inverting amplifier
Inverting amplifier
Integrator, Differentiator
Low Pass, High Pass, Band pass and Band Reject Filters

FEATURES:

No External frequency compensation is required
Short circuit Protection
Off Set Null Capability
Large Common mode and differential Voltage ranges
Low Power Dissipation
No-Latch up Problem
741 is available in three packages: 8-pin metal can, 10-pin flat pack and 8 or 14-pin DIP

IC555: (TIMER)

PIN CONFIGURATION:

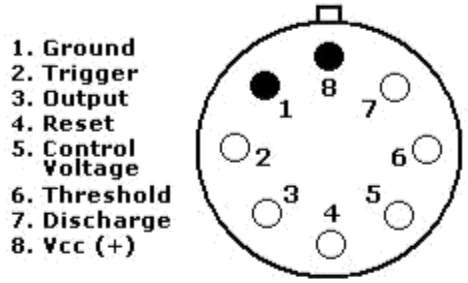


fig. 1. 8-pin T package

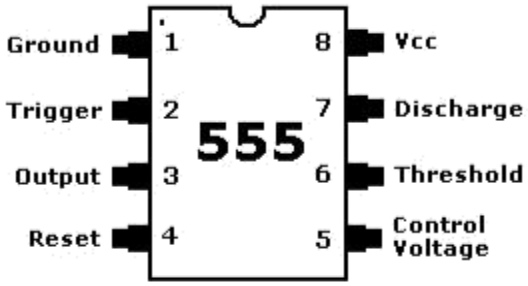
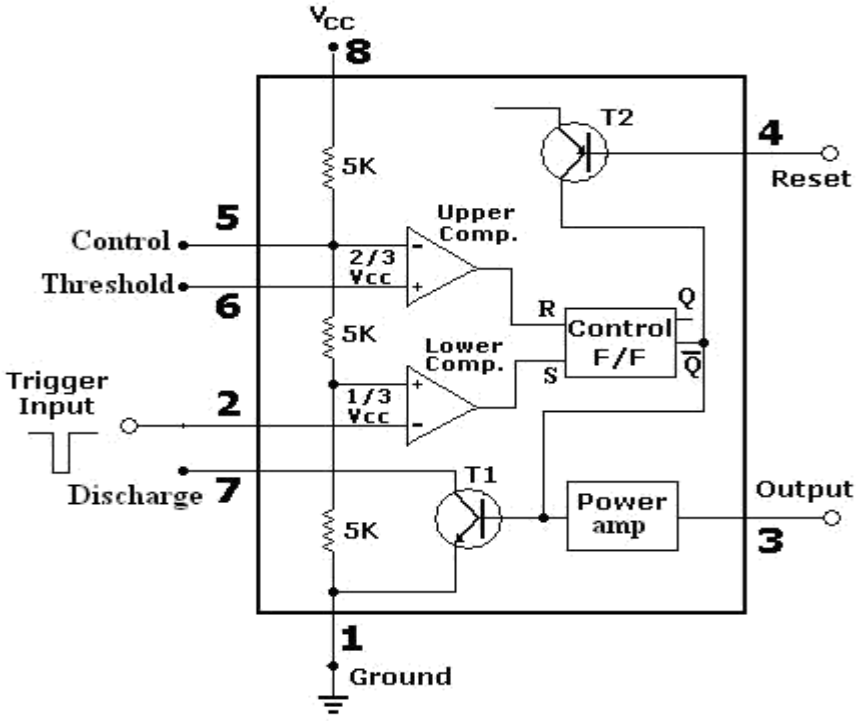


fig. 2. 8-pin V package

FUNCTIONAL BLOCK DIAGRAM:



SPECIFICATIONS

Supply Voltage	**5V to 18V**
Maximum Current rating	200mA
Minimum Triggering Voltage	**- (1/3) VCC**
Operating temperature range	0ºC to 70ºC

APPLICATIONS:

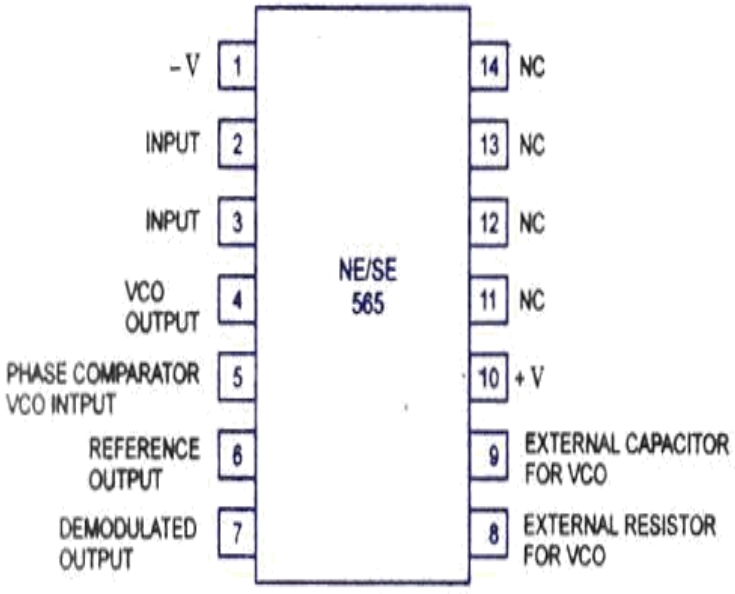
1. Astable Multivibrator, Schmitt trigger, Free running ramp Generator, etc.,
2. Monostable Multivibrator, Frequency divider, Pulse structure

FEATURES:

555 timers are reliable, easy to use and low cost. The device is available as an 8 pin circular style, an 8–pin mini DIP or a 14 Pin DIP

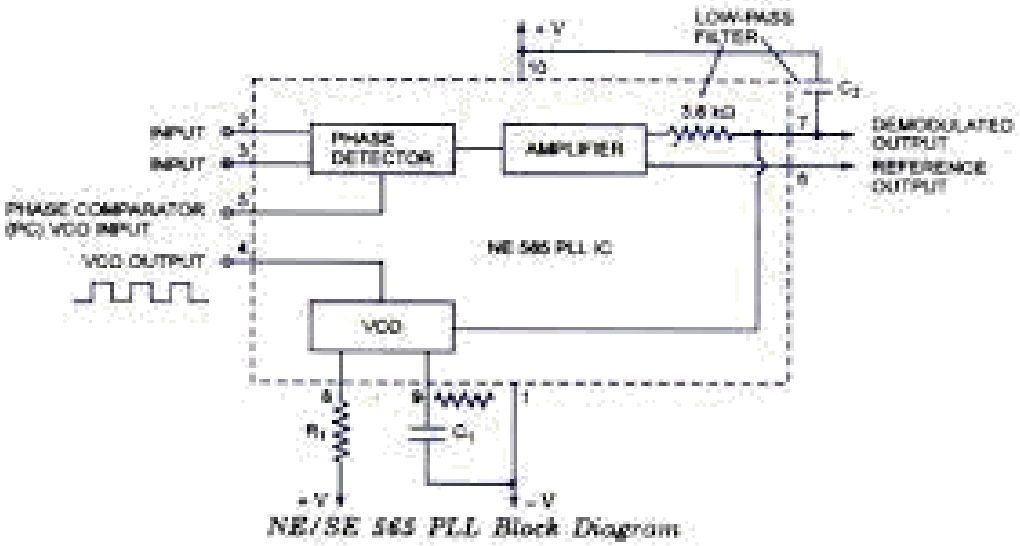
IC565: PHASE LOCKED LOOP (PLL)

PIN CONFIGURATION



14-Pin DIP Package

FUNCTIONAL BLOCK DIAGRAM



NE/SE 565 PLL Block Diagram

MONOLITHIC PLL CHARACTERISTICS

- Operating frequency range: 0.001 Hz to 500 kHz.
- Operating voltage range: ± 6 to ± 12 V.
- Input impedance: 10 k Q typically.
- Output sink current: 1mA typically.
- Output source current: 10 m A typically.
- Drift in VCO centre frequency with temperature: 300 ppm/ °C typically.
- Drift in VCO centre frequency with supply voltage: 1.5 %/V maximum.
- Input level required for tracking: 10 mVrms minimum to 3 V peak -to-peak maximum.
- Bandwidth adjustment range: < ± 1 to > ± 60 %.

APPLICATIONS:

1. Modems
2. FSK Demodulation
3. FM Demodulation
4. Frequency Synthasizers etc.

QUESTIONS:

1. What is the symbol of op-amp?
2. Draw the pin diagram of op-amp.
3. What is the supply voltage range that an op-amp can with stand?
4. What is the input voltage range that an op-amp can with stand?
5. What are the available package types of IC741?
6. What is a virtual ground? What are the differences between the physical ground and the virtual ground?
7. What is the current flowing through the input terminals of an Ideal op-amp?
8. Which loop voltage gain is larger, closed or open?
9. What is the normal value of saturation voltage of an op-amp?
10. Mention a few applications of op-amp.
11. Mention some features of op-amp.
12. What is the main purpose of IC555 timer?
13. Draw the pin diagram of op-amp.
14. Draw the functional diagram of IC555 timer.
15. How many comparators are present in IC555 timer?
16. What are the trigger voltages of UC and LC?
17. What is the functionality of power amplifier in the output stage of IC555 timer?
18. Which is the Flip-Flop used in IC555 timer?
19. What is the use of RESET pin in IC555 timer?
20. What are the available package types of IC555 timer?
21. Mention a few applications of IC555 timer.
22. What is the dc level required for the negative going trigger pulse at pin 2 of IC555 timer?
23. What is IC565?
24. Draw the pin diagram of IC565

EXPERIMENT NO: 1 **DATE:**

OP-AMP APPLICATIONS - ADDER, SUBTRACTOR & COMPARATOR

AIM: To study Adder, Subtractor & Comparator circuits using OP-AMP IC741 and verify their theoretical and practical output.

APPARATUS:
Bread Board
IC741, Resistors
DC Supply
Function Generator
Multi meter
CRO
Probes, Connecting Wires

THEORY:

ADDER: Op-amp can be used to design a circuit whose output is the sum of several input signals. Such a circuit is called a summing amplifier or an adder. Summing amplifier can be classified as inverting & non-inverting summer depending on the input applied to inverting & non-inverting terminals respectively. Circuit Diagram shows a non-inverting adder with n inputs. Here the output will be the linear summation of input voltages. The circuit can be used either as summing amplifier, scaling amplifier, or as averaging amplifier.

From the circuit of adder, it can be noted that at pin3

$I_1+I_2+I_3+\ldots\ldots\ldots\ldots I_n=0$

$$\frac{Va-V1}{R}+\frac{Va-V2}{R}+\frac{Va-V3}{R}+\ldots\ldots\ldots\frac{Va-Vn}{R}=0$$

$$\frac{nVa-(V1+V2+V3+\ldots+Vn)}{R}=0$$

$$V_a=\frac{V1+V2+V3+\cdots+Vn}{n}$$

$$V_o=\left(1+\frac{Rf}{R1}\right)Va$$

$$V_o=\left(1+\frac{Rf}{R1}\right)\left(\frac{V1}{n}+V2+V3+\ldots Vn\right)$$

$$V_o=\left(1+\frac{(n-1)R1}{R1}\right)\left(\frac{V1}{n}+V2+V3+\ldots Vn\right)$$

$$= (1+ (n-1)) \left(\frac{V1}{n} + V2 + V3 = \cdots Vn\right)$$

$$= n \left(\frac{V1}{n} + V2 + v3 + \ldots + Vn\right)$$

$$\mathbf{V_o = V_1 + V_2 + V_3 + \ldots + V_n}$$

This means that the output voltage is equal to the sum of all the input voltages.

SUBTRACTOR: A subtractor is a circuit that gives the difference of the two inputs, $V_o = V_2 - V_1$, Where V_1 and V_2 are the inputs. By connecting one input voltage V_1 to inverting terminal and another input voltage V_2 to the non – inverting terminal, we get the resulting circuit as the Subtractor. This is also called as differential or difference amplifier using op-amps.

Output of a differential amplifier (subtractor) is given as

$$V_o = (-R_f/R_1)(V_1 - V_2)$$

If all external resistors are equal in value, then the gain of the amplifier is equal to -1. The output voltage of the differential amplifier with a gain of -1 is

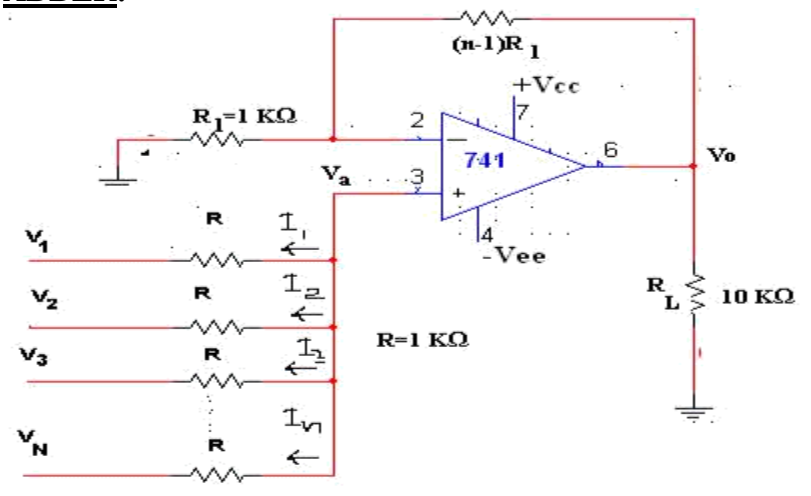
$$V_o = (V_2 - V_1)$$

Thus the output voltage V_o is equal to the voltage V_2 applied to the non – inverting terminal minus the voltage V_1 applied to the inverting terminal. Hence the circuit is called a Subtractor.

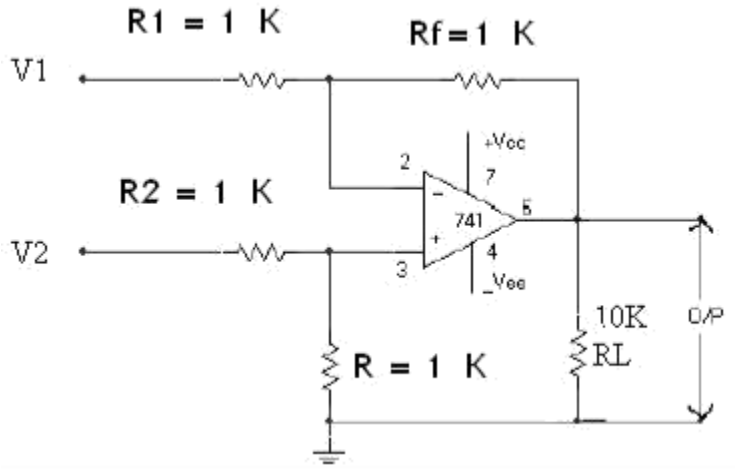
COMPARATOR: A Comparator is a non-linear signal processor. It is an open loop mode application of Op-amp operated in saturation mode. Comparator compares a signal voltage at one input with a reference voltage at the other input. Here the Op-amp is operated in open loop mode and hence the output is $\pm V_{sat}$. It is basically classified as inverting and non-inverting comparator. In a non-inverting comparator V_{in} is given to +ve terminal and V_{ref} to –ve terminal. When $V_{in} < V_{ref}$, the output is $-V_{sat}$ and when $V_{in} > V_{ref}$, the output is $+V_{sat}$ (see expected waveforms). In an inverting comparator input is given to the inverting terminal and reference voltage is given to the non inverting terminal. The output of the inverting comparator is the inverse of the output of non-inverting comparator. The comparator can be used as a zero crossing detector, window detector, time marker generator and phase meter.

CIRCUIT DIAGRAM:

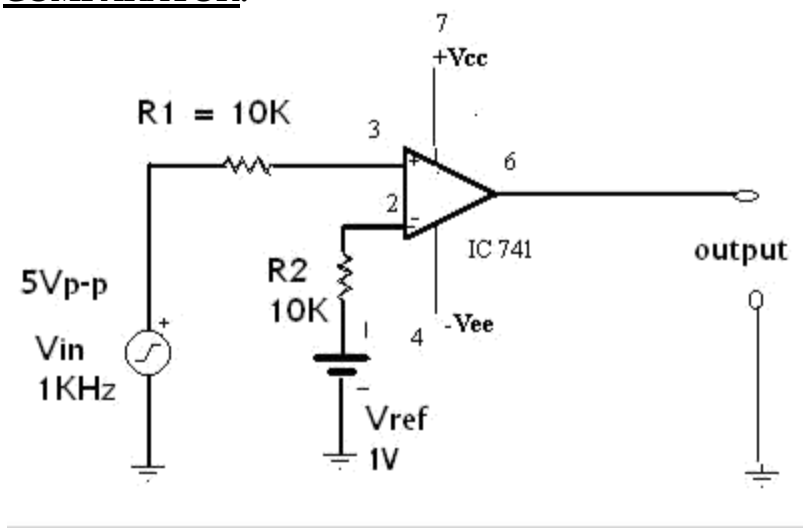
ADDER:



SUBTRACTOR:



COMPARATOR:



PROCEDURE:

ADDER:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Apply dc voltages at each input terminal for V_1 and V_2 from the dc supply and check the output voltage V_o at the output terminal.
4. Tabulate 3 different sets of readings by repeating the above step.
5. Compare practical V_o with the theoretical output voltage $V_o = V_1 + V_2$.

SUBTRACTOR:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Apply dc voltages at each input terminal for V_1 and V_2 from the dc supply and check the output voltage V_o at the output terminal.
4. Tabulate 3 different sets of readings by repeating the above step.
5. Compare practical V_o with the theoretical output voltage $V_o = V_2 - V_1$.

COMPARATOR:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Apply 1 KHz sine wave with 5 V_{pp} at the non-inverting input terminal of IC741 using a function generator.
4. Apply 1V dc voltage as reference voltage at the inverting terminal of IC741.
5. Connect the channel-1 of CRO at the input terminals and channel-2 of CRO at the output terminals.
6. Observe the input sinusoidal signal at channel-1 and the corresponding output square wave at channel-2 of CRO. Note down their amplitude and time period.
7. Overlap both the input and output waves and note down voltages at positions on sine wave where the output changes its state. These voltages denote the Reference voltage.
8. Plot the output square wave corresponding to the sine input with V_{ref} = 1V.

TABLE:

ADDER:

S.No.	V_1 Volts	V_2 Volts	Theoretical $V_o=V_1+V_2$	Practical V_o Volts

SUBTRACTOR:

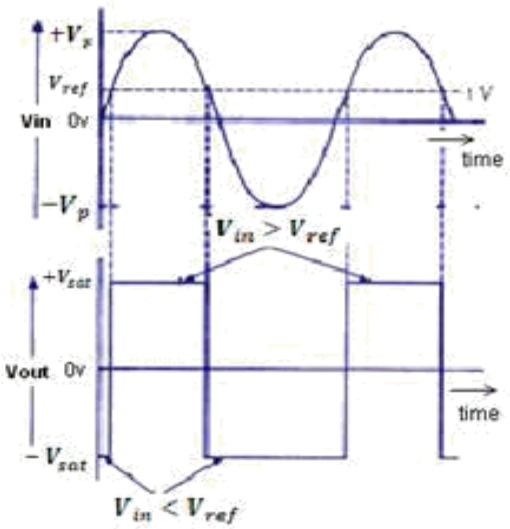
S.No.	V_1 Volts	V_2 Volts	Theoretical $V_o=V_2-V_1$	Practical V_o Volts

COMPARATOR:

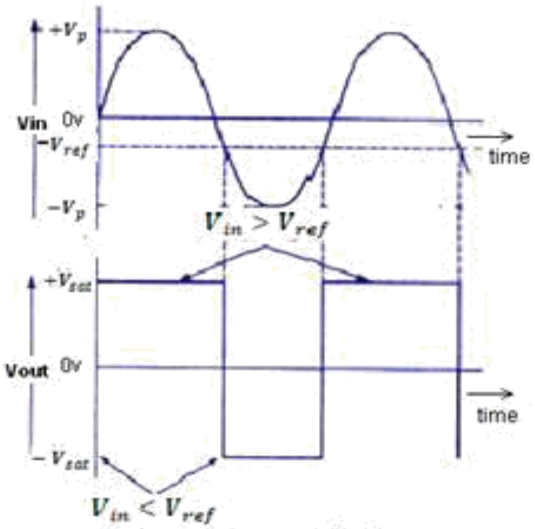
Theoretical Reference voltage (from circuit)	
Practical Reference voltage (from output waveforms)	

EXPECTED WAVEFORMS:

COMPARATOR INPUT & OUTPUT WAVEFORMS



(a) Input and output waveforms For Positive Vref.



(b) Input and output Waveforms For Negative Vref.

RESULT:

QUESTIONS:

1. Draw the circuit diagram of 3 input adder.
2. What is the other name for adder?
3. Draw the circuit diagram of a Subtractor.
4. Which amplifier acts as a Subtractor?
5. How many basic input parameters are required for a comparator?
6. Draw the circuit diagram of a non-inverting comparator and inverting comparator.
7. What is the output of a non-inverting comparator and inverting comparator if the input is sinusoidal?
8. What are the differences between the Inverting and Non–Inverting comparator?
9. What is the name of the comparator if the reference voltage is 0V?
10. Draw the circuit diagram and the output waveform of a Zero Crossing Detector if the input is sinusoidal?
11. What is the name of a regenerative comparator?
12. Draw an op- amp circuit whose output V_o is $V_1 + V_2 - V_3 - V_4$.

EXPERIMENT NO: 2 **DATE:**

INTEGRATOR AND DIFFERENTIATOR USING IC741 OP-AMP

AIM: To study the operation of the Integrator & differentiator using op-amp and trace the output wave forms for sine and square wave inputs.

APPARATUS:
Bread Board
IC741, Resistors, Capacitors
Function Generator
CRO
Probes
Connecting wires

THEORY:

INTEGRATOR:

A circuit in which the output voltage is the integration of the input voltage is called an integrator**.**

$$V_o = -\frac{1}{R_1 C_f} \int V_{in}\, dt \qquad \text{(1)}$$

In the practical integrator to reduce the error voltage at the output, a resistor R_F is connected across the feedback capacitor C_F. Thus, R_F limits the low-frequency gain and hence minimizes the variations in the output voltage.

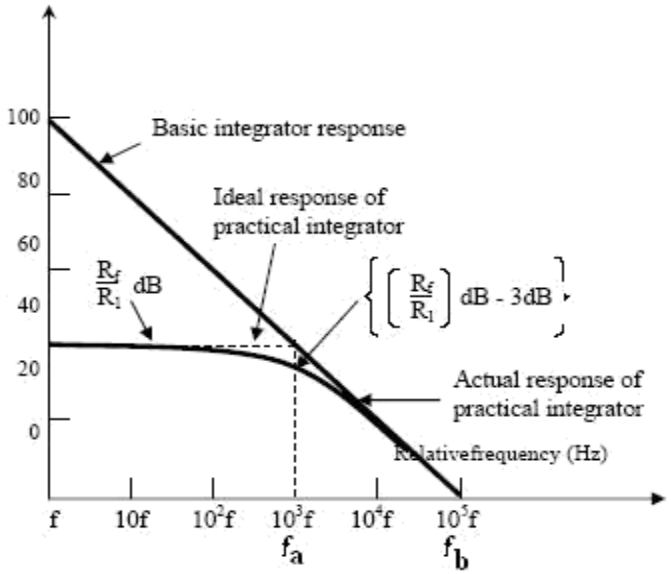


Fig 2.1 Frequency Response of Integrator

The frequency response of the integrator is shown in the fig. 2.1. f_b is the frequency at which the gain is 0 dB and is given by

$$f_b = 1/2\ \pi R_1 C_f.$$

In this fig. there is some relative operating frequency, and for frequencies from f to f_a the gain R_F/R_1 is constant. However, after f_a the gain decreases at a rate of 20 dB/decade. In other words, between f_a and f_b the circuit of fig. 2.1 acts as an integrator. The gain-limiting frequency f_a is given by

$$f_a = 1/2\ \pi R_f C_f.$$

Normally $f_a<f_b$. From the above equation, we can calculate R_f by assuming f_a & C_f. This is very important frequency. It tells us where the useful integration range starts.

If $f_{in} < f_a$ - circuit acts like a simple inverting amplifier and no integration
If $f_{in} = f_a$ results, - integration takes place with only 50% accuracy results, -
If $f_{in} = 10f_a$ integration takes place with 99% accuracy results.

In the circuit diagram of Integrator, the values are calculated by assuming f_a as 50 Hz. Hence the input frequency is to be taken as 500Hz to get 99% accuracy results.

Integrator has wide applications in
1. Analog computers used for solving differential equations in simulation arrangements.
2. A/D Converters
3. Signal wave shaping
4. Function Generators.

DIFFERENTIATOR:

As the name suggests, the circuit performs the mathematical operation of differentiation, i.e. the output voltage is the derivative of the input voltage.

$$V_o = - R_f C_1 \frac{dV_{in}}{dt}$$

Both the stability and the high-frequency noise problems can be corrected by the addition of two components: R_1 and C_f, as shown in the circuit diagram. This circuit is a practical differentiator.

