Experiment No.3: Project-Based Learning Activity: Introduction to Circuit Simulation

Objective:	To develop students' ability to effectively use circuit simulation tools for modeling, analyzing, and verifying the behavior of electrical circuits, like RC series circuit, RLC series circuit, implementation of Network theorems like Mesh and Nodal analysis, enabling a visual understanding of circuit operation, verifying circuit parameters, system responses without the need for physical hardware.

Outcome:	• **Demonstrate proficiency in using a circuit simulation tool.**
	  – Use and experiment with the circuit simulation tool, "*Sequel*" developed at EE department IIT – Bombay.
	  – Explore the simulation tool for various electronic and circuits components provided with it.
	  – Construct and simulate RC series circuit using *Sequel and* analyze the behavior of RC charging.
	  – Construct and simulate RLC series circuit using *Sequel* and analyze the behavior of this circuit by plotting its frequency response.
	• **Simulate and verify key electrical theorems using *Sequel* circuit simulation tool.**
	  – Use the circuit simulation tool *Sequel* to design the given fundamental circuits for verifying key network theorems like Mesh and Nodal analysis.
	  – Simulate the given circuit to verify the circuit parameters like Mesh currents and Node Voltages.
	  – Verify that the observed values of voltages and currents match with those of the calculated values as provided.
	  – Populate observation tables with measured values and reflect on possible deviations from expected values.

Tasks
Problem
Statement
 	
Procedure
Observation
Discussion,
and
Report
	1) **Demonstrate proficiency in using SEQUEL circuit simulation tool.:**
	  – *Construct the given RC series circuit*
	  – *Select the different values for R and C, calculate RC time constant and observe the graph plotted for Voltage vs Time.*
	
	Refer to this introductory tutorial video, RC tutorial

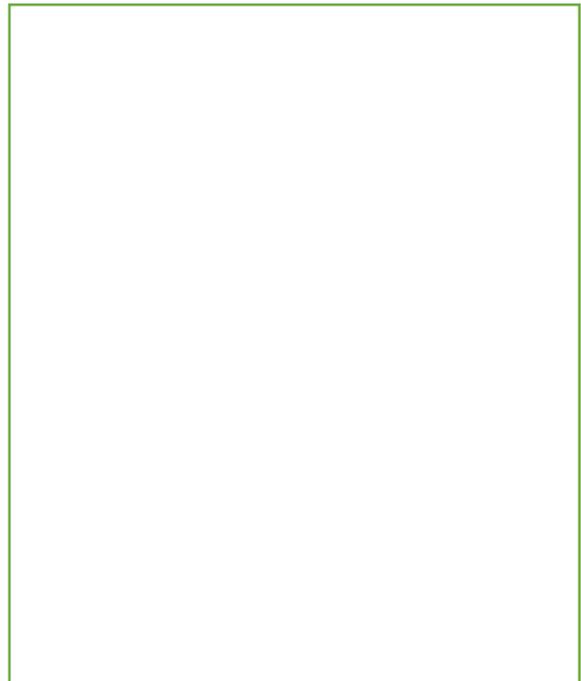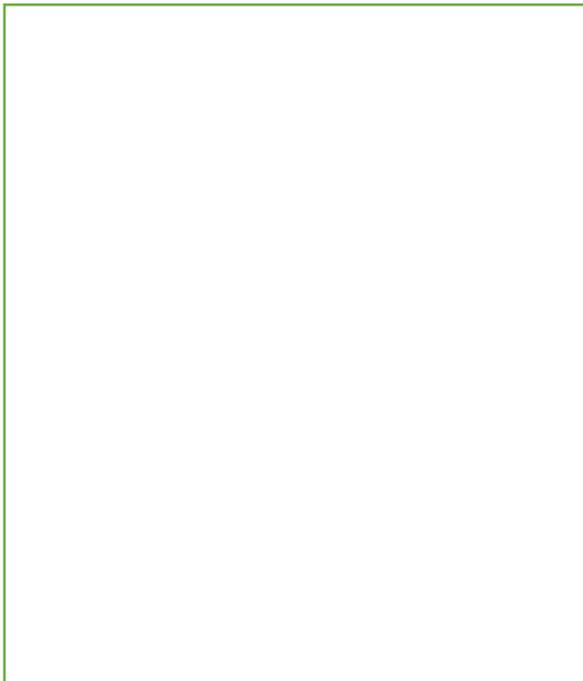Follow Lab instructions, and complete the observation table below:

S. No.	R	C	Time Constant (Calculated) $t = R \times C$	Time Constant (Observed from Graph)*
1.	1KΩ	1μf		
2	10KΩ	100 nF		

*Time constant from graph = time at which graph reaching 70% max value of Voltage (y axis)

Observed Graph

Case1:   Case2:

Tasks
Problem
Statement

Procedure
Observation
Discussion,
and
Report
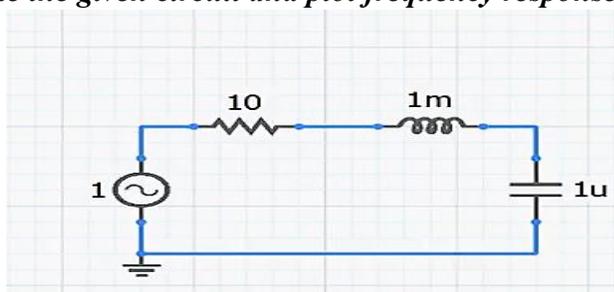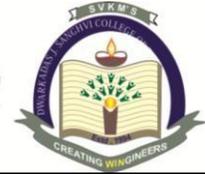
2. Construct given RLC series circuit using *Sequel*. Simulate, and analyze the behavior of this circuit by plotting its frequency response.
   - *Construct the given RLC series circuit.*
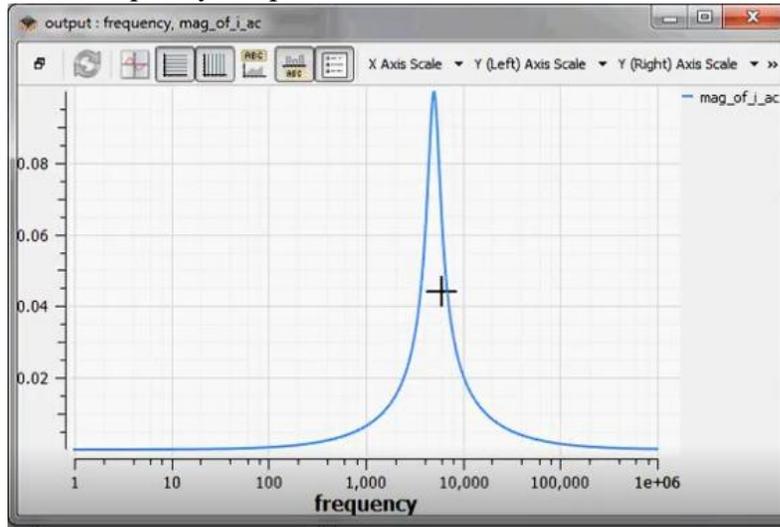   - *Simulate the given circuit and plot frequency response.*

Refer to this introductory video, rlc tutorial.

Follow Lab instructions, and complete the observation table below:

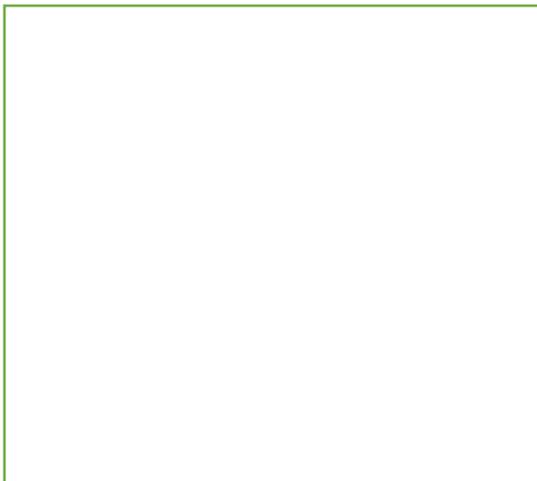| S.No.	L	C	Resonant Frequency (Calculated) $f0 = \dfrac{1}{2\pi\sqrt{LC}}$	Resonant Frequency (Observed from Graph)
1.	1mH	1µf		
2	30mH	1µf		

Ideal Frequency Response:
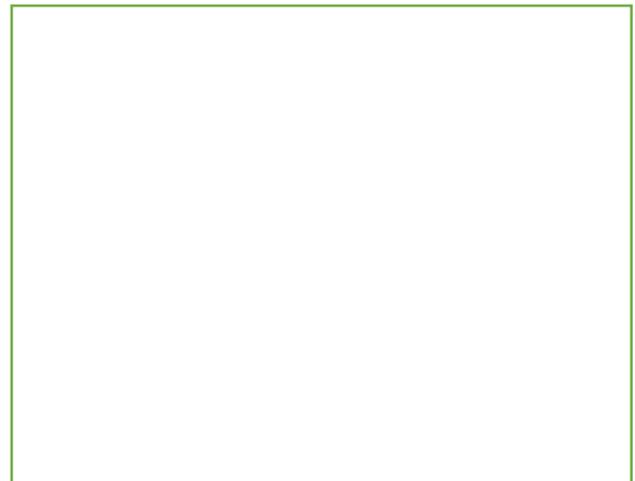

Observed frequency response:
Case1:   Case2:

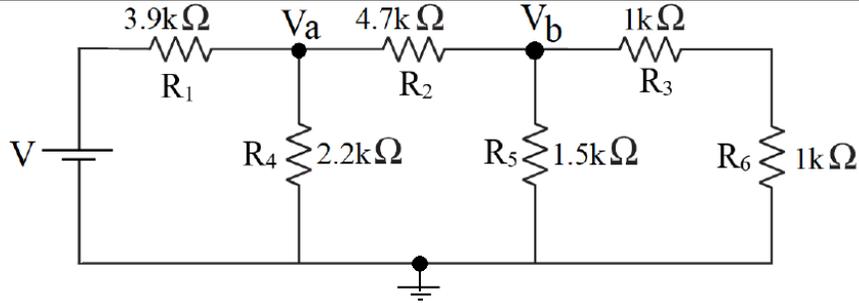|	
	Result: • Constructed Circuit using Sequel: Yes/No • Able to simulate the circuit: Yes/No • Able to plot graph for case1 and case 2: Yes/No **Faculty Signature & Date:** _____

Tasks
Problem
Statement
 	
Procedure
Observation
Discussion,
and
Report
	3. Simulate and verify key electrical theorems using *Sequel* circuit simulation tool.
	   – Construct the given circuits for verifying network theorems - Mesh analysis.
	   – Simulate the given circuit to verify the circuit parameters like Mesh currents and Voltages.