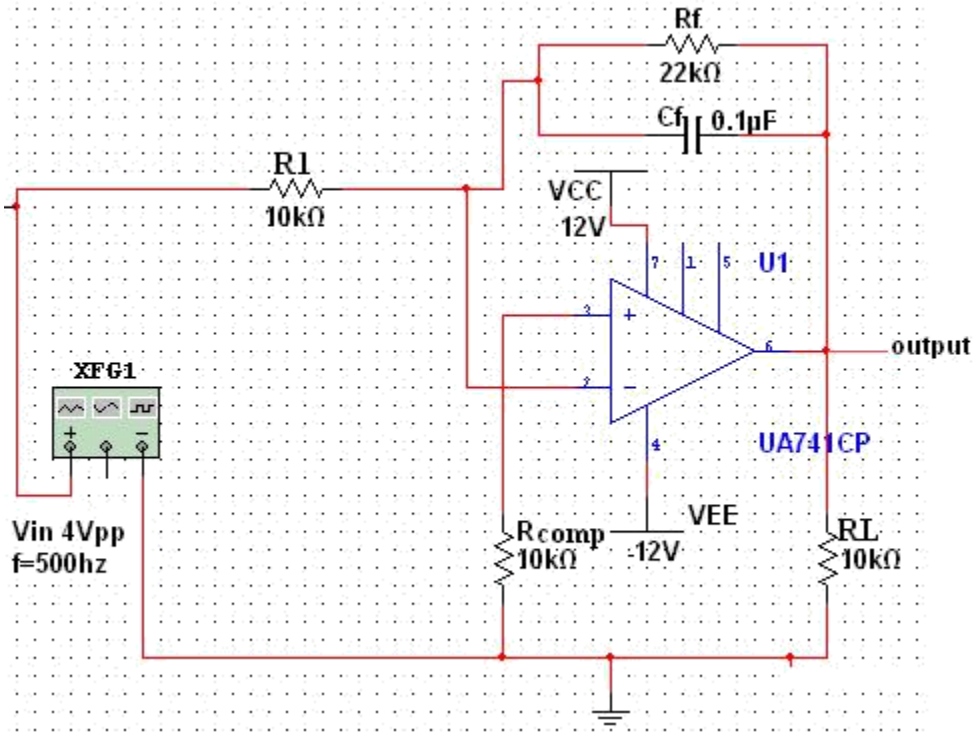
The input signal will be differentiated properly if the time period T of the input signal is larger than or equal to R_fC_1. That is, $T>= R_fC_1$

Differentiator can be designed by implementing the following steps.
1. Select f_a equal to the highest frequency of the input signal to be differentiated. Then, assuming a value of $C_1<1\ \mu$ F, calculate the value of R_f
2. Calculate the values of $R_1$and C_f so that $R_1C_1=R_fC_f$.

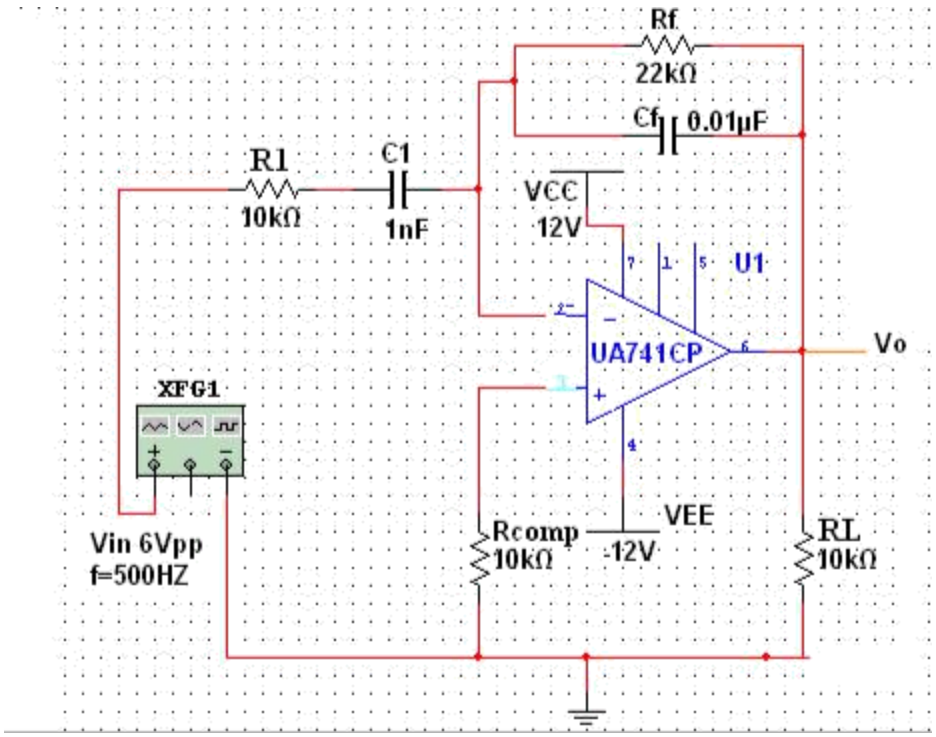
Differentiator has wide applications in
1. Monostable Multivibrator
2. Signal wave shaping
3. Function Generators.

CIRCUIT DIAGRAM:

INTEGRATOR:



DIFFERENTIATOR:



PROCEDURE:

INTEGRATOR:

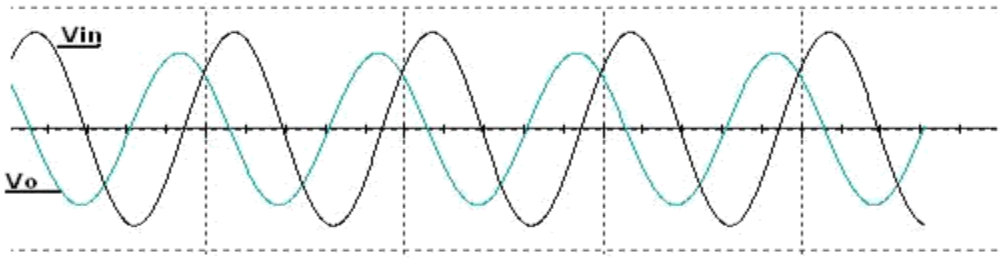
1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Apply sine wave at the input terminals of the circuit using function Generator.
4. Connect channel-1 of CRO at the input terminals and channel-2 at the output terminals.
5. Observe the output of the circuit on the CRO which is a cosine wave (90^o phase shifted from the sine wave input) and note down the position, the amplitude and the time period of V_{in} & V_o.
6. Now apply the square wave as input signal.
7. Observe the output of the circuit on the CRO which is a triangular wave and note down the position, the amplitude and the time period of V_{in} & V_o.
8. Plot the output voltages corresponding to sine and square wave inputs.

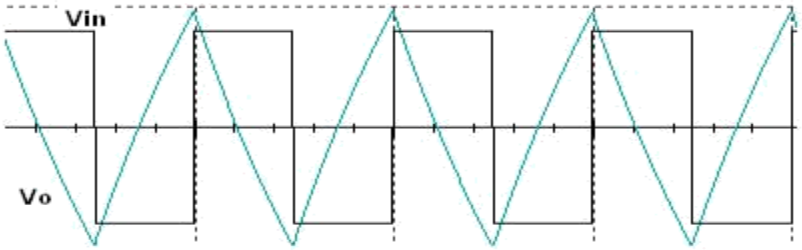
DIFFERENTIATOR:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Apply sine wave at the input terminals of the circuit using function Generator.
4. Connect channel-1 of CRO at the input terminals and channel-2 at the output terminals.
5. Observe the output of the circuit on the CRO which is a cosine wave (90^o phase shifted from the sine wave input) and note down the position, the amplitude and the time period of V_{in} & V_o.
6. Now apply the square wave as input signal.
7. Observe the output of the circuit on the CRO which is a spike wave and note down the position, the amplitude and the time period of V_{in} & V_o.
8. Plot the output voltages corresponding to sine and square wave inputs.

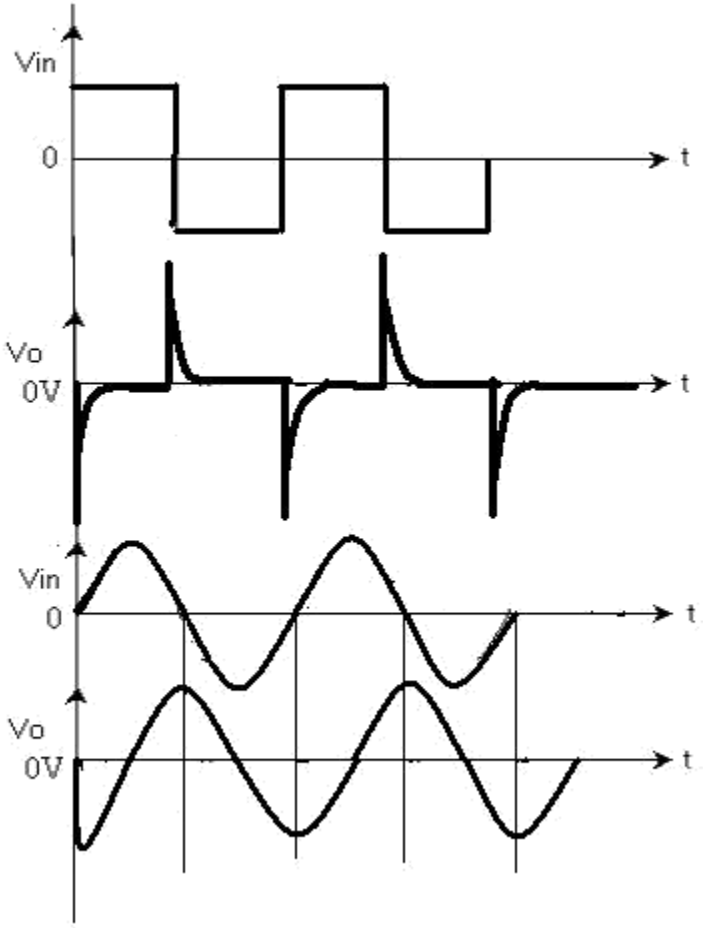
EXPECTED WAVEFORMS:

Integrator





DIFFERENTIATOR:



RESULT:

QUESTIONS:

1. What is an Integrator?
2. Draw the circuit of the Integrator using op-amp IC741.
3. Write down the expression for V_o of an Integrator.
4. Draw the frequency response of the Integrator and explain.
5. Draw the output waveform of the Integrator when the input is a Square wave.
6. What is the purpose behind the connection of Rf in the feedback path of Integrator?
7. What are the applications of Integrator?
8. Why R_{comp} is used in both Integrator and Differentiator circuits?
9. What is a Differentiator?
10. Draw the circuit of the Differentiator using op-amp IC741.
11. Write down the expression for V_o of a Differentiator.
12. Draw the output waveform of the Differentiator when the input is a Sine wave.
13. Why R1 and Cf are connected in the circuit of the Differentiator?
14. What are the applications of Differentiator?

EXPERIMENT NO: 3 **DATE:**

ACTIVE FILTER APPLICATIONS - LPF & HPF (1ST ORDER)

A) 1st ORDER LOW PASS FILTER

AIM: To plot the frequency response of Butterworth LPF (First order) and find the high cut-off frequency.

APPARATUS:
Bread Board
Function Generator
CRO
Probes
Connecting Wires
741 Op-amp, Resistors, Capacitors

THEORY:

Filters are classified as follows:

Based on components used in the circuit

- Active filters – Use active elements like transistor or op-amp(provides gain) in addition to passive elements
- Passive filters – Use only passive elements like resistors, capacitors and inductors, hence no gain here.

Based on frequency range

- Low pass filter(LPF) – Allows low frequencies
- High pass filter(HPF) – Allows high frequencies
- Band pass filter(BPF) – Allows band of frequencies
- Band reject filter(BRF) – Rejects band of frequencies

All pass filter – Allows all frequencies but with a phase shift

Active Filter is often a frequency – selective circuit that passes a specified band of frequencies and blocks or attenuates signals of frequencies outside this band.

These Active Filters are most extensively used in the field of communications and signal processing. They are employed in one form or another in almost all sophisticated electronic systems such as Radio, Television, Telephone, Radar, Space Satellites, and Bio-Medical Equipment.

Active Filters employ transistors or Op – Amps in addition to that of resistors and capacitors. Active filters have the following advantages over passive filters. (1) Flexible gain and frequency adjustment. (2) No loading problem (because of high input impedance and low output impedance) and (3) Active filters are more economical than passive filters.

A first – Order Low – Pass Butterworth filter uses RC network for filtering. Note that the op-amp is used in the non-inverting configuration; hence it does not load down the RC network. Resistors R_1 and R_F determine the gain of the filter.

The gain magnitude equation of the Low – Pass filter can be obtained by converting equation into its equivalent polar form, as follows.

$$| V_o / V_{in} | = A_F / \sqrt{1 + (f / f_H)^2}$$

Where $f_H = \frac{1}{2 \prod RC}$ = high cut-off frequency of the filter.

The operation of the low – pass filter can be verified from the gain magnitude equation. 1. At very low frequencies, that is $f < f_H$

$| V_o/V_{in} | = A_F$

2. At $f = f_H$, $| V_o/V_{in} | = A_F/\sqrt{2} = 0.707\ A_F$

3. At $f > f_H$ $| V_o/V_{in} | < A_F$

Thus the Low – Pass filter has a constant gain A_F from 0 Hz to the almost high cut-off frequency, f_H, it has the gain $0.707A_F$ at exactly f_H, and after f_H it decreases at a constant rate with an increase in frequency. The gain decreases 20 dB (= 20 log 10) each time the frequency is increased by 10. Hence the rate at which the gain rolls off after f_H is 20 dB/decade. The frequency $f = f_H$ is called the cut-off frequency because the gain of the filter at this frequency is down by 3 dB (=20log 0.707) from 0 Hz. Other equivalent terms for cut-off frequency are -3dB frequency, break frequency, or corner frequency.

DESIGN:

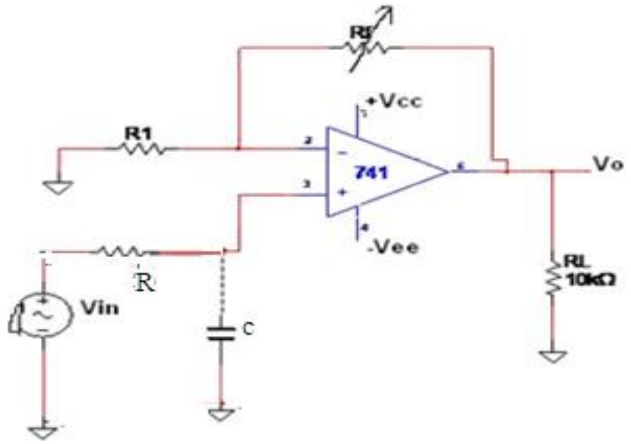
1. Choose a value for high cut-off frequency, f_H(1 KHz) and a value for gain, A_F (2)
2. Assume a value of C≤1μF (0.1 μF)
3. Calculate the value of R using the equation $R = \frac{1}{2\pi C f_H}$
4. Finally, select values of R_1 and R_F dependent on the desired pass band gain A_F

   using $A_F = 1 + R_F/R_1$

   $2 = 1 + R_F/R_1$

   $R_F = R_1$
5. Assume a value for R_1 (10KΩ) and calculate R_F.

CIRCUIT DIAGRAM:



(You can assume any value for C which is available in the Lab)

PROCEDURE:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Connect channel -1 of CRO to input terminals (V_{in}) and channel -2 to output terminals (V_o).
4. Set V_{in} = 1V & f_{in}=10Hz using function generator.
5. By varying the input frequency in regular intervals, note down the output voltage.
6. Calculate the gain (V_o/V_{in}) and Gain in dB = 20 log (V_o/V_{in}) at every frequency.
7. Plot the frequency response curve (taking frequency on X-axis & Gain in dB on Y-axis) using Semi log Graph.
8. Find out the high cut-off frequency, f_H (at Gain= Constant Gain, A_f – 3 dB) from the frequency response plotted.
9. Verify the practical (f_H from graph) and the calculated theoretical cut-off frequency (f_H = $1/2\pi RC$).

TABLE:

V_{in} = 1V

S.No.	Input Frequency f(Hz)	Output Voltage V_o (V)	Gain Magnitude $\lvert V_o/V_{in} \rvert$	Gain in dB = $20\log\lvert V_o/V_{in} \rvert$

CALCULATIONS:

THEORETICAL Cut-off frequency:

$f_H = 1 / (2\pi RC)$ = high cut-off frequency of the Low pass filter.

=

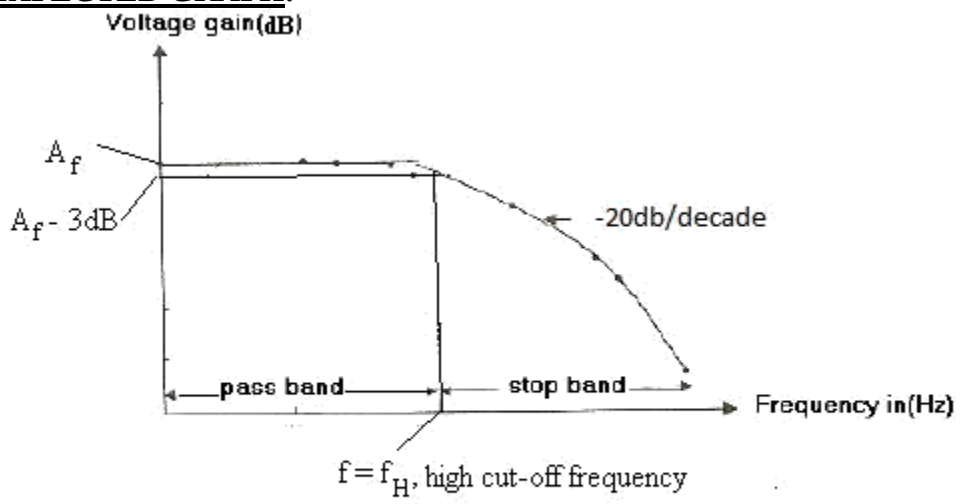
PRACTICAL Cut-off frequency (from Graph) :

f_H = high cut-off frequency of the Low pass filter

= 3dB cut-off frequency

=

EXPECTED GRAPH:



RESULT:

ACTIVE LOW PASS & HIGH PASS BUTTERWORTH FILTERS (1st ORDER).

B) 1st ORDER HIGH PASS FILTER

AIM: To plot the frequency response of Butterworth HPF (First order) and find the low cut-off frequency.

APPARATUS:
Bread Board
Function Generator
CRO
Probes
Connecting Wires
741 Op-amp, Resistors, Capacitors

THEORY:

First Order High Pass Filter consists of RC network for filtering. First Order High Pass filter can be constructed from a First Order Low Pass filter simply by interchanging frequency determining components R & C . Op-Amp is used in the non – inverting configuration. Resistor R_1 and R_F determine the gain of the Filter.

The voltage gain magnitude equation of the second order High-pass filter is

$$\left|\frac{V_0}{V_{in}}\right| = \frac{A_F\,(f/f_L)}{\sqrt{[1+(f/f_L)^2]}}$$

where $A_F = 1 + R_F / R_1$

f = Operating (input) frequency.

$f_L = \frac{1}{2\pi RC}$ = Low cut-off frequency of the filter.

This is the frequency at which the magnitude of the gain is 0.707 times its pass band value. Obviously, all frequencies higher than f_L are Pass Band frequencies, with the highest frequency determined by the closed-loop bandwidth of the OP-Amp.

The operation of the high–pass filter can be verified from the gain magnitude equation. 1. At very low frequencies, that is $f < f_L$

$| V_o/V_{in} | < A_F$

2. At $f = f_L$, $| V_o/V_{in} | = A_F/\sqrt{2} = 0.707\ A_F$

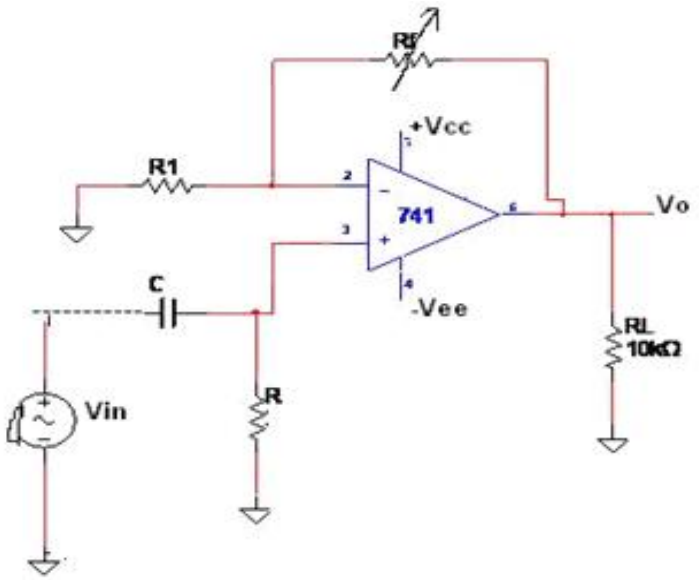
3. At $f > f_L$, $| V_o/V_{in} | = A_F$

For example, in the first order High – Pass filter the gain rolls – off or increases at the rate of 20dB/decade in stop band, that is for input signal frequency lesser than Low cut-off frequency (f_L) ;

High Pass filter has constant gain A_F, after the Low cut-off frequency onwards (f_L).

DESIGN: Follow the same procedure as given for low-pass filter.

CIRCUIT DIAGRAM:



PROCEDURE:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Connect channel -1 of CRO to input terminals (V_{in}) and channel -2 to output terminals (V_o).
4. Set V_{in} = 1V & f_{in}=10Hz using function generator.
5. By varying the input frequency in regular intervals, note down the output voltage.
6. Calculate the gain (V_o/V_{in}) and Gain in dB = 20 log(V_o/V_{in}) at every frequency.
7. Plot the frequency response curve (taking frequency on X-axis & Gain in dB on Y-axis) using Semi log Graph.
8. Find out the low cut-off frequency, f_L (at Gain= Constant Gain, A_f – 3 dB) from the frequency response plotted.
9. Verify the practical (f_L from graph) and the calculated theoretical cut-off frequency (f_L = $1/2\pi RC$).

TABLE:

V_{in} = 1V

S.No.	Input Frequency f(Hz)	Output Voltage V_o (V)	Gain Magnitude $\lvert V_o/V_{in} \rvert$	Gain in dB = $20\log\lvert V_o/V_{in} \rvert$

CALCULATIONS:

THEORETICAL Cut-off frequency:

$f_L = 1 / (2\pi RC)$ = Low cut-off frequency of the HPF.

=

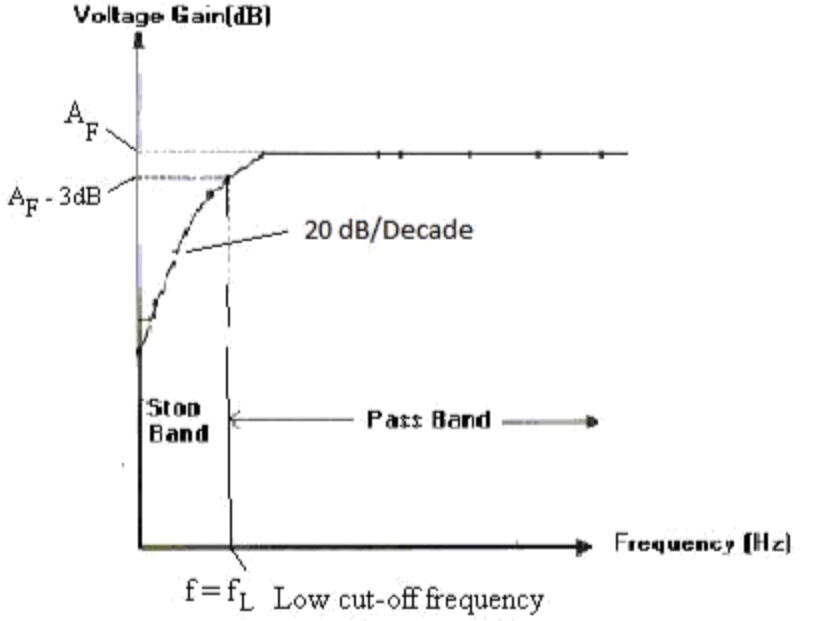
PRACTICAL Cut-off frequency:

f_L = Low cut-off frequency of the HPF.

= 3dB cut-off frequency

=

EXPECTED GRAPH:



RESULT:

QUESTIONS:

1. How filters are classified? Give one example for each classification.
2. What is an active filter and why it is called so?
3. How an active filter differs from a passive filter?
4. What are the advantages of active filters over passive filters?
5. Draw the circuit diagrams of active filters LPF and HPF.
6. Draw the frequency response of all filters (LPF, HPF, BPF, BRF and All-pass).
7. What is the gain roll off rate for a 1^{st} order and 2^{nd} order filter?
8. What is the formula for cut-off frequency?
9. What is a 3 dB frequency and why it is called so?
10. What are the other names for 3 dB frequency?

EXPERIMENT NO: 4 **DATE:**

IC 741 WAVEFORM GENERATORS – SINE, SQUAREWAVE AND TRIANGULAR WAVES

AIM: To design a Waveform Generator which generates Sine, Square and Triangular waveforms using IC741 and to verify it's various output waveforms.

APPARATUS: Bread Board
CRO
Probes
741 Op-amp, Resistors, Capacitors

THEORY:

Waveform generator using IC741 is a circuit which generates Sine wave, Square wave and Triangular wave. This circuit is a combination of Wien Bridge oscillator, Zero crossing detector (Comparator with zero reference voltage) and Integrator. The Wien Bridge oscillator generates Sine wave which is fed to the input of Zero crossing detector. This detector gives the square wave output which is connected to the input of the Integrator which in turn produces the Triangular wave output.

The frequency of oscillations of the Sine wave output of Wien Bridge oscillator is given by

$$f_o = 1/2\pi RC$$

The frequency of oscillations of Square and Triangular wave outputs will also be the same frequency as that of the Sine wave output.

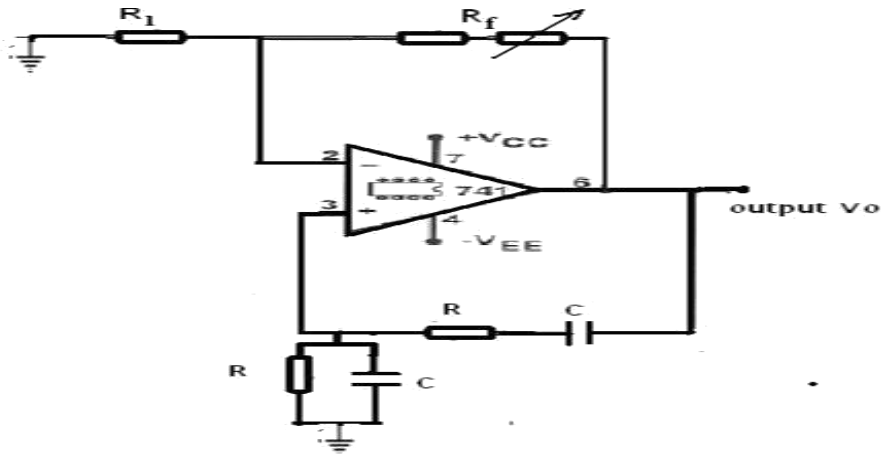
For theory of individual circuits i.e. Wien Bridge oscillator, Zero Crossing Detector and Integrator, please refer to the THEORY section of respective experiments mentioned earlier in this manual.

DESIGN FOR WIEN BRIDGE OSCILLATOR:

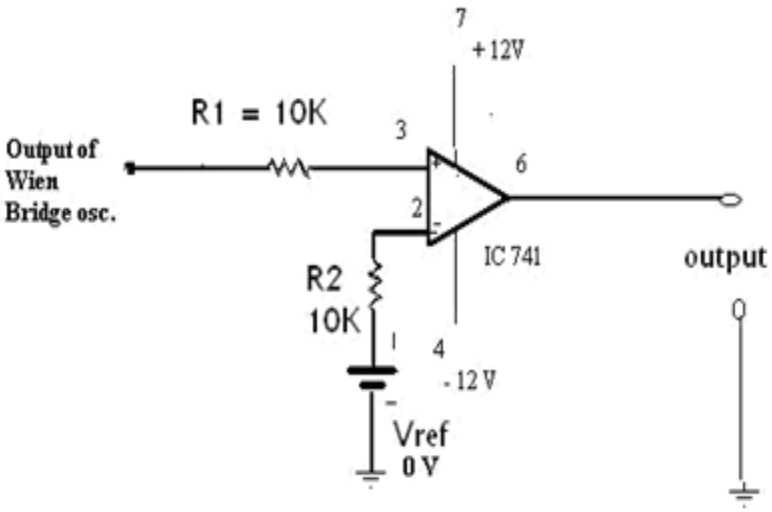
1. Choose a desired frequency of oscillation, say f_o =500 Hz.
2. Choose a value for capacitor C (0.1 μF) and then calculate the value of R by using the equation for f_o ($f_o = 1/2\pi RC$).
3. Choose a value for R_1 (10 KΩ) and calculate the value of R_f from the gain equation ($A_v = 1+R_f/R_1 = 3$). (Note: In practical, the value of R_f may need to be varied to be more than the calculated value.)

CIRCUIT DIAGRAM:

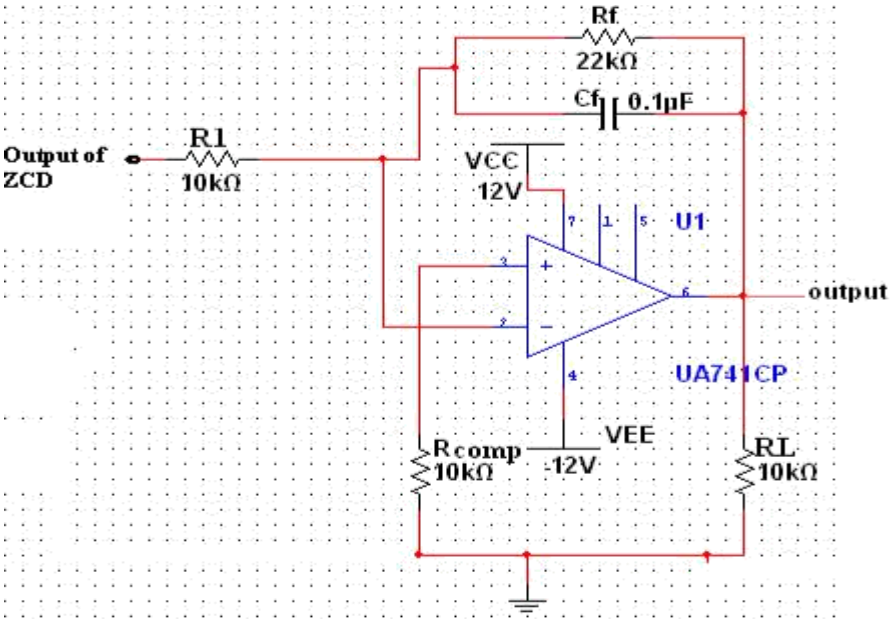
SINE WAVE GENERATOR (WIEN BRIDGE OSCILLATOR):



SQUARE WAVE GENERATOR (ZERO CROSSING DETECTOR):



TRIANGULAR WAVE GENERATOR (INTEGRATOR):



PROCEDURE:

SINE WAVE GENERATOR:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Connect output to the CRO.
4. Adjust the potentiometer to get an undistorted waveform.
5. Note down the amplitude and the time period, T of the sine wave and calculate the frequency of oscillation, $f_o = 1 / T$.
6. Verify the practical frequency of oscillation calculated in the preceding step with the theoretical value, $f_o = 1/2\pi RC$.
7. Plot the waveform.

SQUARE WAVE GENERATOR:

1. Switch OFF the power supply.
2. Connect the components/equipment as shown in the circuit diagram.
3. Switch ON the power supply.
4. Connect the input to the channel-1 of CRO and output to the channel-2 of CRO.
5. Observe the square wave output at channel-2 and note down the amplitude and time period, T of the wave form.
6. Verify that the frequency of oscillation of both the input and the output waves is same. Also verify that both the input and the output waves are in same phase.
7. Plot the output waveform in accordance with the input waveform.

TRIANGULAR WAVE GENERATOR:

1. Switch OFF the power supply.
2. Connect the components/equipment as shown in the circuit diagram.
3. Switch ON the power supply.
4. Connect the input to the channel-1 of CRO and output to the channel-2 of CRO.
5. Observe the triangular wave output at channel-2 and note down the amplitude and time period, T of the wave form.
6. Verify that the frequency of oscillation of both the input and the output waves is same. Also verify that the output wave is inverted i.e. 180^{o} phase shift from the input wave.
7. Plot the output waveform in accordance with the input waveform.

CALCULATIONS:

THEORETICAL Frequency of Oscillation

$f_o = 1/2\pi RC$

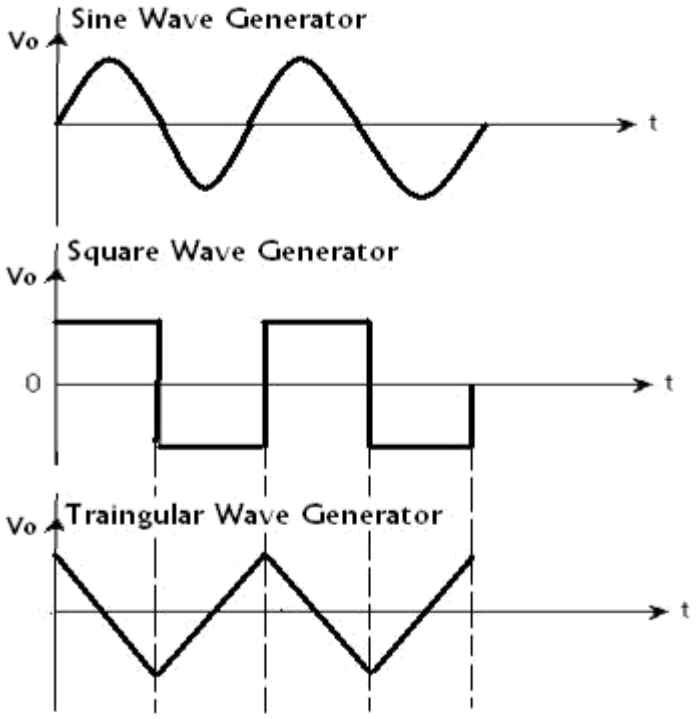
=

PRACTICAL Frequency of Oscillation

$f_o = 1/T$

=

EXPECTED WAVEFORMS:



RESULT:

QUESTIONS:

1. What is a Function Generator?
2. What are the different stages in a Function Generator and how they are connected?
3. Draw the output waveforms at different stages of Function Generator.
4. What is the relationship among the frequencies of output waveforms at different stages of Function Generator?
5. Will there be any phase shift between the input and the output of any stage in the Function Generator and what factor it depends on?
6. Why is R_{comp} used in the circuit of Triangular wave generator?
7. Why is potentiometer used in the circuit of Wien Bridge Oscillator?

EXPERIMENT NO: 5 **DATE**

IC555 TIMER – MONOSTABLE & ASTABLE MULTIVIBRATOR CIRCUITS

A) MONOSTABLE MULTIVIBRATOR

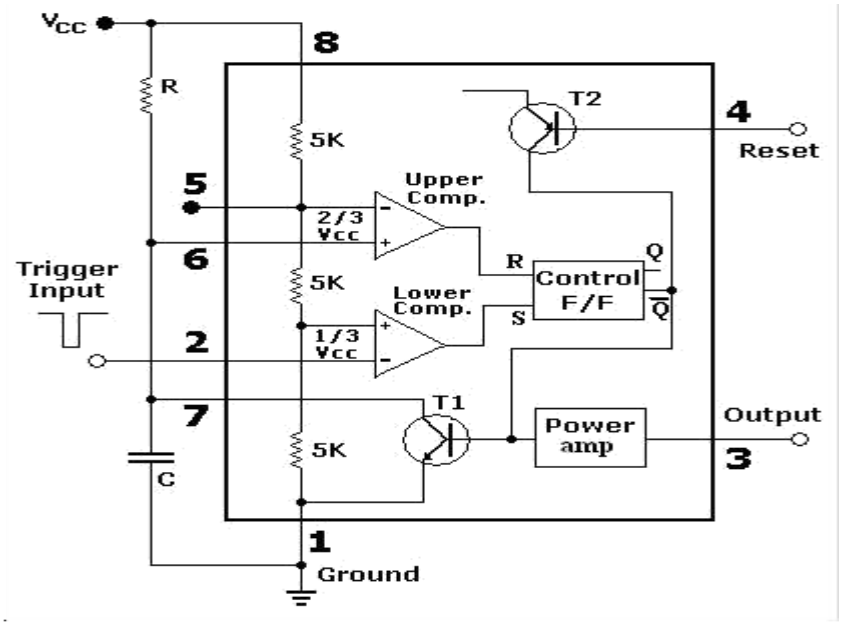
AIM: To design a Monostable Multivibrator using IC555 and compare it's theoretical and practical pulse width.

APPARATUS: Bread Board.
CRO
Probes
Connecting wires
555 Timer, Resistors, Capacitors

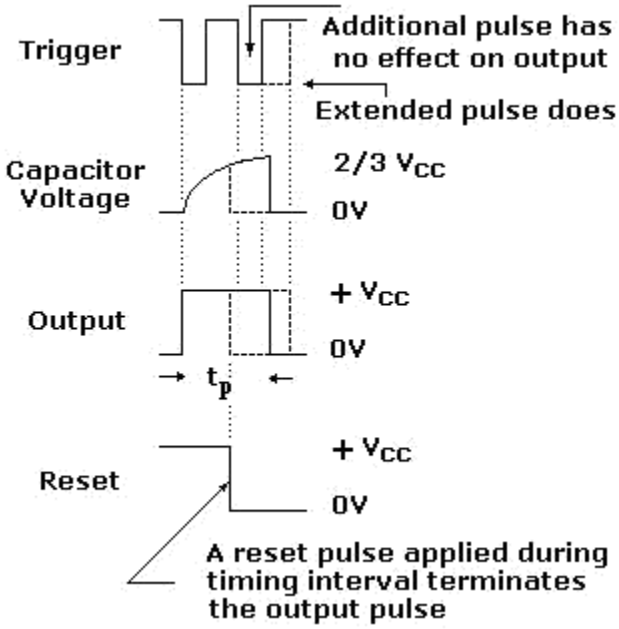
THEORY:

Monostable multivibrator is also called as one–shot Multivibrator. When the output is low, the circuit is in stable state, transistor T1 is ON and Capacitor C is shorted to the ground. However, upon application of a negative trigger pulse to Pin–2, transistor T1 is turned OFF, which releases short circuit across the external capacitor and drives the output High. The capacitor C now starts charging up toward V_{CC} through R. However when the voltage across the external capacitor equals 2/3 $V_{CC,}$ upper comparator's output switches from low to high which in turn derives the output to its low state. And the output of the flip flop turns transistor T1 ON, and hence the capacitor C rapidly discharges through the transistor. The output of the Monostable remains low until a trigger pulse is again applied. Then the cycle repeats. The time during which the output remains high is given by

$$t_p = 1.1\ R\ C$$



Waveforms for IC555 Monostable Multivibrator

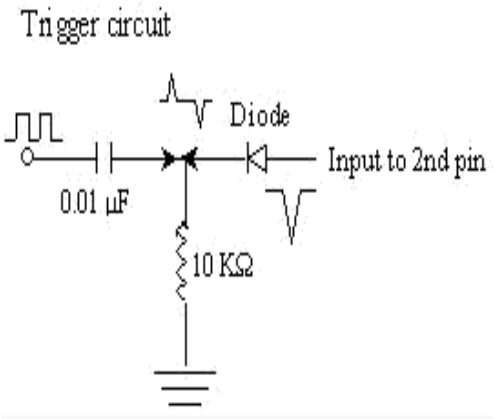


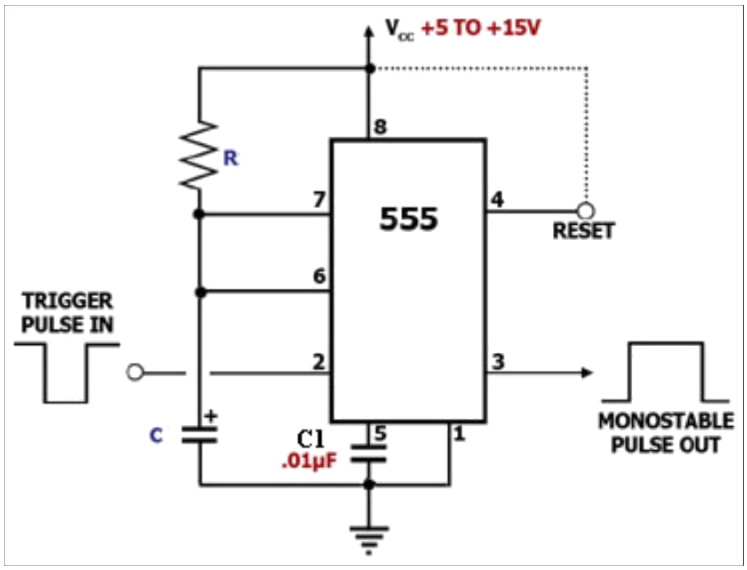
Once triggered, the circuit's output will remain in the high state until the set time t_p elapses. The output will not change its state even if an input trigger is applied again during this time interval t_p.

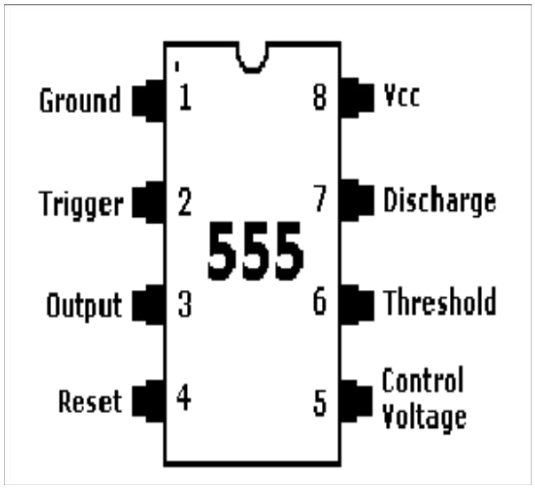
DESIGN:

1. Choose a desired pulse width, say t_p =1.1 ms.
2. Choose a value for capacitor C (0.1 µF) and then calculate the value of R by using the equation for t_p.

CIRCUIT DIAGRAM:







PROCEDURE:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Connect function generator at the trigger input.
4. Connect channel-1 of CRO to the trigger input and channel-2 of CRO to the output (Pin 3).
5. Using Function Generator, apply 1 KHz square wave with amplitude of approx. equal to 9 V_{pp} at the trigger input.
6. Observe the output voltage with respect to input and note down the pulse width and amplitude.
7. Now connect channel-2 of CRO across capacitor and observe the voltage across the capacitor and note it down.
8. Compare the practical pulse width noted in the step above with its theoretical value (t_p=1.1 RC)

CALCULATIONS:

THEORETICAL Pulse width

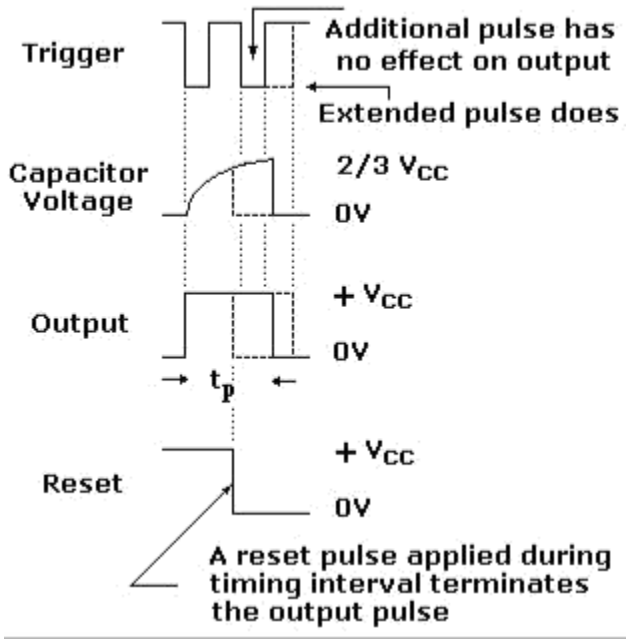
R = C =

t_p = 1.1 RC =

PRACTICAL Pulse width

t_p =

EXPECTED WAVEFORMS:



RESULT:

QUESTIONS:

1. What is the other name for monostable multivibrator (MSMV)?
2. When MSMV is in stable state, what is the output level?
3. Why trigger is required in the case of MSMV?
4. Which type of trigger pulse is required for MSMV?
5. What is the formula for the output pulse width of MSMV?
6. How long MSMV stays in unstable state?

(B) ASTABLE MULTIVIBRATOR

AIM: To design an Astable Multivibrator using IC555 and compare it's theoretical and practical time period and duty cycle.

APPARATUS: Bread Board.
CRO
Probes
Connecting wires
555 Timer, Resistors, Capacitors

THEORY:

An Astable multivibrator, often called a free-running Multivibrator, is a rectangular-wave-generating circuit. Unlike the Monostable multivibrator, this circuit does not require an external trigger to change the state of the output, hence the name free running. However, the time during which the output is either high or low is determinate by the Two resistors and a capacitor, which are externally connected to the 555 timer.

Figure 1 shows the 555 timer connected as an Astable multivibrator. Initially, when the output is high, capacitor C starts charging towards V_{cc} through R_A and R_B. However as soon as voltage across the capacitor equals 2/3 V_{cc}, comparator 1 triggers the flip-flop, and the output switches low. Now the capacitor C starts discharging through R_B and the transistor Q_1. When the voltage across C equals 1/3 V_{cc}, comparator 2's output triggers the flip-flop, and the output goes high. Then the cycle repeats. The output voltage and the capacitor voltage waveforms are shown in the following figures.

As shown in this figure, the capacitor is periodically charged and discharged between 2/3 V_{cc} and 1/3 V_{cc}, respectively. The time during which the capacitor charges from 1/3 V_{cc} to 2/3 V_{cc} is equal to the time the output is high and is given by

$$t_c = 0.69 (R_A + R_B) C \qquad (1)$$

Similarly, the time during which the capacitor discharges from 2/3 V_{cc} to 1/3 V_{cc} is equal to the time the output is low and is given by

$$t_d = 0.69 (R_B)C \qquad (2)$$

Thus the total time period of the waveform is

$$T = t_c + t_d = 0.69(R_A + 2R_B) \qquad (3)$$

Therefore the frequency of oscillation is $f_o = 1/T = 1.45/(R_A + 2R_B)C$

And $$\% \text{ Duty cycle} = (t_c/T) *100 \qquad (4)$$

CIRCUIT DIAGRAM:

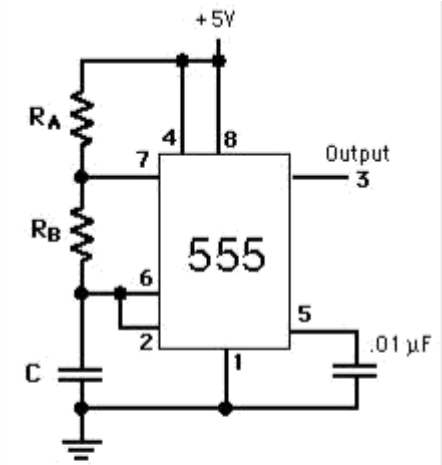
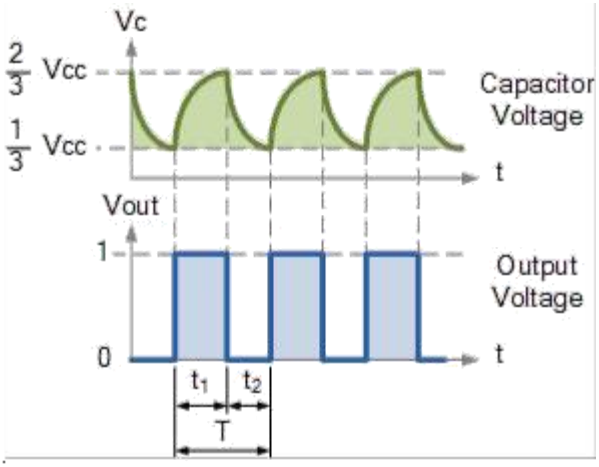


Figure 1: The 555 as an Astable Multivibrator

EXPECTED WAVE FORMS:



PROCEDURE:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.
3. Connect channel-1 of CRO to the output (Pin 3).
4. Observe the output voltage and note down the time period and duty cycle.
5. Now connect channel-2 of CRO across capacitor and observe the voltage across the capacitor and note it down.
6. Compare the practical time period and duty cycle.

CALCULATIONS:

THEORETICAL time periods

$t_c = 0.69 (R_A + R_B) C$

$t_d = 0.69 (R_B)C$

Total time period of the waveform, $T = t_c + t_d$

% Duty Cycle = $(t_c / T) *100$

PRACTICAL (from output waveforms)

time period, T =

% Duty cycle =

RESULT:

\

QUESTIONS:

1. What is the other name for Astable multivibrator (AMV)?
2. What is the formula for the time period of the waveform of AMV?
3. What is the formula for the % of Duty cycle?

EXPERIMENT NO: 6 **DATE**

SCHMITT TRIGGER CIRCUIT USING IC741

AIM: To study the Schmitt trigger characteristics by using IC741 and compare theoretical and practical values of the Upper Threshold voltage, V_{UT} and the Lower Threshold voltage, V_{LT}.

APPARATUS:
741 Op-Amp
Resistors
Bread board
Function generator
CRO
Probes
Connecting wires

THEORY:

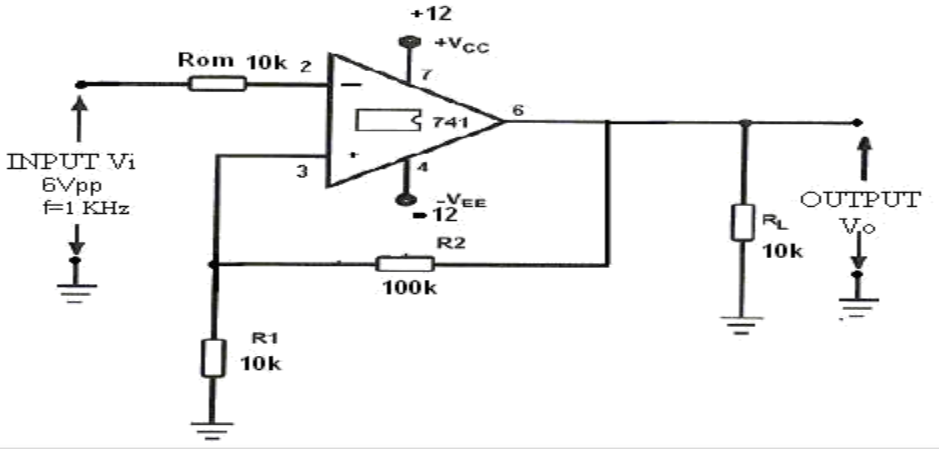
Circuit shows an inverting comparator with positive feedback. This circuit converts an irregular shaped waveform to square wave or pulse. This circuit is known as Schmitt trigger or Regenerative comparator or Squaring circuit. The input voltage V_{in} triggers (changes the state of) the output V_o every time it exceeds certain voltage levels called Upper threshold voltage, V_{UT} and Lower threshold voltage, V_{LT}. The hysteresis width is the difference between these two threshold voltages i.e. $V_{UT} - V_{LT}$. These threshold voltages are calculated as follows.

$V_{UT} = (R_1/R_1+R_2)\ V_{sat}$ when $V_o = V_{sat}$

$V_{LT} = (R_1/R_1+R_2)\ (-V_{sat})$ when $V_o = -V_{sat}$

The output of Schmitt trigger is a square wave when the input is sine wave or triangular wave, where as if the input is a saw tooth wave then the output is a pulse wave.

CIRCUIT DIAGRAM:

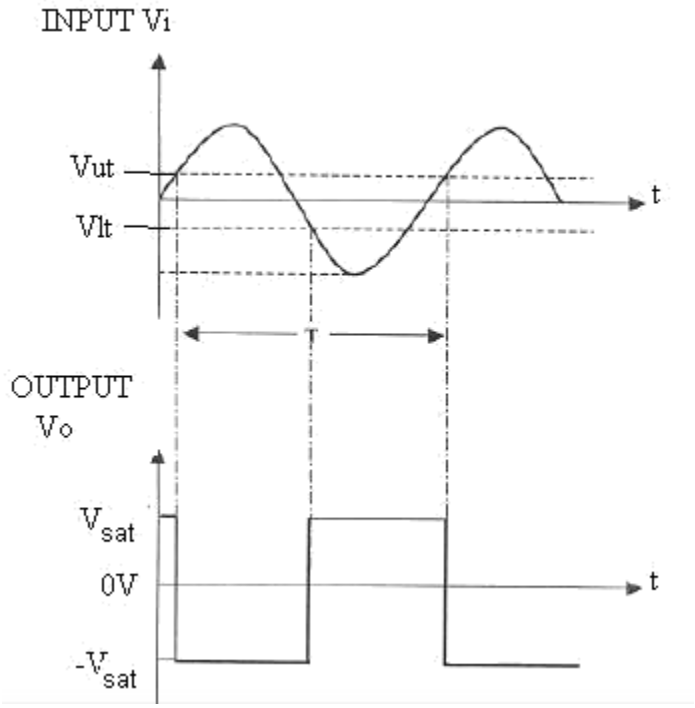


PROCEDURE:

1. Connect the components/equipment as shown in the circuit diagram.
2. Switch ON the power supply.

3. Apply the input sine wave using function generator.
4. Connect the channel–1 of CRO at the input terminals and Channel-2 at the output terminals.
5. Observe the output square waveform corresponding to input sinusoidal signal.
6. Overlap both the input and output waves and note down voltages at positions on sine wave where output changes its state. These voltages denote the Upper threshold voltage and the Lower threshold voltage (see EXPECTED WAVEFORMS below).
7. Verify that these practical threshold voltages are almost same as the theoretical threshold voltages calculated using formulas given in the THEORY section above.
8. Sketch the waveforms by noting down the amplitude and the time period of the input V_{in} and the output V_o.

EXPECTED WAVEFORMS:



Vo versus Vin plot of Hysterisis Voltage

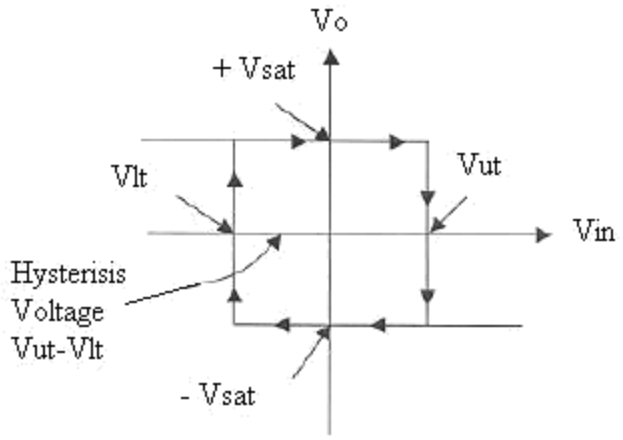


TABLE:

S.No	Theoretical Values				Practical value	
	R_1	R_2	$V_{ut}=\frac{R_1}{R_1+R_2}(+V_{sat})$	$V_{lt}=\frac{R_1}{R_1+R_2}(-V_{sat})$	V_{ut}	V_{lt}
1						
2						
3						

RESULT:

QUESTIONS:

1. Which is type of comparator called Schmitt trigger using IC741?
2. What is the output wave of Schmitt trigger if the input is sine wave?
3. What type of waveform is obtained when triangular or ramp waveforms are applied to Schmitt trigger circuit?
4. Explain how a square wave is obtained at the output of timer when sine wave input is given?
5. What is the Threshold voltage?
6. How do you calculate the theoretical values of V_{UT} and V_{LT} in the case of IC741?
7. What is the Hysteresis width?
8. What is the minimum amplitude of the input sine wave in the case of Schmitt trigger using IC741?

EXPERIMENT NO:7 **DATE:**

IC565 PLL APPLICATIONS

AIM:

1. To study the operation of NE565PLL
2.To use NE565 as multiplier

EQUIPMENTSANDCOMPONENTS:

APPARATUS:

1. DC power supply	1No.
2. CRO	1No.
3. Function Generator	1No.
4. Bread Board	1No.

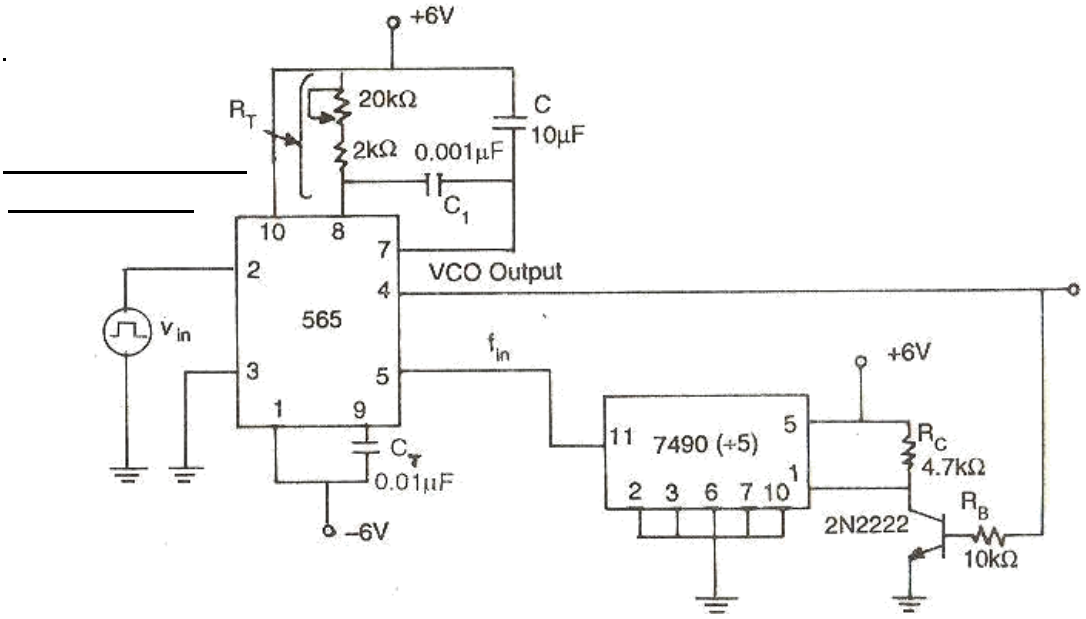
THEORY:

The 565 is available as a14-pin DIP package. It is produced by Signatic Corporation. The output frequency of the VCO can be rewritten as

$$f_o = 0.25 / R_T C_T \text{ Hz.}$$

Where R_T and $C_T$are the external resistor and capacitor connected to pin8 and pin9. A value between 2k and 20k is recommended for R_T .The VCO free running frequency is adjusted with R_T and C_T to be at the centre for the input frequency range.

CIRCUIT DIAGRAM:



PROCEDURE:

i. Connect the circuit using the component values as shown in the figure
ii .Measure the free running frequency of VCO at pin4 with the input signal V_{in}set= zero. Compare it with the calculated value=$0.25/R_TC_T$
iii. Now apply the input signal of 1Vpp square wave at a1kHz to pin2
iv. Connect1st channel of the scope to pin2 and display this signal on the scope.
v .Gradually increase the input frequency till the PLL is locked to the input frequency. This frequency f_1 gives the lower ends of the capture range. Go on increase the input frequency; till PLL tracks the
input signal, say to a frequency f_2.This frequency f_2 gives the upper end of the lock range. If the input frequency is increased further the loop will get unlocked.

vi. Now gradually decrease the input frequency till the PLL is a gain locked. This is the frequency f_3, the upper end of the capture range .Keep on decreasing the input frequency until the loop is unlocked. This frequency f_4 gives the lower end of the lock range
vii. The lock range $f_L=(f_2-f_4)$ compare it with the calculated value of $(7.8\ f_o/12)$
Also the capture range is $f_c=(f_3-f_1)$. Compare it with the calculated value of

capture range. $f_c = [f_L / (2)(3.6)(10^3)C]^{1/2}$

viii To use PLL as a multiplier,make connections as show in fig. The circuit uses a 4-bit binary counter7490 used as a divide-by-5circuit.
ix. Set the input signal at 1Vpp square wave at 500Hz
x.. Vary the VCO frequency by adjusting the 20K potentiometer till the PLL is locked.
Measure the output frequency.
xi. Repeat step9 and10 for input frequency of 1kHz and 1.5kHz.

OBSERVATIONS:

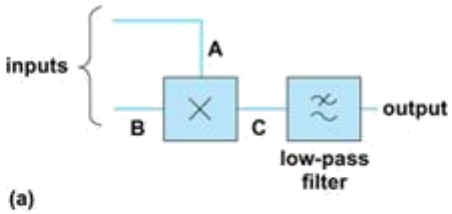
f_O =
f_L =
f_C =

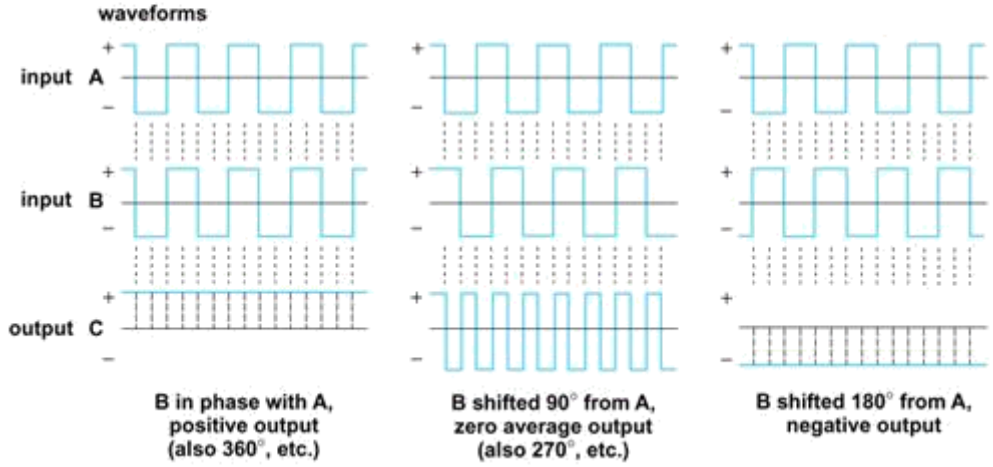
CALCULATIONS:

$f_L = (f_2 - f_4) = 7.8f_O/12$

$f_C = (f_3 - f_1) = [f_L/(2)\ (3.6)(10^3)C]^{1/2}$

GRAPH:





RESULT:

f_O =

f_L =

f_C =

EXPERIMENT No: 8 **DATE**

VOLTAGE REGULATOR USING IC 723, THREE TERMINAL VOLTAGE REGULATORS – 7805, 7809, 7912

AIM:

To study the Fixed Voltage Regulators (1) 7805
(2) 7809
(3) 7812
(4) 7912
(5) 723 Variable Voltage Regulator

THEORY:

DC power for electronic circuits is most conveniently obtained from commercial ac lines by using rectifier - filter system, Called a dc power supply. The rectifier-filter combination constitutes an ordinary dc power supply . The dc voltage from an ordinary power supply remains constant so long as ac mains voltage or load is unaltered . However, in many electronic applications, it is desired that dc voltage should remain constant irrespective of changes in ac mains or load. Under such situations, voltage regulating devices are used with ordinary power supply. This constitutes regulated dc power supply and keeps the dc voltage at fairly constant value.

ORDINARY DC POWER SUPPLY

An ordinary or regulated dc power supply contains a rectifier and a filter circuit as shown in Fig-1. The output from the rectifier is pulsating dc. These pulsations are due to the presence of ac component in the rectifier output. The filter circuit removes the ac component so that steady dc voltage is obtained across the load.

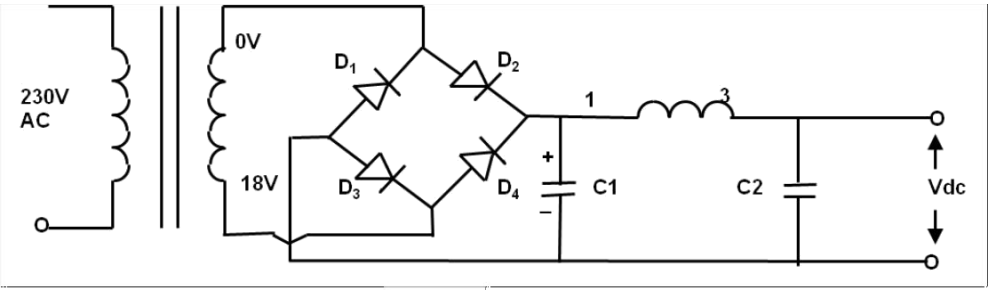


Fig -1.

Limitations: An ordinary dc power supply has two following drawbacks:

1. The dc output voltage changes directly with input ac voltage.
2. The dc output voltage decreases as the load current increases. This is due to voltage drop in (a) Transformer windings (b) Rectifier (c) Filter circuit

These variations in dc output voltage may cause inaccurate or erratic operation or even malfunctioning of many electronic circuits. Eg. In an oscillator, the frequency will shift and in transmitters, distorted output will result, therefore, ordinary power supply is unsuited for many applications and is being replaced by regulated power supply.

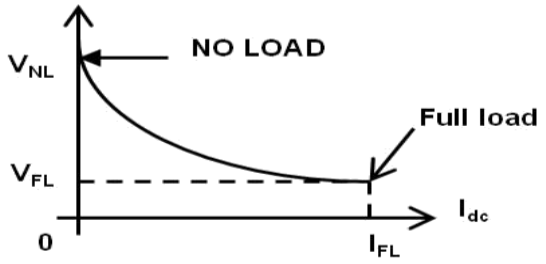
For comparison of different types of power supplies, the following terms are commonly used:

1. **Voltage Regulation** : The dc voltage available across the output terminals of a given power supply depends upon load current. If the load current Idc is increased by decreasing RL as in Fig-

2, there is greater voltage drop in the power supply and hence smaller dc output voltage will be available . Reverse will happen if the load current decreases. The variation of output voltage w.r.t. the amount of load current drawn from
the power supply is known as voltage regulation and is expressed by the following relation:

% voltage regulation = $(V_{NL} - V_{FL}) / V_{FL} * 100$

V_{NL} = dc output voltage at no load.
V_{FL} = dc output voltage at full load



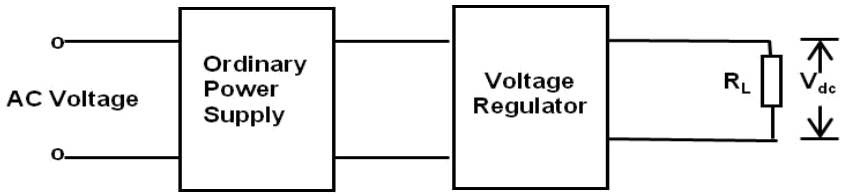
In a well designed power supply, the full load voltage is only slightly less than no -load voltage I.e. voltage regulation approaches zero. Therefore, lower the voltage regulation, the lesser the difference between full-load and no-load voltage and better is the power supply. Power supplies used in practice have a voltage regulation of 1% i.e. full load voltage is within 1% of the no -load voltage. Fig-3 shows the change of dc output voltage with load current. This is known as voltage regulation curve.

2. **Minimum Load Resistance** : The change of load connected to a power supply varies the load current and hence the dc output voltage. In order that a power supply gives the rated output voltage and current, there is minimum load resistance allowed. For instance, I a power supply is required to deliver a full-load current I_{FL} at full load voltage V_{FL}, then,

$$R_{L\ (min)} = \frac{V_{FL}}{I_{FL}}$$

Regulated Power Supply

A dc power supply which maintains the output voltage constant irrespective of ac mains fluctuations or load variations is known as regulated dc power supply. A regulated power supply consists of an ordinary power supply and voltage regulating device an in fig-4. The output of ordinary power supply is fed to the voltage regulator which produces the final output. The output voltage (Vdc) remains constant whether the load current changes or there are fluctuations in the input ac voltage.



Need : In an ordinary power supply, the voltage regulation is poor i.e dc output voltage changes appreciably with load current. Moreover, output voltage also changes due to variations in the input ac voltage. This is due to the following reasons:-

i) In practice, there are considerable variations in ac line voltage caused by outside factors beyond our control. This changes the dc output voltage. Most of the electronic circuits will refuse to work satisfactorily on such output voltage fluctuations. This necessitates to use regulated dc power supply.

ii) The internal resistance of power supply is relatively large (>30W). Therefore, output voltage is markedly affected by the amount of load current drawn from the supply. These variations in dc voltage may cause erratic operation of electronic circuits. Therefore, regulated dc power supply is the only solution in such situations.

HARDWARE SPECIFICATIONS:

1. Built - in 16V - 0 - 16V / 350mA
   12V - 0 - 12V / 350mA
   8V - 0 - 8V / 350mA AC sources
2. Bridge rectifier using IN4007 diodes - 1No.
3. Filter capacitors (470μF / 35V) - 2Nos.
4. Fixed Voltage Regulator 7805 - 1No.
   7809 - 1No.
   7812 - 1No.
   7912 - 1No.
5. Variable Voltage Regulator using 723 IC

EXPERIMENTAL PROCEDURE:

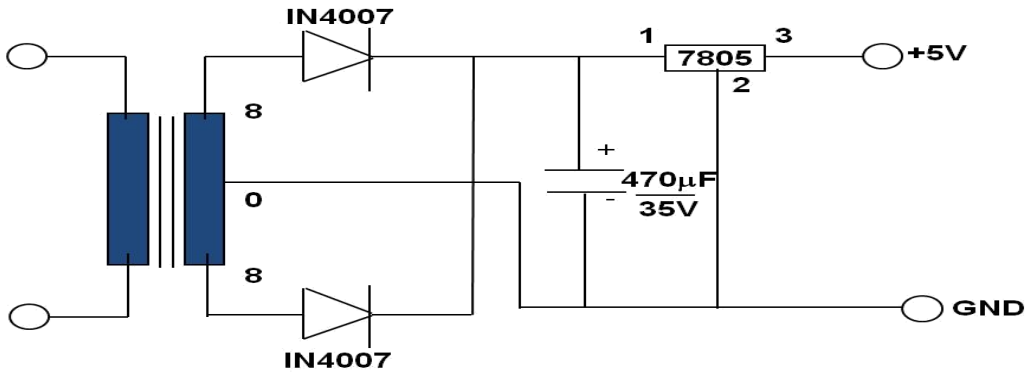


Fig - 4

1. Connect the circuit as shown in fig - 4.
2. Connect different load resistors available in the front panel, note down the output current and voltage.
3. Also test the circuit with 12V - 0 - 12V, 16V - 0 - 16V AC sources also.
4. Remove 7805 and connect 7809, 7812 also repeat 2 and 3 steps.
5. Connect the circuit shown in fig - 5.

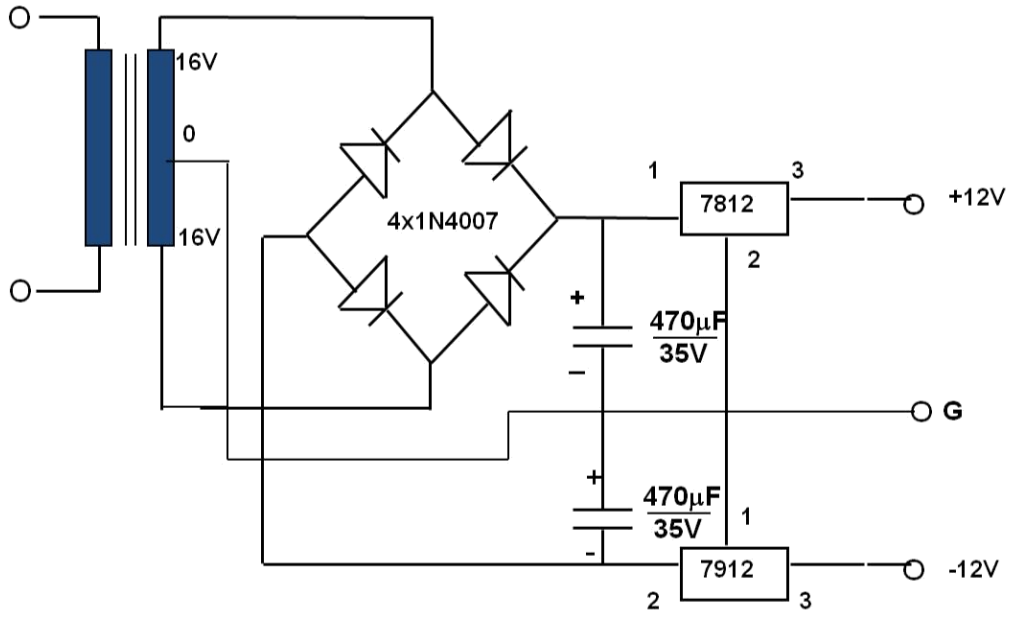


Fig - 5

723 Regulator

It is a monolithic voltage regulator constructed on a single Silicon chip. The device consists of a temperature compensated reference amplifier,, error amplifier, power series pass transistor and current Limit circuitry. Additional NPN or PNP pass element may be used when output currents exceeding 150mA are required. Provisions are made for adjustable current limiting and remote shutdown. In addition to the above the device features low stand by current drain, low temperature drift and high Ripple rejections. The 723 is intended for use with positive or negative supplies as a series, shunt , switching or floating Regulator. Applications include laboratory power supplies, air borne systems and linear circuits.
and other power supplies for digital

CIRCUIT DESCRIPTION:

Fig -4 shows the circuit of a variable regulator constructed with 723 IC. Pin is connected to positive terminal of the supply. 10KW potentiometer connected to pin 4, controls the output voltage. Output is available at Pin10. This output is not sufficient to drive loads. So it is passed through the Darlington pair of transistors (CL100).

EXPERIMENTAL PROCEDURE:

1. Switch ON the experimental board by connecting power card to the AC mains.
2. Make sure that the potentiometer are in minimum position.
3. Connect the 8V AC tapping of the transformer secondary to the bridge rectifier input and short Raw DC +Ve point and pin 12 of 723 (See Fig-)
4. Measure the output voltage with a DMM and also measure the output with the 10KW potentiometer with its maximum position.
5. Now, disconnect the 8V AC tapping and connect 10V AC tapping and notedown the minimum and maximum output voltages with 10KW minimum and maximum positions.
6. Repeat the same procedure for 12V, 16V and 18V AC transformer secondary tapings and tabulate these values in Table-1.
7. Now, again connect 8V AC to the bridge rectifier input, set the output DC voltage at 5V with 10KW potentiometer.
8. Connect the load resistor with (0-50mA) milliammeter and vary the load resistor and note down the readings of the output voltage and output current with different load resistor.

9. Tabulate these values in Table-2.

S.No.	AC INPUT (RMS Volts)	DC OUTPUT minimum	maximum
1.	8V	2.96V	5.58V
2.	10V	3.00V	5.69V
3.	12V	3.03V	5.77V
4.	16V	3.08V	5.89V
5.	18V	3.10V	5.90V

Table -1

S.No.	LOAD CURRENT I_{OUT} (mA)	OUTPUT VOLTAGE V_{OUT} (volts)
1.	49mA	5.27V
2.	45mA	5.30V
3.	40mA	5.32V
4.	35mA	5.36V
5.	30mA	5.40V
6.	25mA	5.41V
7.	20mA	5.43V
8.	15mA	5.50V
9.	10mA	5.5V
10.	5mA	5.5V

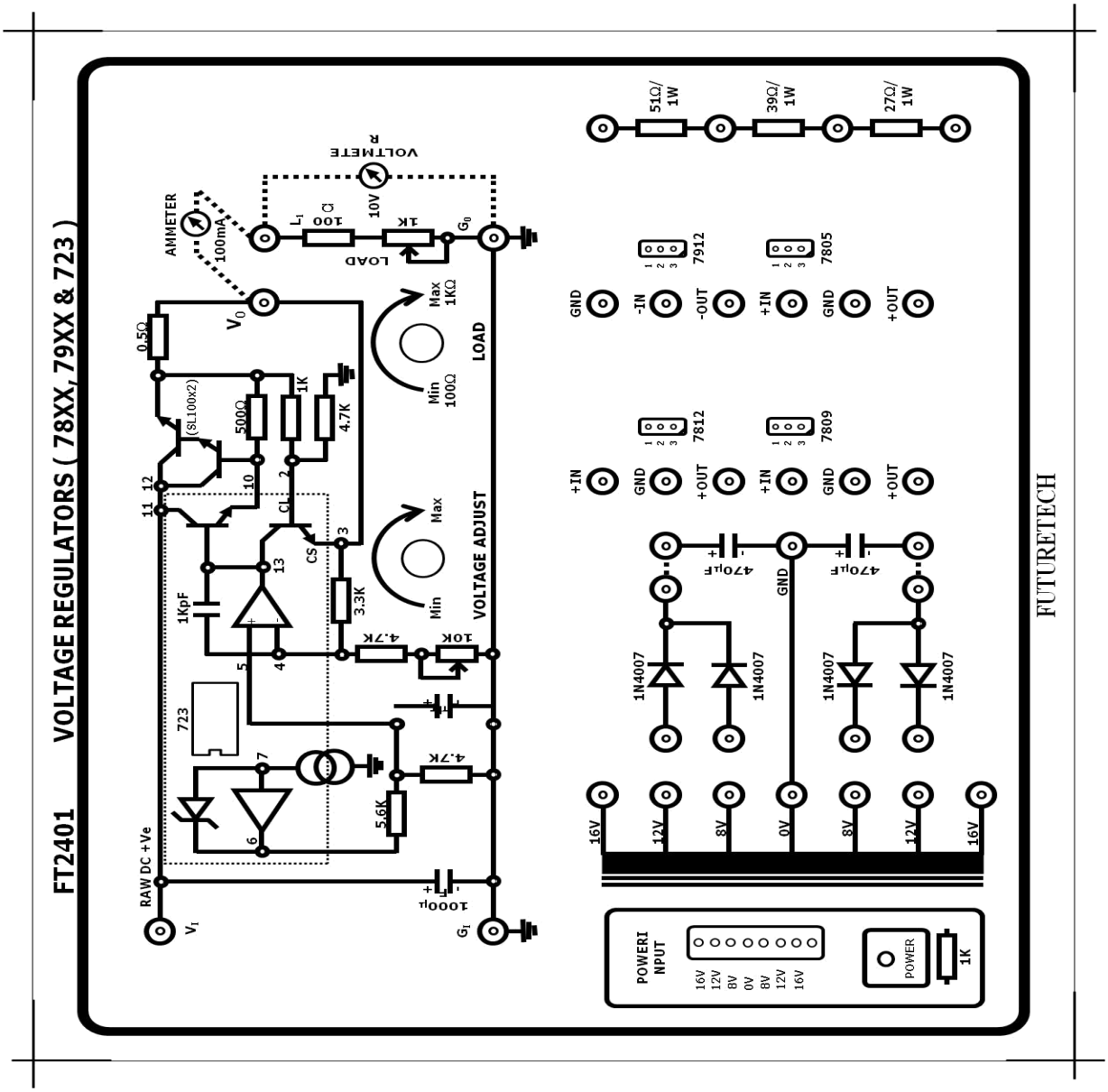
Table -2

LOAD REGULATION:

10. Calculate & Tabulate the load regulation at each load current.

% Regulation = $(V_{NL} - V_{FL})/V_{FL} * 100$

11. Draw a graph between load current I_{dc} and load voltage V_{dc} regulation.



RESULT:

CYCLE - II

INTRODUCTION - XILINX

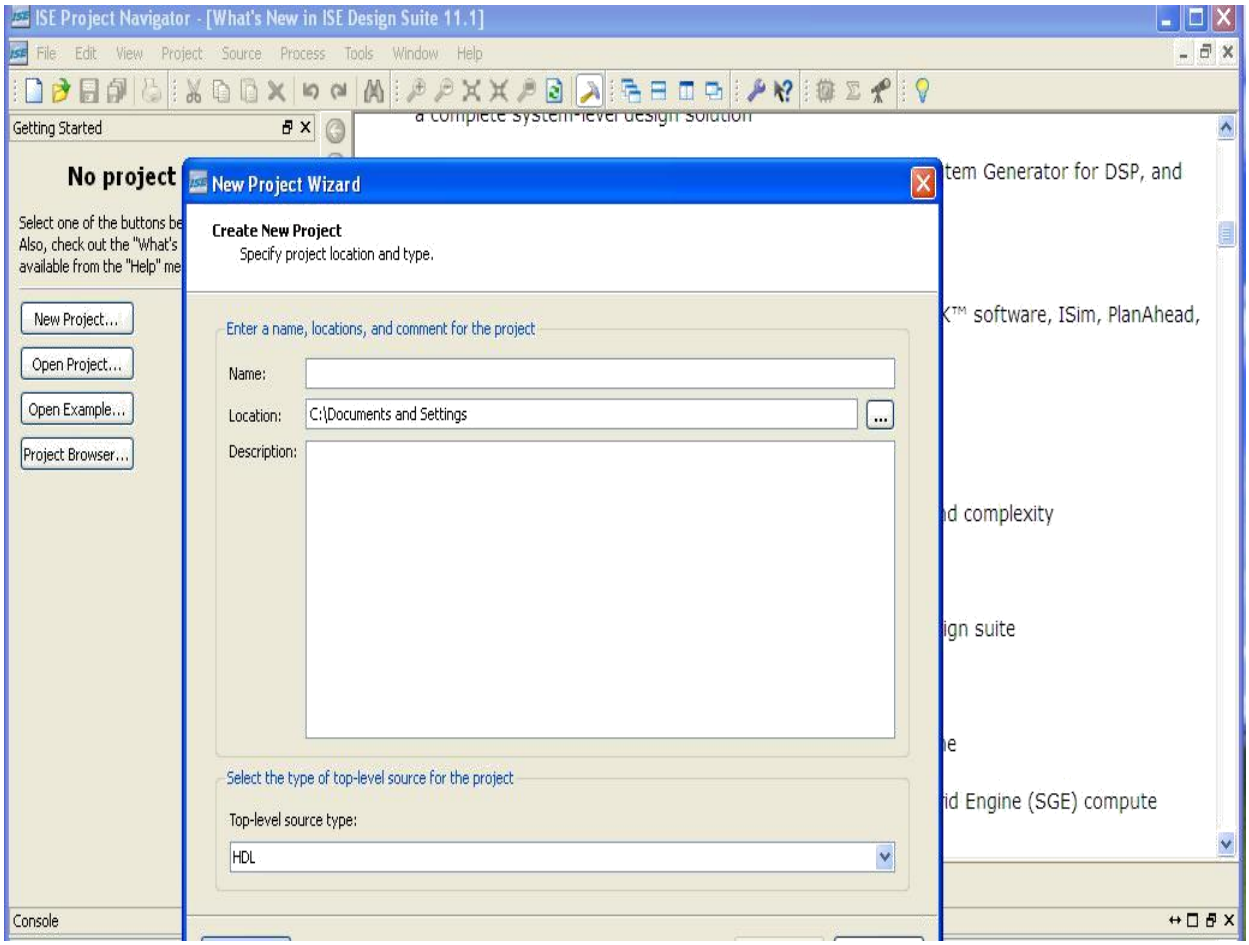
Xilinx ISE is a software tool produced by Xilinx for synthesis and analysis of HDL designs, which enables the developer to synthesize ("compile") their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer.

In our Lab, the scope is limited to design and analyze the design using test benches & simulation.

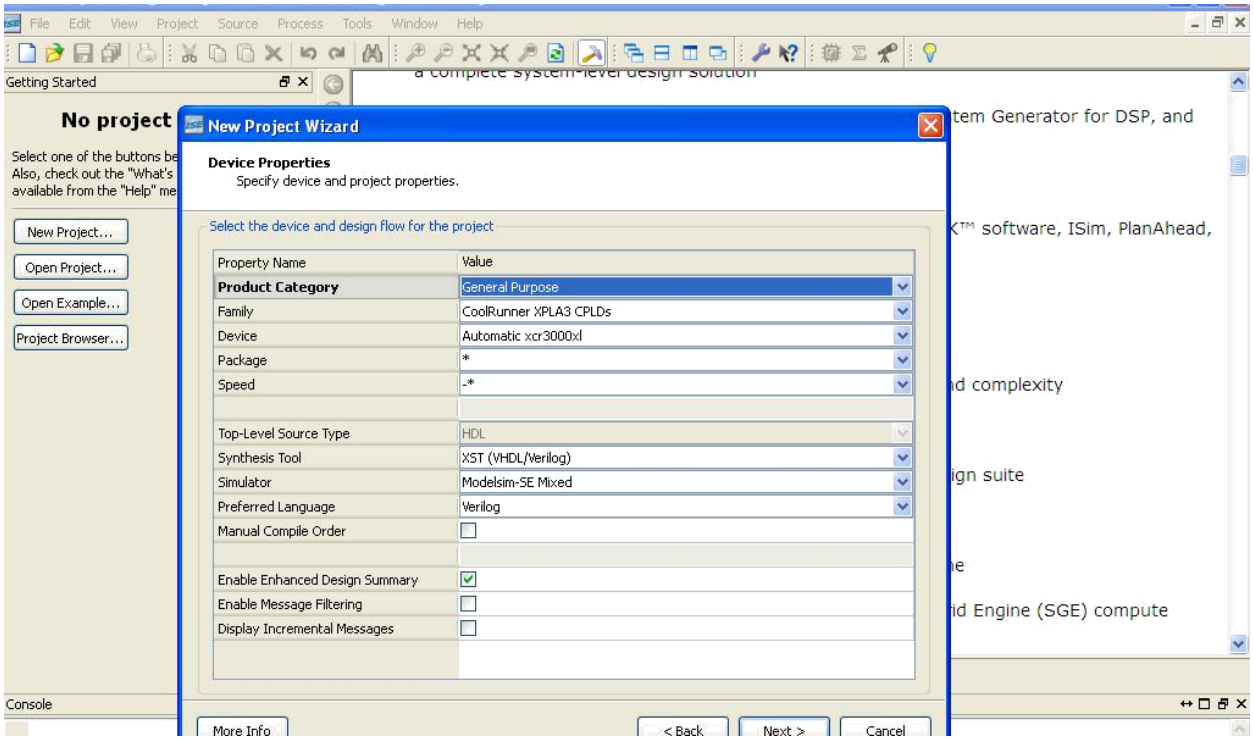
The following is the step by step procedure to design in the Xilinx ISE:

1. ***New Project Creation***

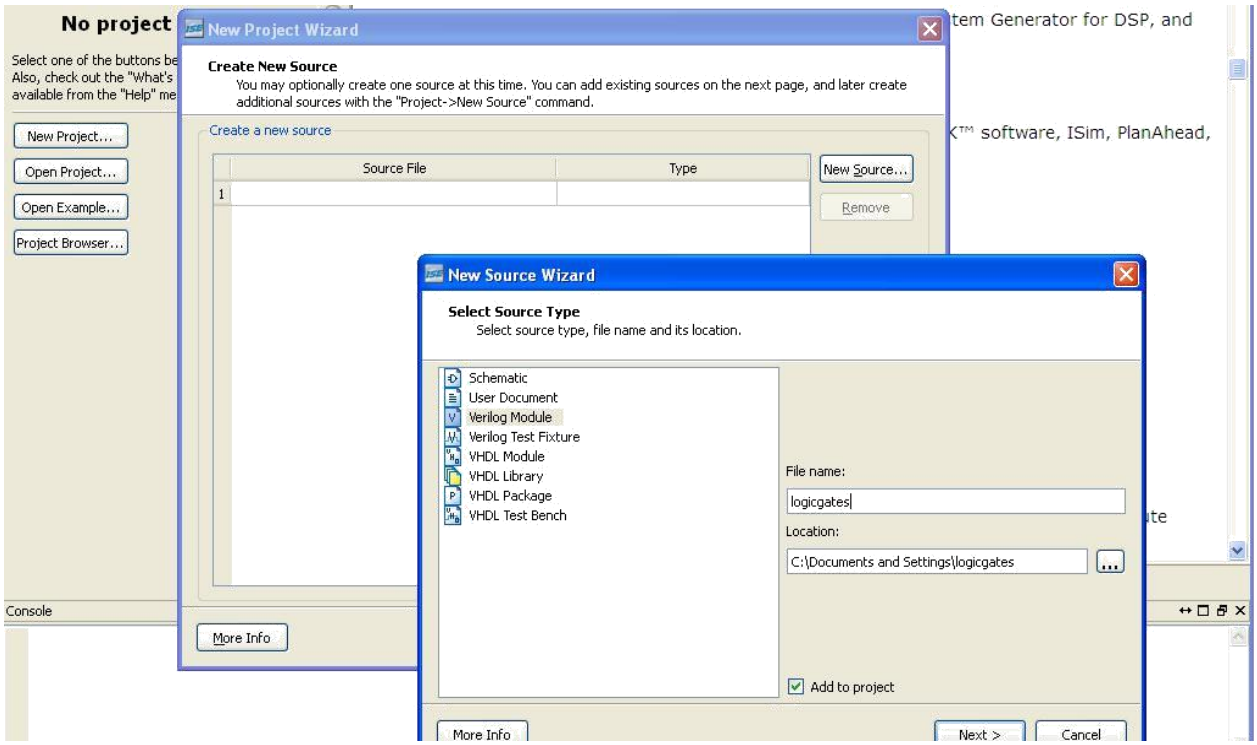
   Once the Xilinx ISE Design suite is started, open a new project & enter your design name and the location path. By default 'HDL' is selected as the top-level source type. (If not, please select Top-level source type as 'HDL')



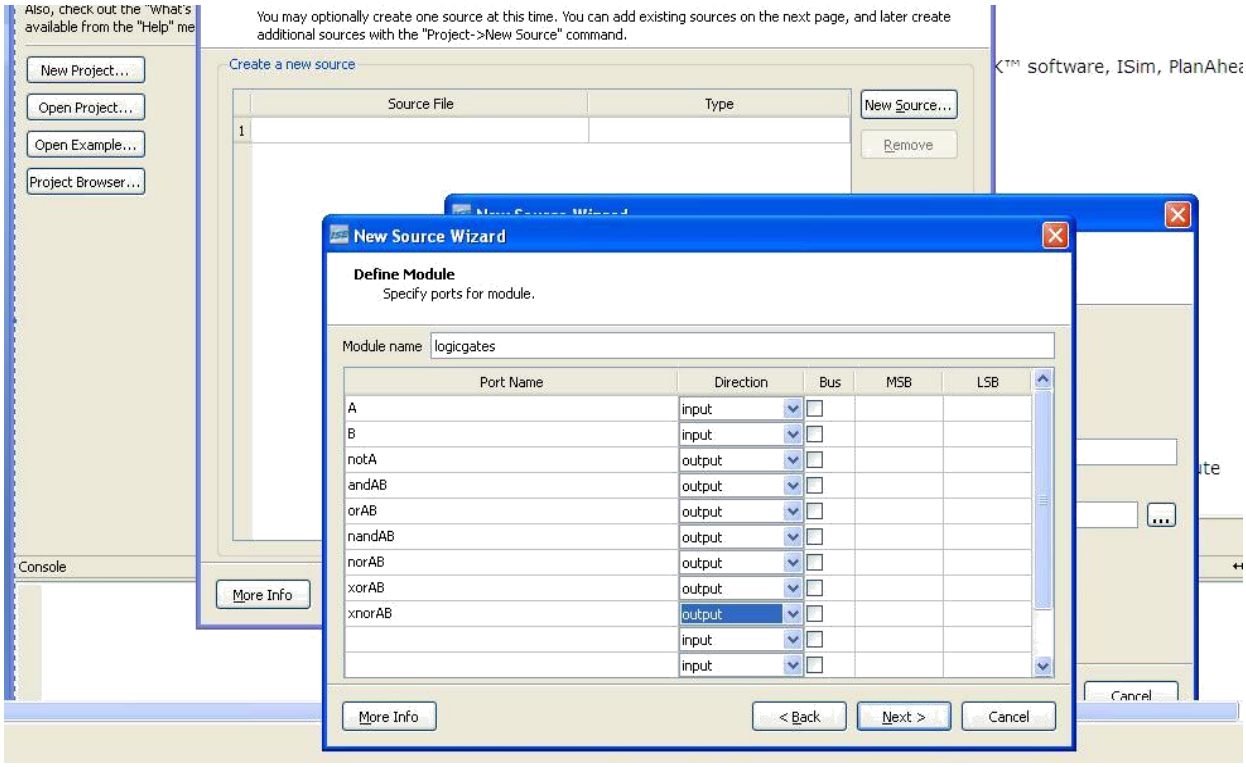
2. Continue to the next window and check if the Preferred Language is selected as 'Verilog'



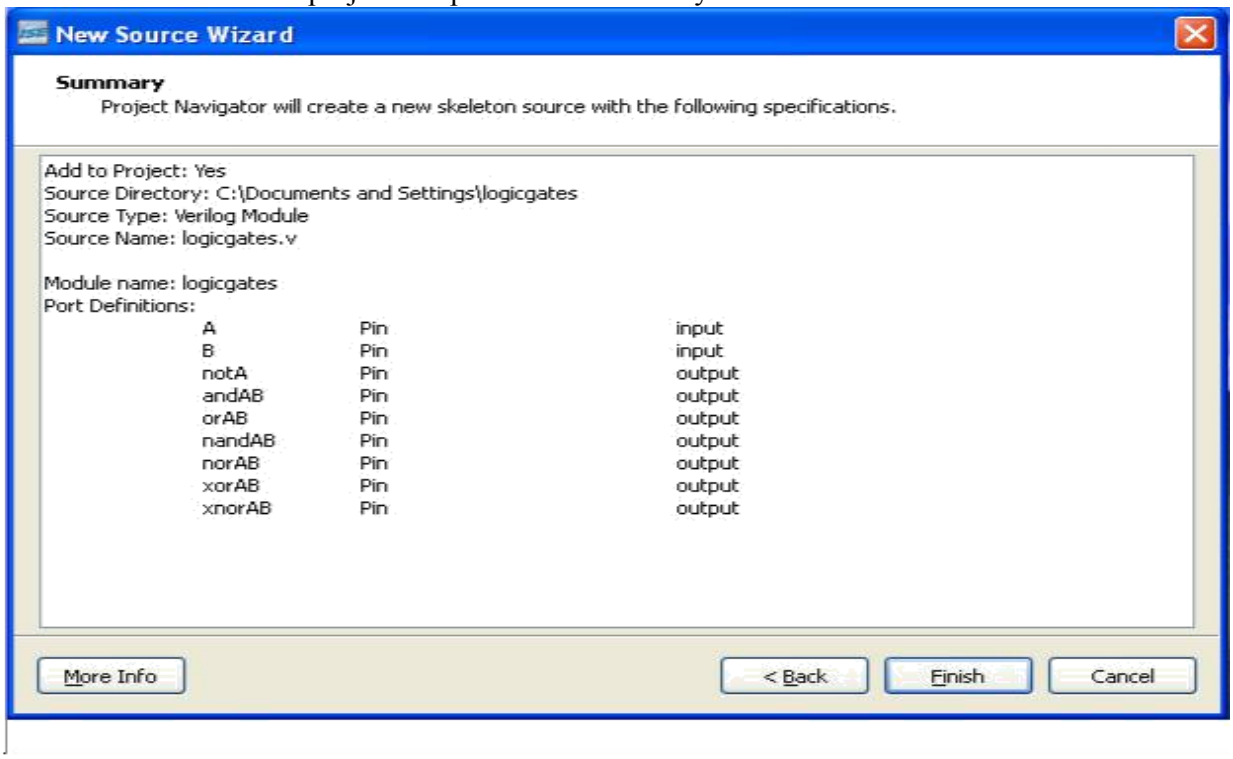
3. Proceed by clicking 'Next' and create a 'New Source' using the 'Create New Source' Window



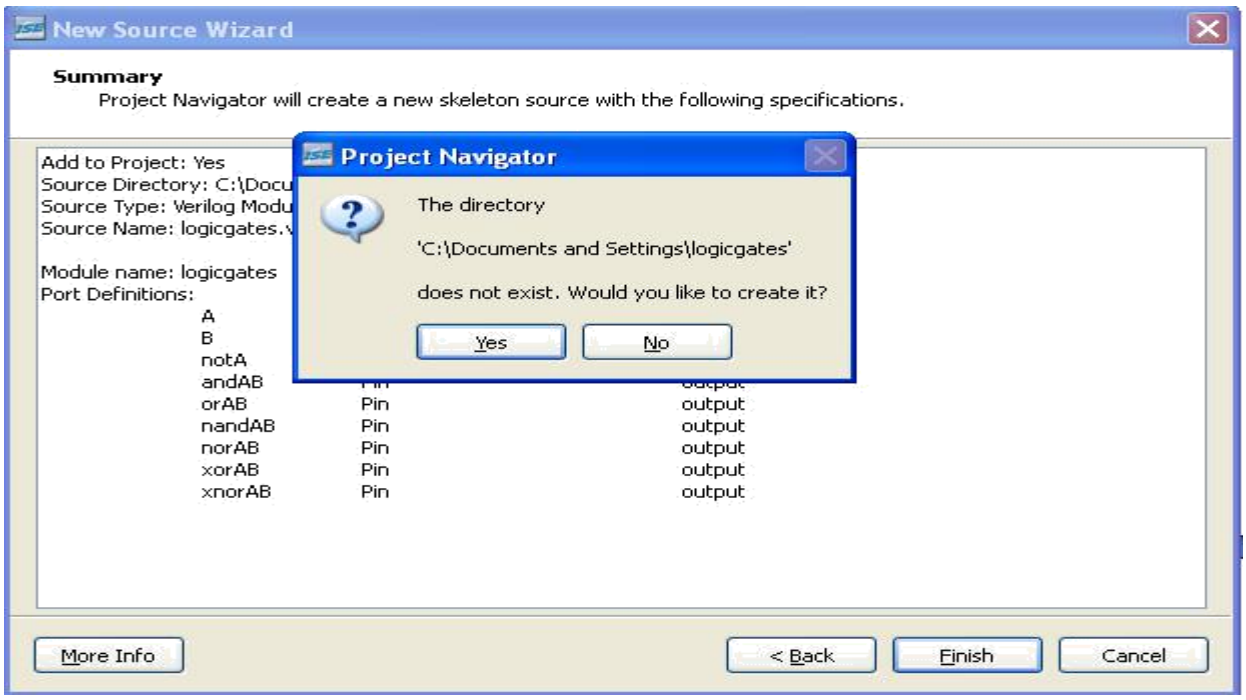
4. Select the source type as ‘Verilog Module’ and input a filename and proceed to ‘Next’. In the next window ‘Define Module’ enter the ports.



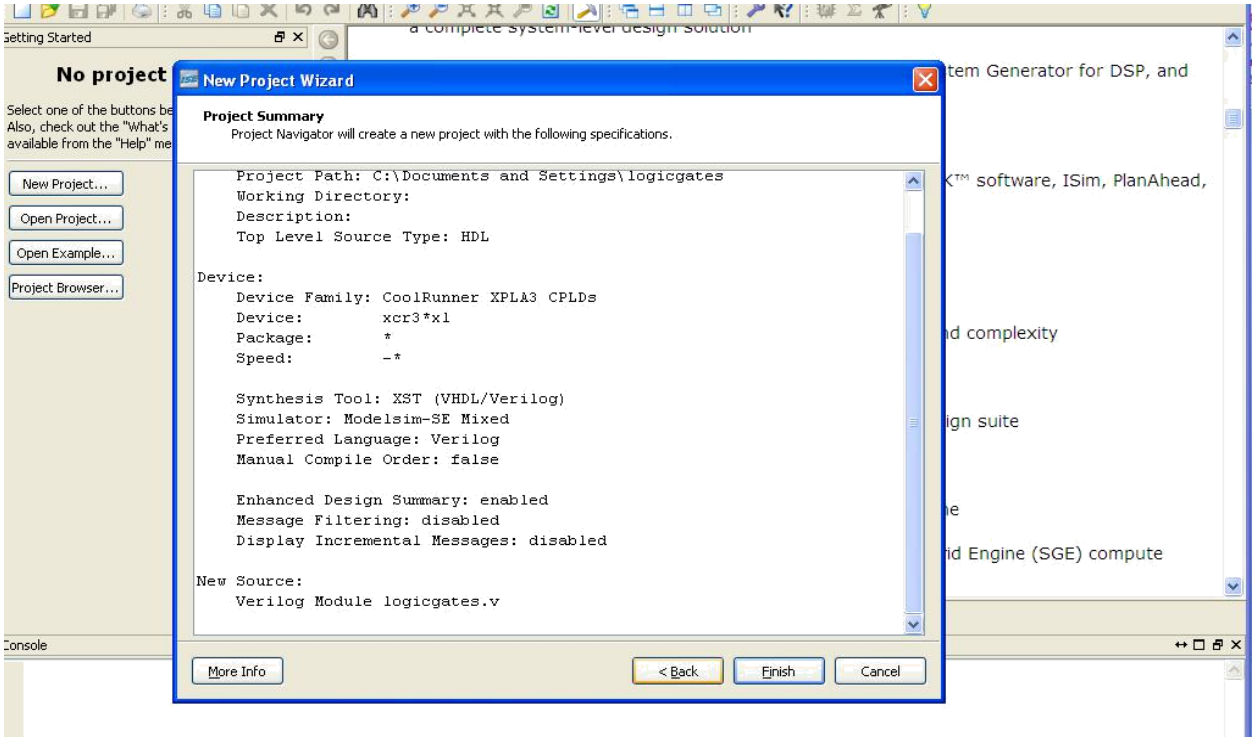
5. Finish with the New project setup with the ‘Summary’ window.



6. Once 'Finish' is selected a pop-up appears to create the directory. Select 'yes'

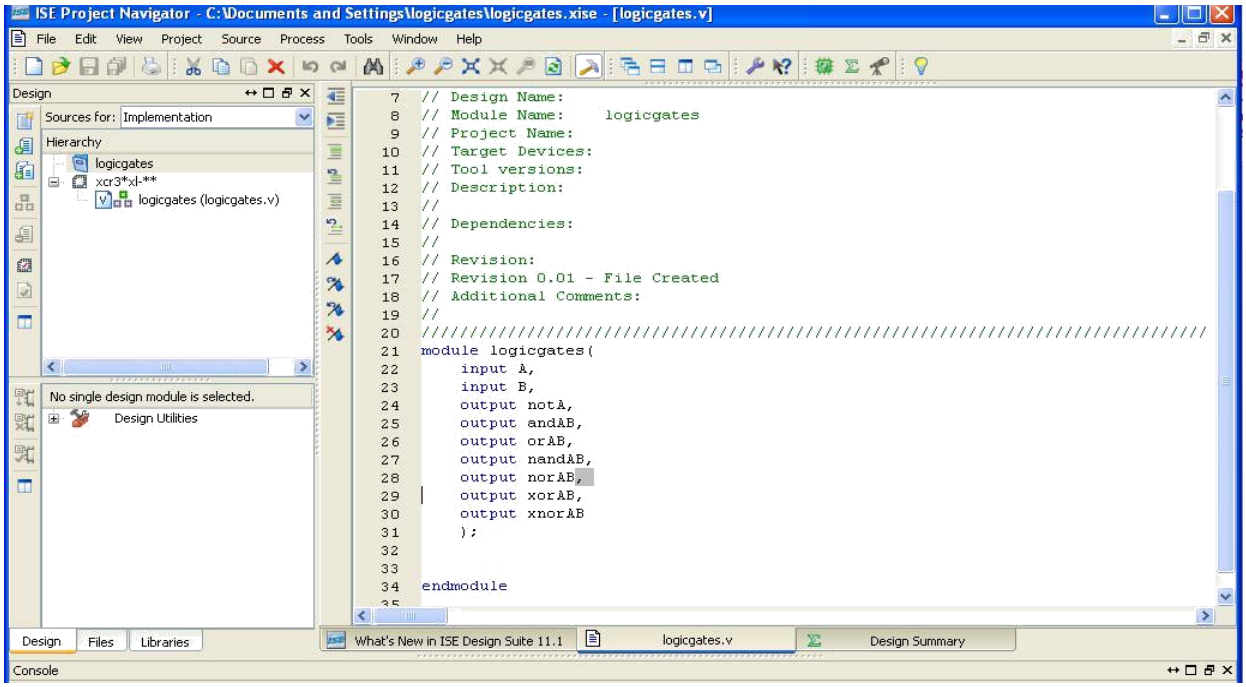


7. Then proceed to 'Next' in the "New Project Wizard' to 'Add Existing Sources'. 'Add source' if an existing source is available, If not proceed to 'Next' and finish with the 'Project Summary' window

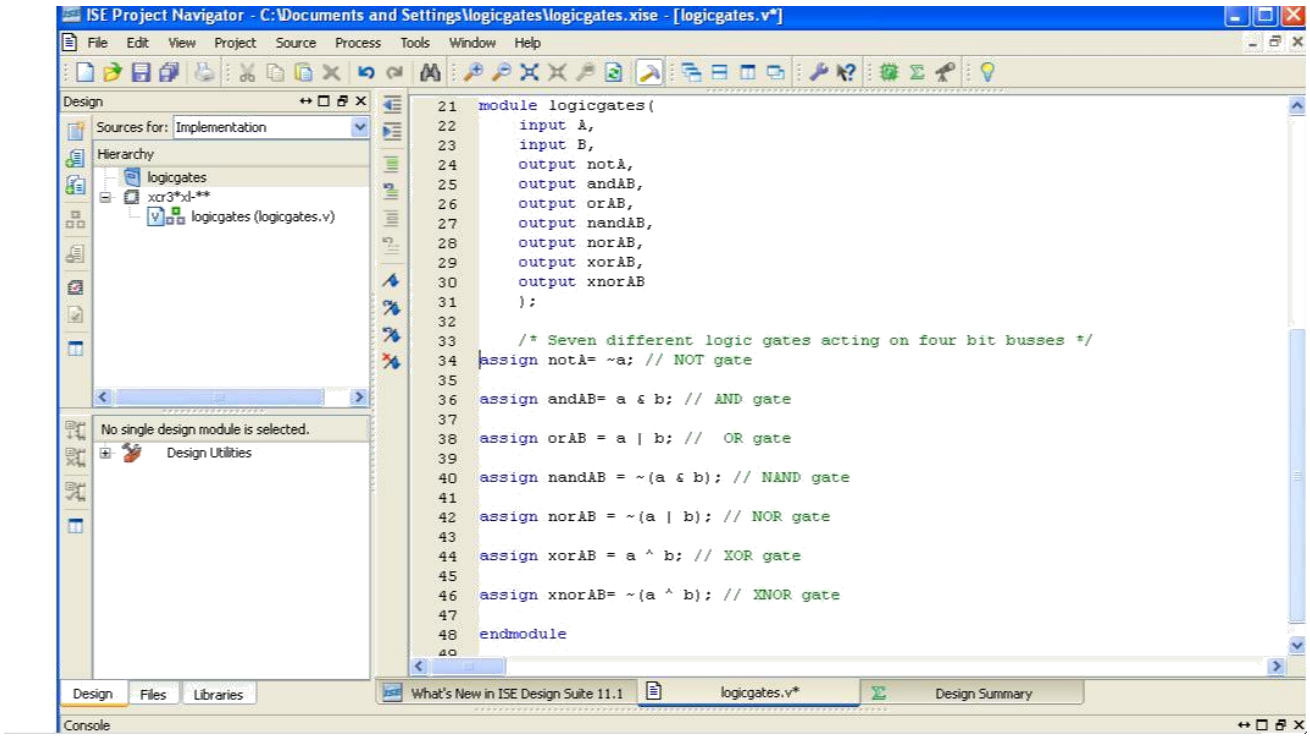


8. Design Entry and Syntax Check

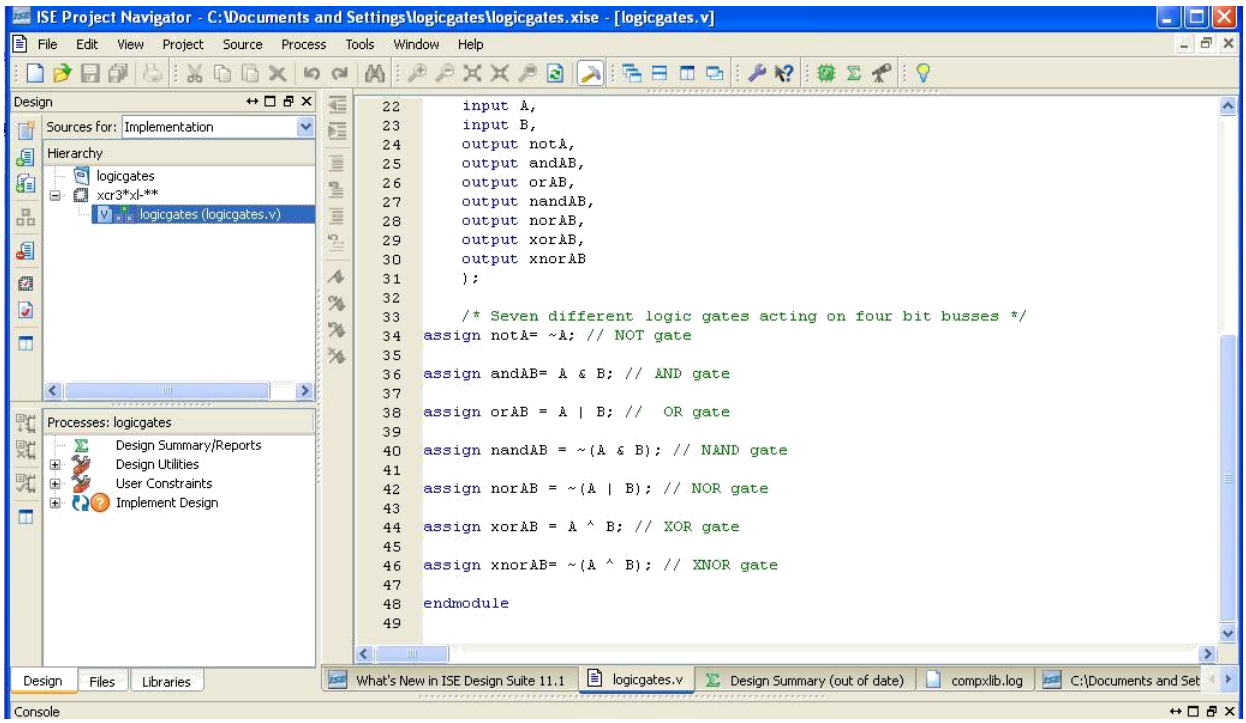
The ports defined during the 'Project Creation' are defined as a module in the 'filename.v' file



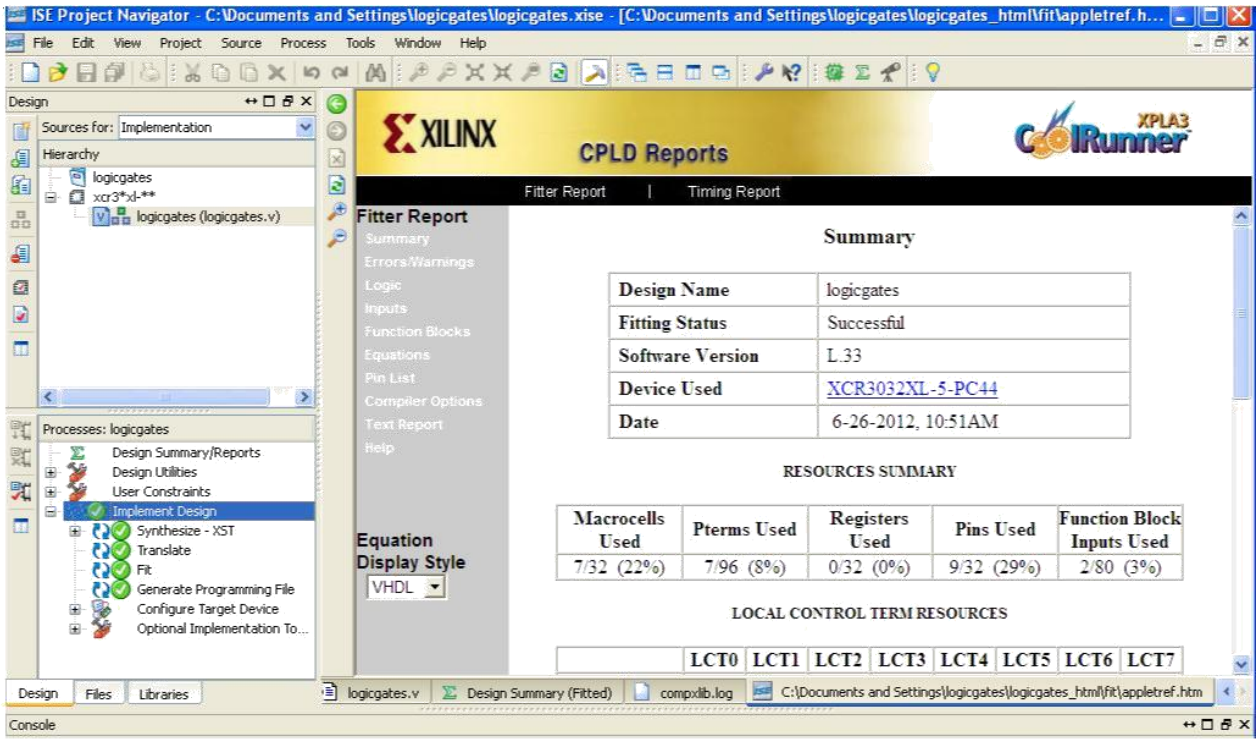
9. Input your design (verilog code) within the module definition



10. Select the design from the 'Hierarchy'window. In the below window of Processes 'Implement Design 'would be orange (in color) ready for implementation

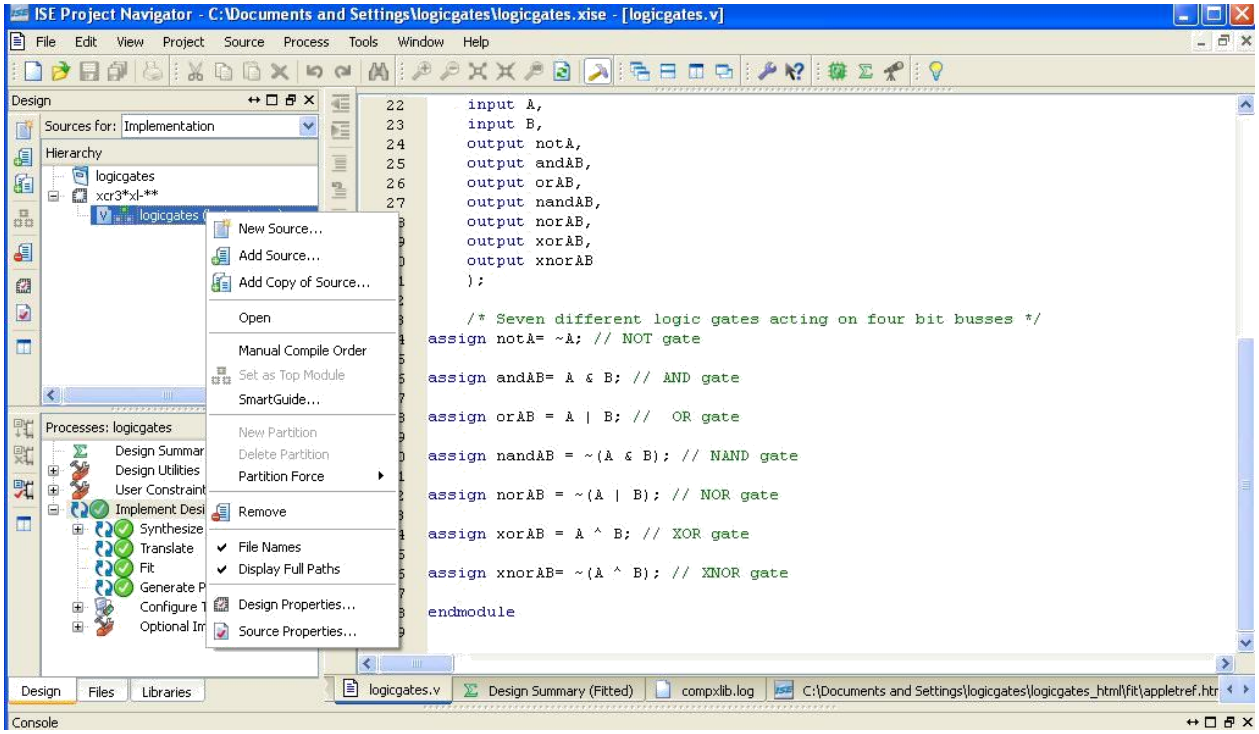


11. Double click on implement design, it turns green (in color) once the design is implemented successfully and the Summary report is displayed.

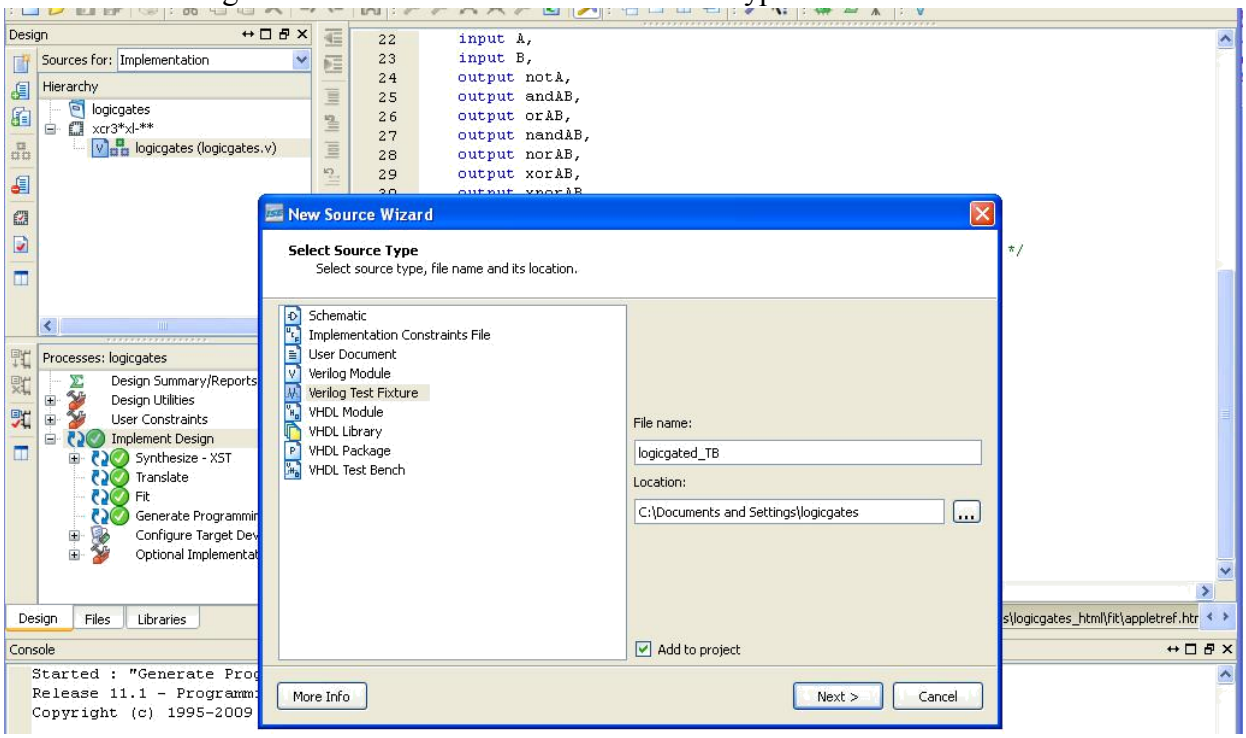


12. Test-Bench creation, Simulation & Verification

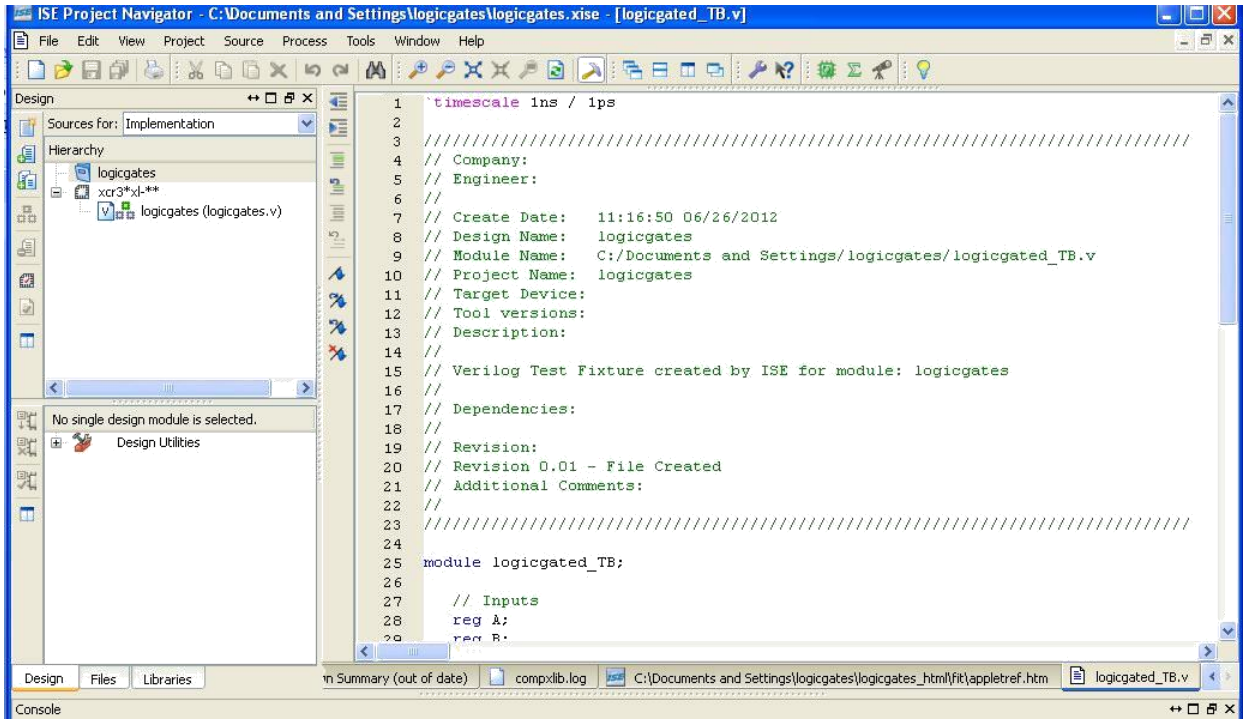
To add a test-bench to the existing design, right click on the '.v' file from the Hierarchy window and select 'New Source'



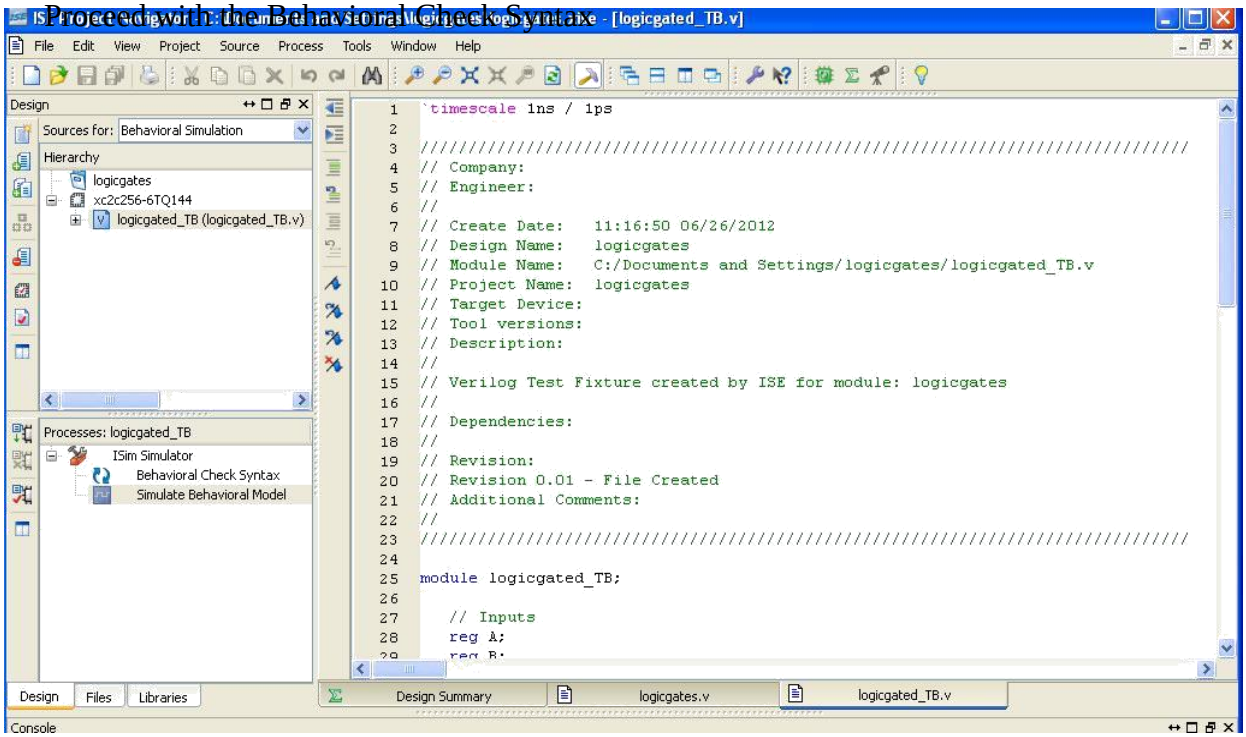
13. Select 'Verilog Text Fixture' from the Select Source Type and name the Test-Bench



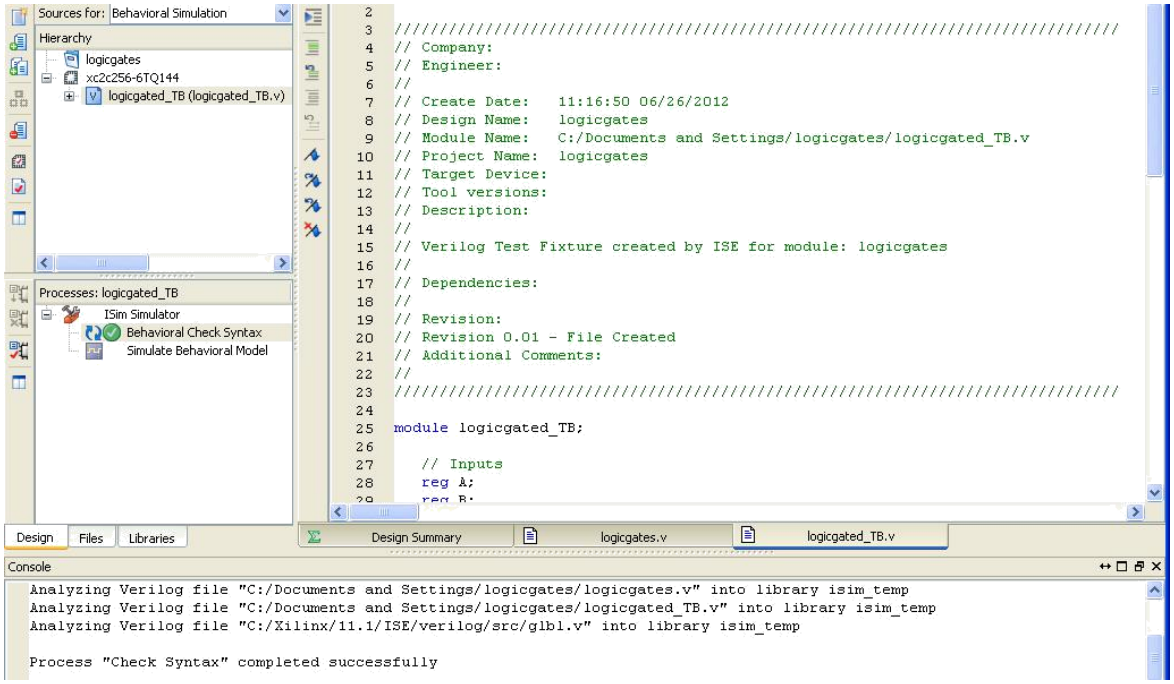
14. Continue to 'Finish' and a test bench is added in the project area



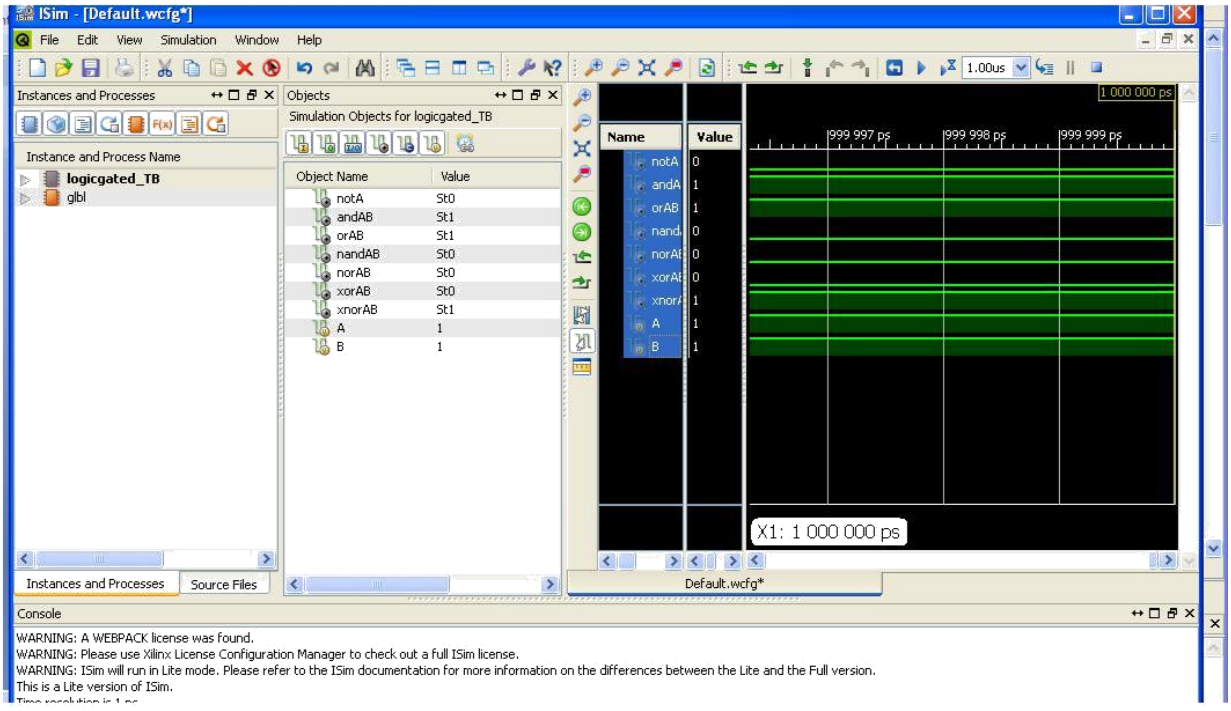
15. Edit the test bench as per your simulation requirements and select 'Behavioral Simulation' in the 'Design Window'. In the Processes window Isim Simulator would be displayed. First Proceed with the Behavioral Check Syntax



16. Double click on ‘Behavioral Check Syntax’ & check for no errors



17. Then double click on ‘Simulate Behavioral Model’ and the ISIM simulator window would open. Check for the outputs



EXPERIMENT No: 1 **DATE**

HDL CODE TO REALIZE ALL LOGIC GATES

AIM:

To develop the source code for logic gates by using VERILOG and obtain the simulation.

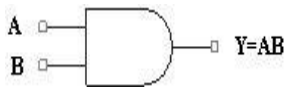
SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

LOGIC DIAGRAM:

AND GATE:

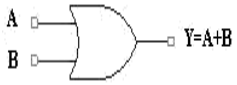
LOGIC DIAGRAM: TRUTH TABLE:



A	B	Y=AB
0	0	0
0	1	0
1	0	0
1	1	1

OR GATE:

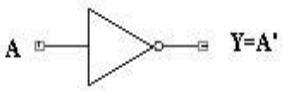
LOGICDIAGRAM TRUTH TABLE:



A	B	Y=A+B
0	0	0
0	1	1
1	0	1
1	1	1

NOT GATE:

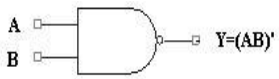
LOGIC DIAGRAM: TRUTH TABLE:



A	Y=A’
0	0
0	1

NAND GATE:

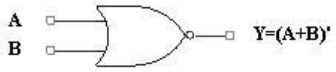
LOGICDIAGRAM TRUTH TABLE



A	B	Y=(AB)’
0	0	1
0	1	1
1	0	1
1	1	0

NOR GATE:

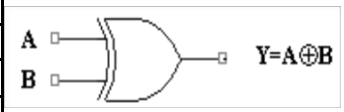
LOGIC DIAGRAM: TRUTH TABLE:



A	B	Y=(A+B)’
0	0	1
0	1	0
1	0	0
1	1	0

XOR GATE:

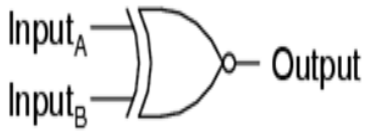
LOGICDIAGRAM TRUTH TABLE:



A	B	Y=A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

XNOR GATE:

LOGIC DIAGRAM:



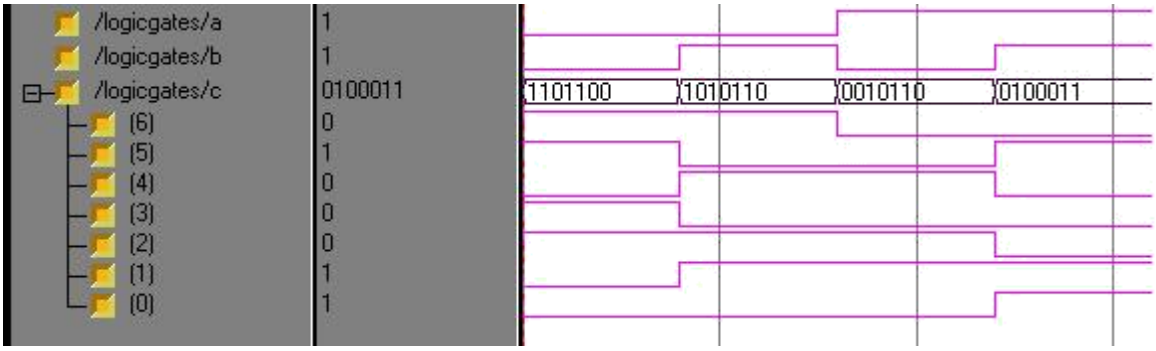
TRUTH TABLE:

INPUTS		OUTPUT
A	B	$Y = \overline{A \oplus B}$ $= AB + \overline{A}\,\overline{B}$
0	0	1
0	1	0
1	0	0
1	1	1

VERILOG SOURCE CODE:

```
module logicgates1(a, b, c);
   input a;
   input b;
   OUTPUT: [6:0] c;
assign c[0]= a & b;
        assign c[1]= a | b;
        assign c[2]= ~(a & b);
        assign c[3]= ~(a | b);
        assign c[4]= a ^ b;
        assign c[5]= ~(a ^ b);
        assign c[6]= ~ a;

endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of all logic gates are verified by simulating the VERILOG code.

EXPERIMENT No: 2 **DATE**

DESIGN OF 2-TO-4 ENCODER

AIM:

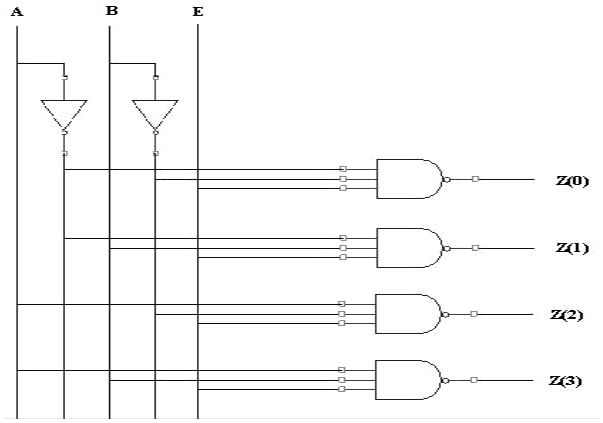
To develop the source code for encoder by using VERILOG and obtain the simulation.

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

DECODER

LOGIC DIAGRAM:



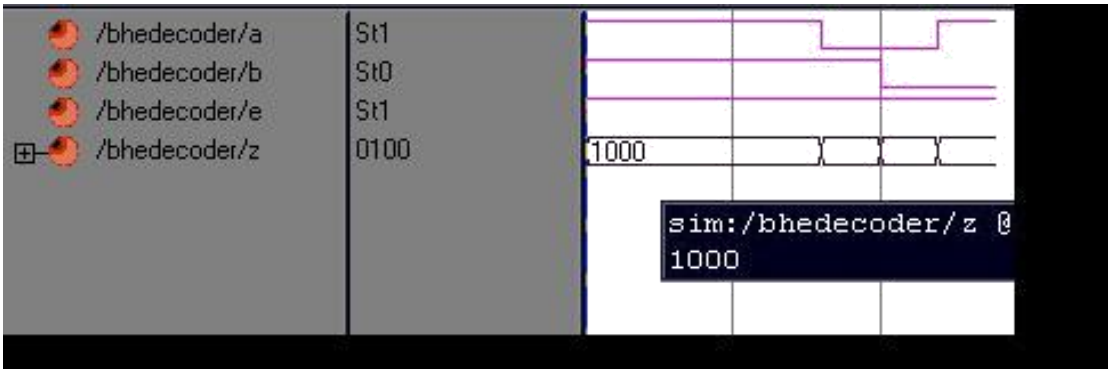
TRUTH TABLE:

A	B	C	Z(0)	Z(1)	Z(2)	Z(3)
0	0	1	0	1	1	1
0	1	1	1	0	1	1
1	0	1	1	1	0	1
1	1	1	1	1	1	0

VERILOG SOURCE CODE:

```
module decoderbehv(a, b, en, z);
   input a;
   input b;
   input en;
   output [3:0] z;
         reg [3:0] z;
reg abar,bbar;
         always @ (a,b,en) begin
         z[0] = (abar&bbar&en);
         z[1] = (abar&b&en);
         z[2] = (a&bbar&en);
         z[3] = (a&b&en);
         end
endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of encoder are verified by simulating the VERILOG code.

EXPERIMENT No:3 **DATE**

DESIGN OF 8-TO-3 ENCODER

AIM:

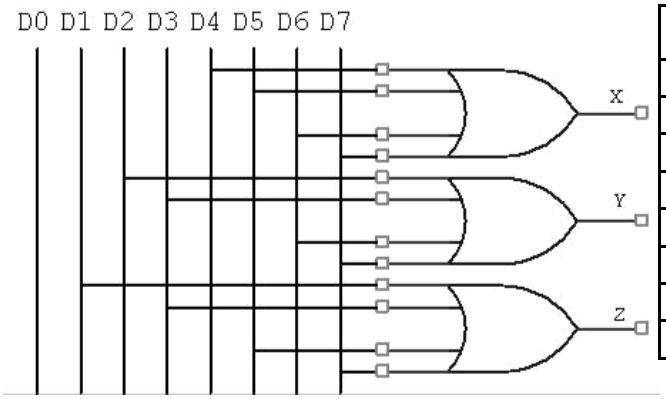
To develop the source code for encoder by using VERILOG and obtain the simulation.

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

ENCODER:

LOGIC DIAGRAM:



TRUTH TABLE:

D0	D1	D2	D3	D4	D5	D6	D7	X	Y	Z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

VERILOG SOURCE CODE:

```
module encoderbehav(d, a,b,c);
   input [7:0] d;
   output x;
   output y;
   output z;
         reg a,b,c;
         always @ (d [7:0]) begin
         a= d[4] | d[5] | d[6] | d[7];
         b= d[2] | d[3] | d[6] | d[7];
         c= d[1] | d[3] | d[5] | d[7];
         end
   endmodule
```

SIMULATION OUTPUT:

RESULT:

Thus the OUTPUT's of Encoded are verified by simulating the VERILOG code.

EXPERIMENT No: 4 **DATE**

DESIGN OF 8-to-1MULTIPLEXER AND 1X8 DEMULTIPLEXER

AIM:

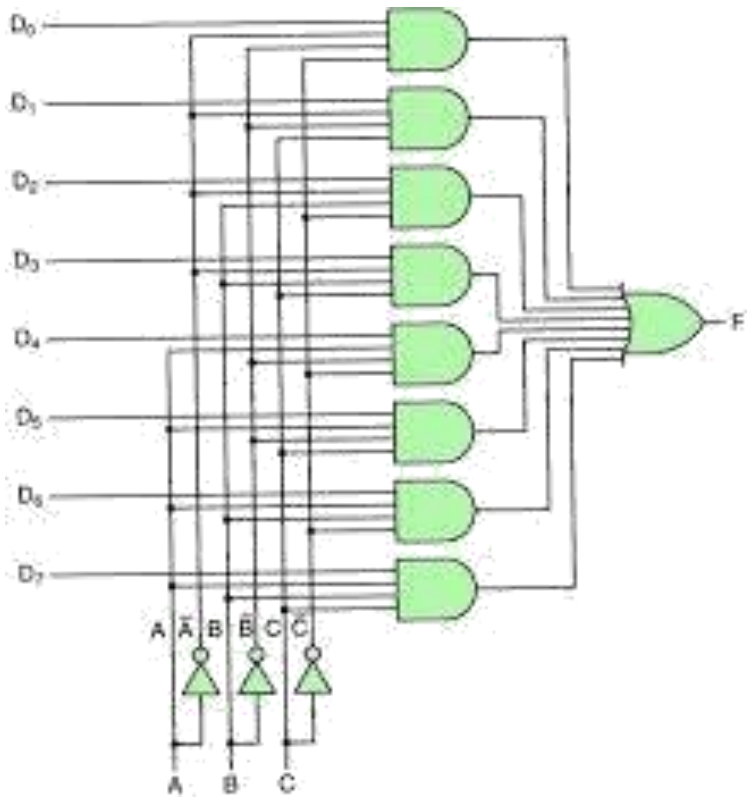
To develop the source code for 8x1 multiplexer and demultiplexer by using VERILOG and obtain the simulation.

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

MULTIPLEXER:

LOGIC DIAGRAM:



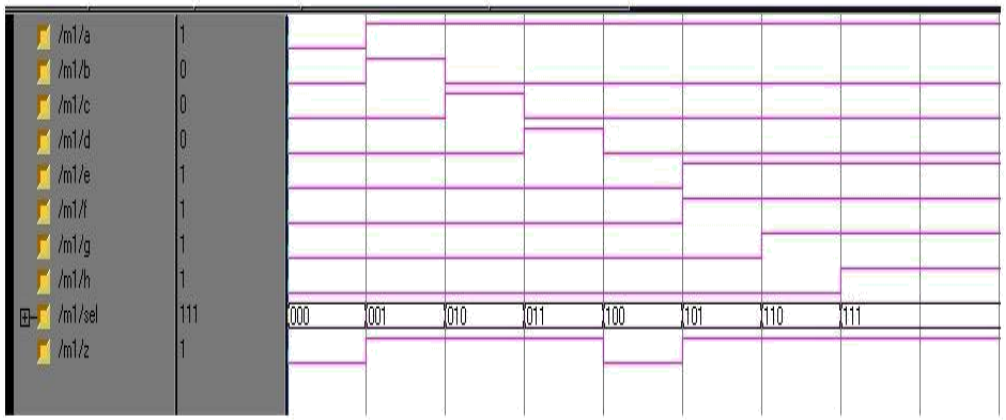
TRUTH TABLE:

A	B	C	D	X
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

VERILOG SOURCE CODE:

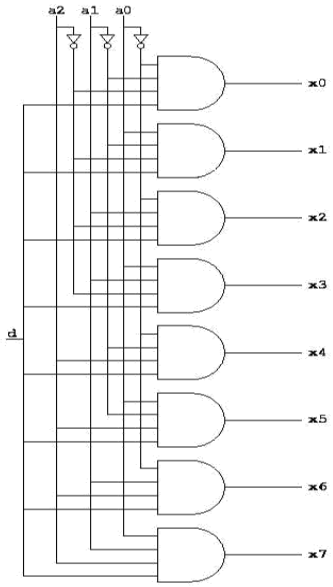
```
module MUX8TO1(sel, A,B,C,D,E,F,G,H, MUX_OUT);
input [2:0] sel;
input A,B,C,D,E,F,G,H;
output  reg MUX_OUT;
always@(A,B,C,D,E,F,G,H,sel)
begin
case(sel)
3'd0:MUX_OUT=A;
3'd1:MUX_OUT=B;
3'd2:MUX_OUT=C;
3'd3:MUX_OUT=D;
3'd4:MUX_OUT=E;
3'd5:MUX_OUT=F;
3'd6:MUX_OUT=G;
3'd7:MUX_OUT=H;
default:; // indicates null
endcase
end
endmodule
```

SIMULATION OUTPUT:



DEMULTIPLEXER:

LOGIC DIAGRAM:



RESULT:

Thus the OUTPUT's of Multiplexers and Demultiplexers are verified by simulating the VHDL and VERILOG code.

EXPERIMENT No: 5 **DATE**

DESIGN OF 4-BIT BINARY TO GRAY CONVERTER

AIM:

To develop the source code for binary to gray converter by using VERILOG and obtained the simulation.

SOFTWARE & HARDWARE:

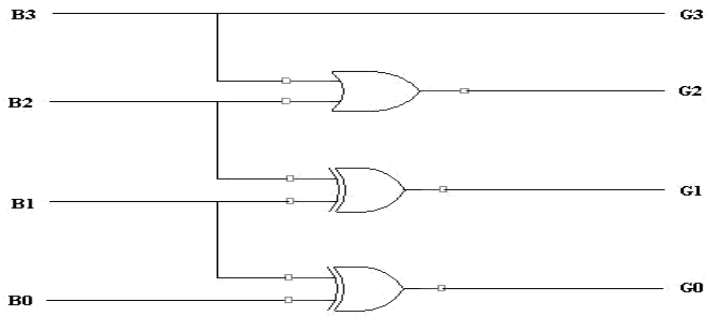
1. XILINX 9.2i
2. FPGA-SPARTAN-3E

CODE CONVERTER (BCD TO GRAY):

TRUTH TABLE:

BCD	GRAY
0000	0000
0001	0001
0010	0011
0011	0010
0100	0110
0101	0111
0110	0101
0111	0100
1000	1100
1001	1101

LOGIC DIAGRAM:

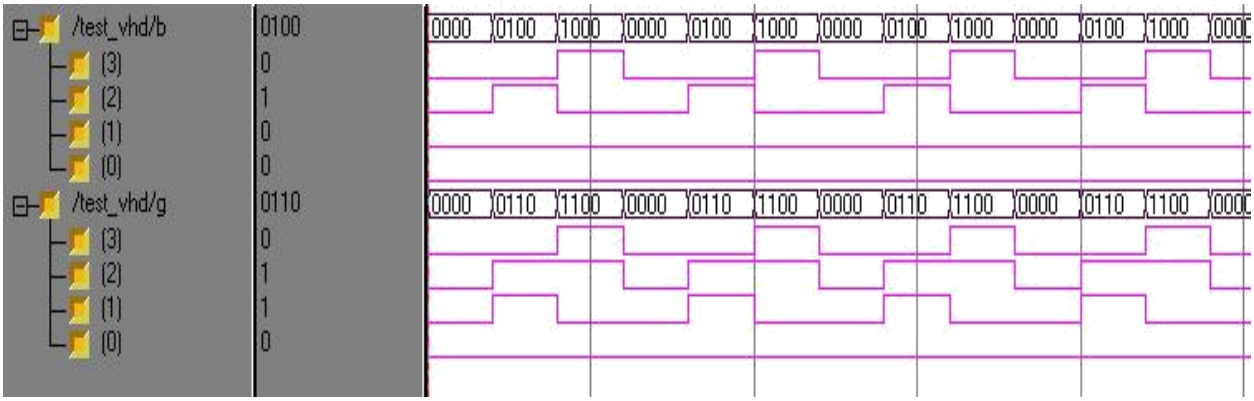


BEHAVIORAL MODELING:

```
module b2g_behv(b, g);
   input [3:0] b;
   output [3:0] g;
         reg [3:0] g;
```

```
        always@(b) begin
        g[3]=b[3];
        g[2]=b[3]^b[2];
        g[1]=b[2]^b[1];
        g[0]=b[1]^b[0];
        end
endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of binary to gray converter are verified by simulating the VERILOG code.

EXPERIMENT No: 6 **DATE**

4-BIT COMPARATOR

AIM:

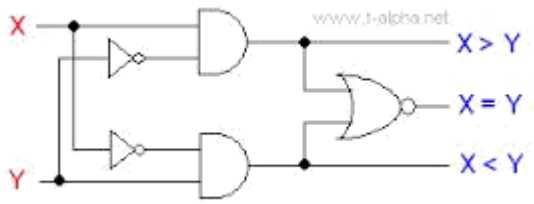
To develop the source code for 4-Bit comparator by using VERILOG and obtained the simulation .

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

4-BIT COMPARATOR:

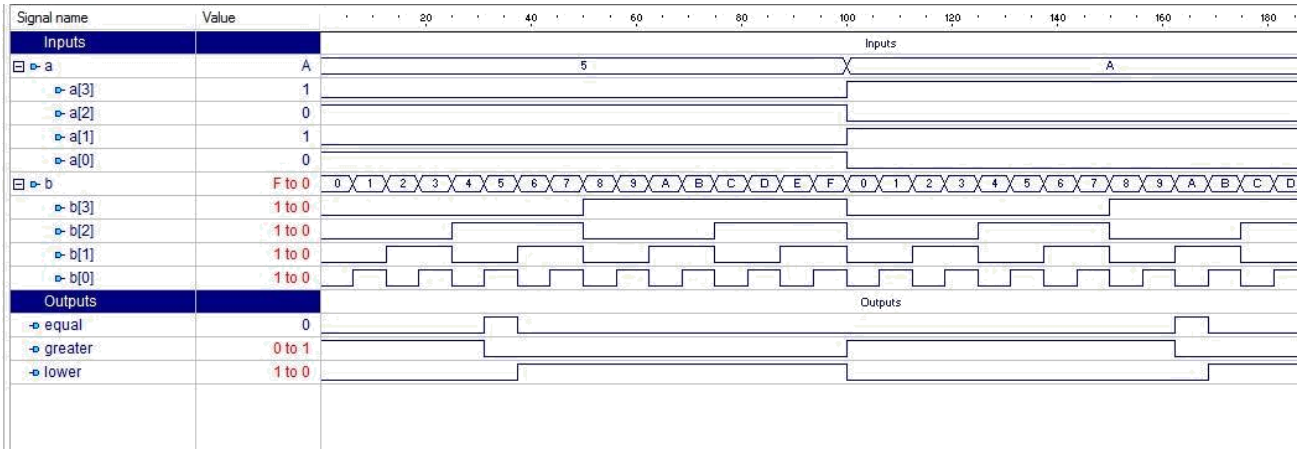
LOGIC DIAGRAM:



VERILOG SOURCE CODE:

```
module comparator ( a ,b ,equal ,greater ,lower );
output equal ;
output greater ;
output lower ;
input [3:0] a ;
input [3:0] b ;
always @ (a or b) begin
 if (a<b) begin
  equal = 0;
  lower = 1;
  greater = 0;
 end else if (a==b) begin
  equal = 1;
  lower = 0;
  greater = 0;
 end else begin
  equal = 0;
  lower = 0;
  greater = 1;
 end
end
endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of 4-bit comparator is verified by simulating the VERILOG code.

EXPERIMENT No: 7 **DATE**

DESIGN OF FULL ADDER USING THREE MODELING STYLES

AIM:

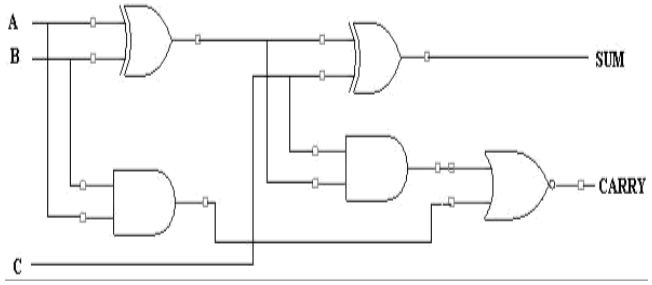
To develop the source code for full adder using three modeling styles by using VERILOG and obtained the simulation.

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

FULL ADDER:

LOGIC DIAGRAM:



TRUTH TABLE:

A	B	C	SUM	CARRY
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

VERILOG SOURCE CODE:

Dataflow Modeling:

```
module fulladddataflow(a, b, c, sum, carry);
   input a;
   input b;
   input c;
   output sum;
   output carry;
assign#2 p=a&b;
assign#2 q=b&c;
assign#2 r=c&a;
assign#4 sum=a^b^c;
assign#4carry =(p1 | p2) | p3;

 endmodule
```

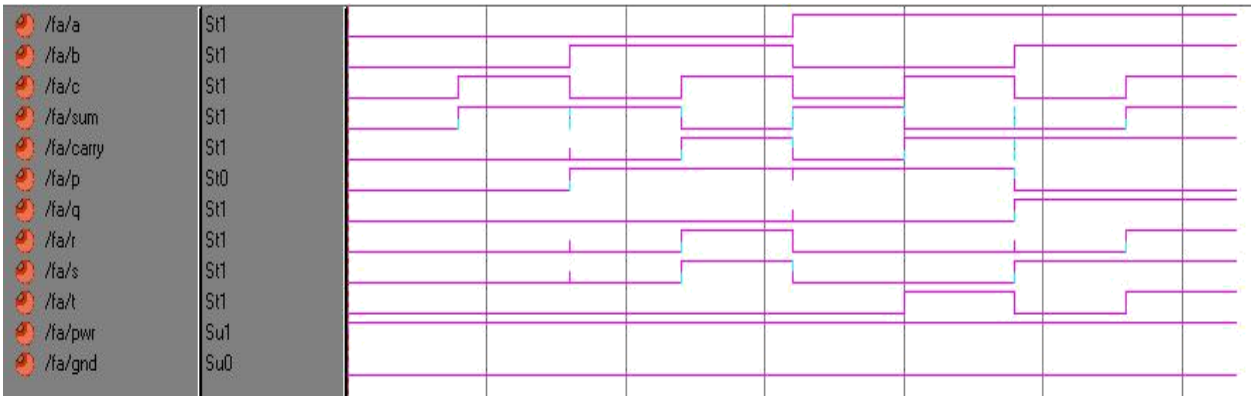
Behavioral Modeling:

```
module fuladbehavioral(a, b, c, sum, carry);
   input a;
   input b;
   input c;
   output sum;
   output carry;
        reg sum,carry;
        reg p1,p2,p3;
        always @ (a or b or c) begin
        sum = (a^b)^c;
        p1=a & b;
        p2=b & c;
        p3=a & c;
        carry=(p1 | p2) | p3;
        end
endmodule
```

Structural Modeling:

```
module fa_struct(a, b, c, sum, carry);
   input a;
   input b;
   input c;
   output sum;
   output carry;
        wire t1,t2,t3,s1
        xor
        x1(t1a,b),
        x2(sum,s1,c);
        and
        a1(t1,a,b),
        a2(t2,b,c),
        a3(t3,a,c);
        or
        o1(carry,t1,t2,t3);
endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of full adder using three modeling styles are verified by simulating the VERILOG code.

EXPERIMENT No: 8 **DATE**

DESIGN OF FLIP FLOPS (SR,JK,D,T).

AIM:

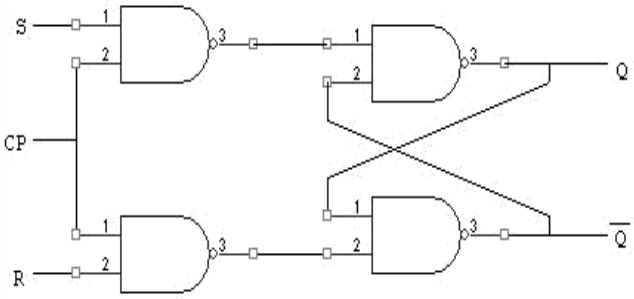
To develop the source code for FLIP FLOPS by using VERILOG and obtained the simulation.

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

SR FLIPFLOP:

LOGIC DIAGRAM:



TRUTH TABLE:

Q(t)	S	R	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	X
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	X

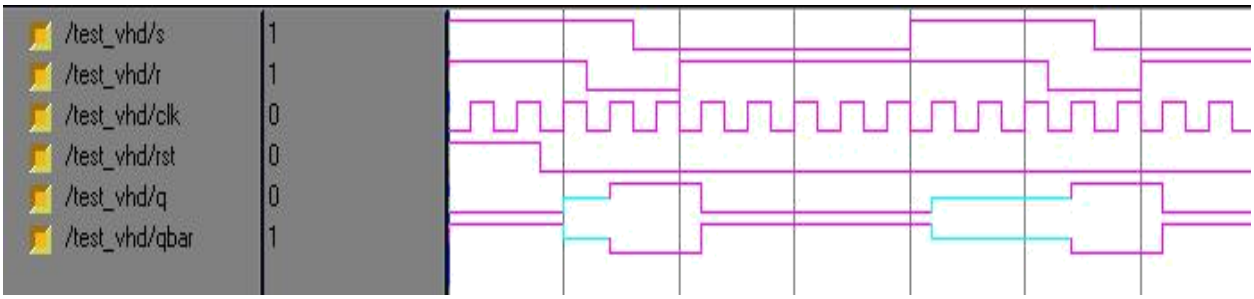
VERILOG SOURCE CODE:

Behavioral Modeling:

```
module srflipflop(s, r, clk, rst, q, qbar);
   input s;
   input r;
   input clk;
   input rst;
   output q;
   output qbar;
         reg q,qbar;
         always @ (posedge(clk) or posedge(rst)) begin
         if(rst==1'b1) begin
         q= 1'b0;qbar= 1'b1;
         end
         else if(s==1'b0 &&    r==1'b0)
          begin
```

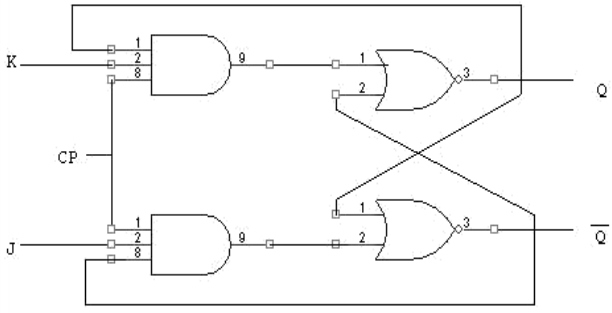
```
        q=q; qbar=qbar;
        end
                else if(s==1'b0 &&      r==1'b1)
                 begin
        q= 1'b0; qbar= 1'b1;
        end
                else if(s==1'b1 &&          r==1'b0)
                 begin
        q= 1'b1; qbar= 1'b0;
        end
        else
        begin
        q=1'bx;qbar=1'bx;
        end
        end
endmodule
```

SIMULATION OUTPUT:



JK FLIPFLOP:

LOGIC DIAGRAM:



TRUTH TABLE:

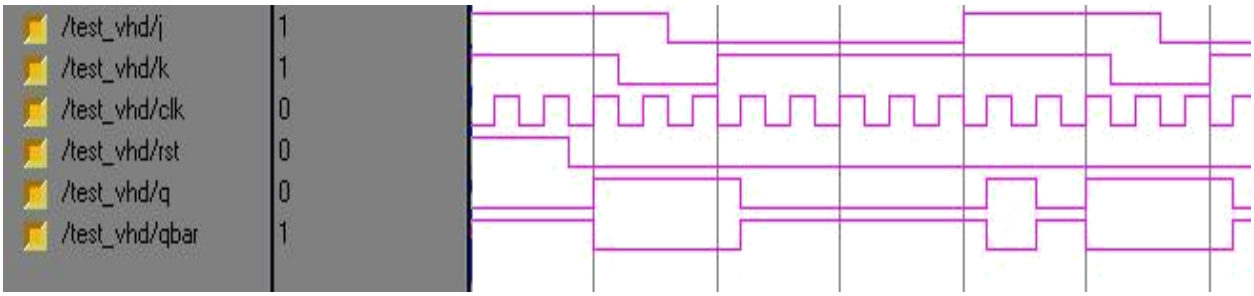
Q(t)	J	K	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

VERILOG SOURCE CODE:

Behavioral Modeling:

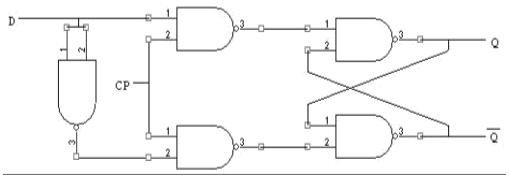
```
module jkff(j, k, clk, rst, q, qbar);
   input j;
   input k;
   input clk;
   input rst;
   output q;
   output qbar;
         reg q;
         reg qbar;
         always @ (posedge(clk) or posedge(rst))  begin
         if (rst==1'b1)
         begin
         q=1'b0;
         qbar=1'b1;
         end
         else if (j==1'b0 && k==1'b0)
         begin
         q=q;
         qbar=qbar;
         end
         else if (j==1'b0 && k==1'b1)
         begin
         q=1'b0;
         qbar=1'b1;
         end
         else if (j==1'b1 && k==1'b0)
         begin
         q=1'b1;
         qbar=1'b0;
         end
         else
         begin
         q=~q;
         qbar=~qbar;
         end
         end
         endmodule
```

SIMULATION OUTPUT:



D FLIPFLOP:

LOGIC DIAGRAM:



TRUTH TABLE:

Q(t)	D	Q(t+1)
0	0	0
0	1	1
1	0	0
1	1	1

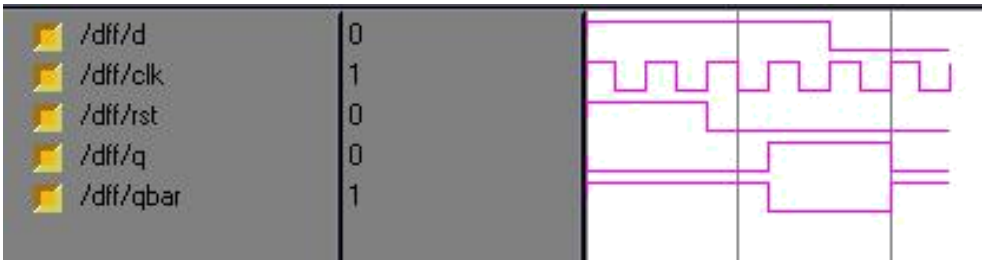
VERILOG SOURCE CODE:

Behavioral Modeling:

```
module dff(d, clk, rst, q, qbar);
   input d;
   input clk;
   input rst;
   output q;
   output qbar;
        reg q;
        reg qbar;
        always @ (posedge(clk) or posedge(rst)) begin
        if (rst==1'b1)
        begin
        q=1'b0;
        qbar=1'b1;
        end
        else if (d==1'b0)
        begin
        q=1'b0;
        qbar=1'b1;
        end
```

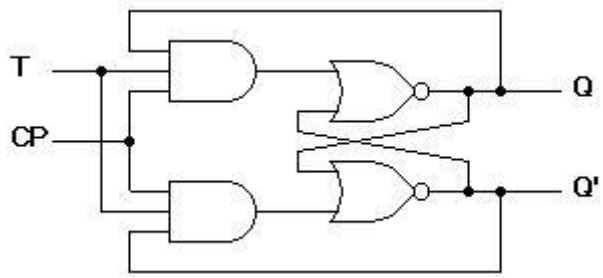
```
        else
        begin
        q=1'b1;
        qbar=1'b0;
        end
        end
  endmodule
```

SIMULATION OUTPUT:



T-FLIP FLOP

LOGIC DIAGRAM:



TRUTH TABALE:

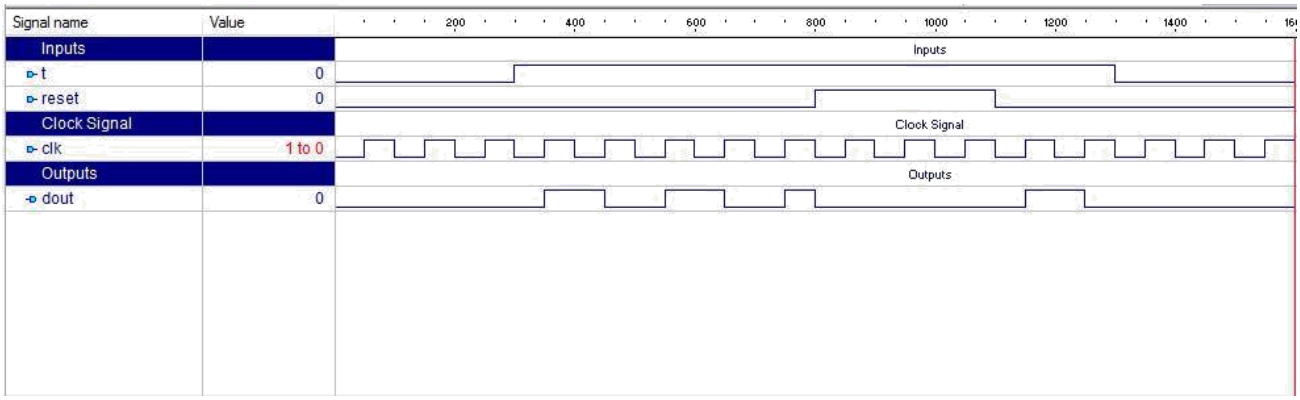
clk	D	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
0	1	Q	$\overline{Q}$
1	0	0	1
1	1	1	0

VERILOG SOURCE CODE:

```
module t_flip_flop ( t ,clk ,reset ,dout );

output dout ;
input t ;
input clk ;
```

```
wire clk ;
input reset ;
initial dout = 0;

always @ (posedge (clk)) begin
 if (reset)
  dout <= 0;
 else begin
  if (t)
   dout <= ~dout;
 end
end

endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of Flip Flops are verified by simulating the VERILOG code.

EXPERIMENT-9 **DATE**

DESIGN OF 4-BIT BINARY COUNTER AND BCD COUNTER

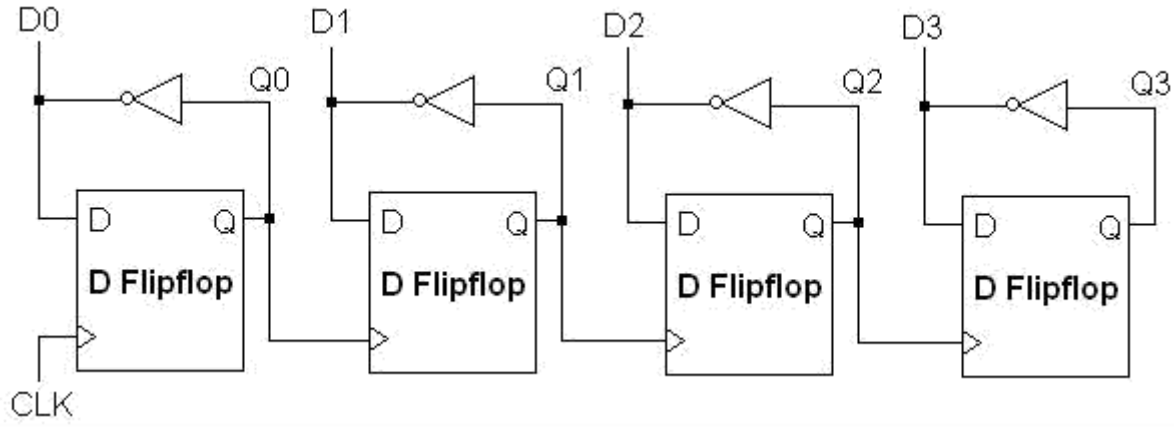
AIM:

To develop the source code for 4-bit binary counter and BCD counter by using VERILOG and obtained the simulation.

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

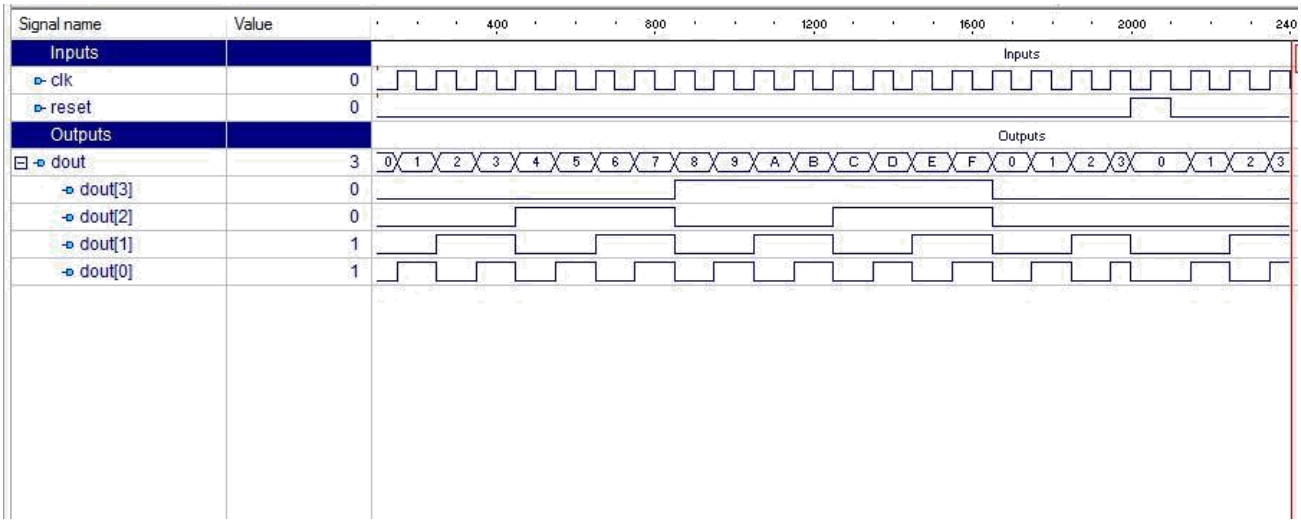
LOGIC DIAGRAM:



VERILOG SOURCE CODE:

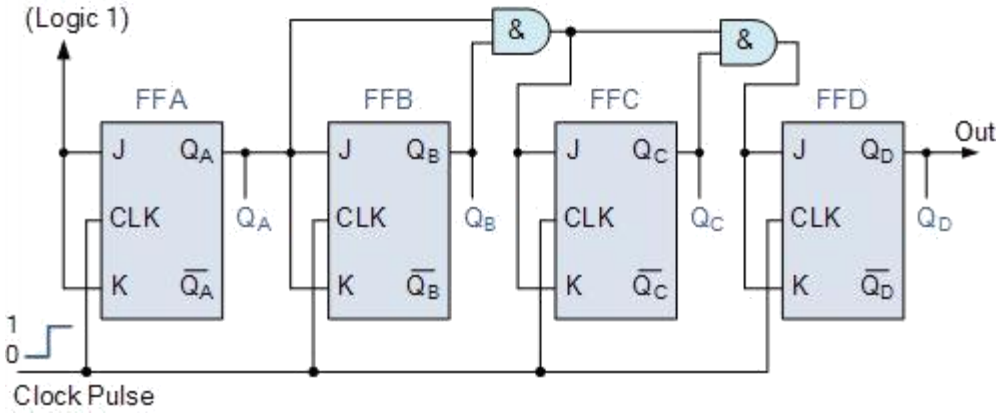
```
module Counter_4Bit ( clk ,reset ,dout );

output [3:0] dout ;
input clk ;
input reset ;

initial dout = 0;

always @ (posedge (clk)) begin
 if (reset)
  dout <= 0;
 else
  dout <= dout + 1;
end
endmodule
```

SIMULATION OUTPUT:



BCD COUNTER

LOGIC DIAGRAM

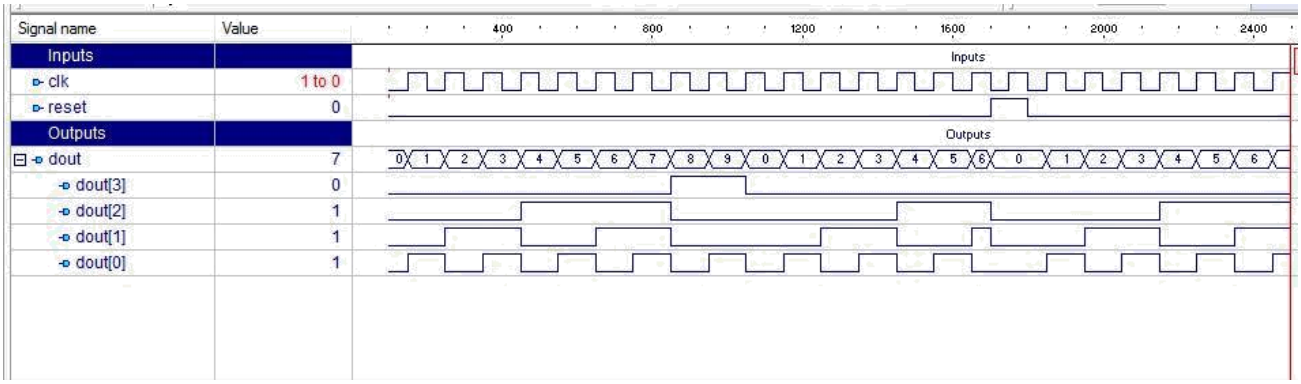


VERILOG SOURCE CODE

```
module BCD_Counter ( clk ,reset ,dout );
output [3:0] dout ;;


input clk ;
input reset ;
initial dout = 0 ;
```

```
always @ (posedge (clk)) begin
 if (reset)
  dout <= 0;
 else if (dout<=9) begin
  dout <= dout + 1;
 end else if (dout==9) begin
  dout <= 0;
 end
end
endmodule
```

SIMULATION OUTPUT:



RESULT:

Thus the OUTPUT's of 4-bit counter and BCD COUNTER using three modeling styles are verified by synthesizing and simulating the VERILOG code

EXPERIMENT :10 **DATE**

FINITE STATE MACHINE DESIGN

AIM:

To develop the source code for finite state machine design by using VERILOG and obtained the simulation

SOFTWARE & HARDWARE:

1. XILINX 9.2i
2. FPGA-SPARTAN-3E

FSM DESIGN

VERILOG SOURCE CODE:

```
module fsm_using_function (
   clock      , // clock
   reset      , // Active high, syn reset
   req_0      , // Request 0
  req_1      , // Request 1
  gnt_0      , // Grant 0
  gnt_1
  );
  //-------------Input Ports-----------------------------
  input   clock,reset,req_0,req_1;
   //-------------Output Ports----------------------------
  output  gnt_0,gnt_1;
  //-------------Input ports Data Type-------------------
    //-------------Output Ports Data Type------------------
  reg     gnt_0,gnt_1;
  //-------------Internal Constants--------------------------
  parameter SIZE = 3           ;
  parameter IDLE  = 3'b001,GNT0 = 3'b010,GNT1 = 3'b100 ;
  //-------------Internal Variables----------------------------
  reg   [SIZE-1:0]          state        ;// Seq part of the FSM
    //----------Code startes Here------------------------
  assign next_state = fsm_function(state, req_0, req_1);
  //----------Function for Combo Logic-----------------
  function [SIZE-1:0] fsm_function;
    input  [SIZE-1:0]  state ;
    input    req_0 ;
    input    req_1 ;
    case(state)
     IDLE : if (req_0 == 1'b1) begin
                 fsm_function = GNT0;
               end else if (req_1 == 1'b1) begin
                 fsm_function= GNT1;
               end else begin
                 fsm_function = IDLE;
               end
     GNT0 : if (req_0 == 1'b1) begin
                 fsm_function = GNT0;
```

```
              end else begin
                fsm_function = IDLE;
              end
   GNT1 : if (req_1 == 1'b1) begin
                fsm_function = GNT1;
          end else begin
                fsm_function = IDLE;
              end
   default : fsm_function = IDLE;
  endcase
endfunction
//----------Seq Logic-----------------------------
always @ (posedge clock)
begin : FSM_SEQ
  if (reset == 1'b1) begin
    state <=  #1  IDLE;
  end else begin
    state <=  #1  next_state;
  end
end
//----------Output Logic-----------------------------
always @ (posedge clock)
begin : OUTPUT_LOGIC
if (reset == 1'b1) begin
  gnt_0 <=  #1  1'b0;
  gnt_1 <=  #1  1'b0;
end
else begin
  case(state)
    IDLE : begin
                gnt_0 <=  #   1'b0;
                gnt_1 <=  1   1'b0;
                          #
                          1
             end
   GNT0 : begin
                 gnt_0 <=  #1  1'b1;
                 gnt_1 <=  #1  1'b0;
              end
   GNT1 : begin
                 gnt_0 <=  #1  1'b0;
                 gnt_1 <=  #1  1'b1;
              end
   default : begin
                  gnt_0 <=  #1  1'b0;
                  gnt_1 <=  #1  1'b0;
                end
  endcase
end
end // End Of Block OUTPUT_LOGIC

endmodule // End of Module arbiter
```

RESULT:

Thus the OUTPUT's of finite state machine design is verified by simulating the VERILOG code.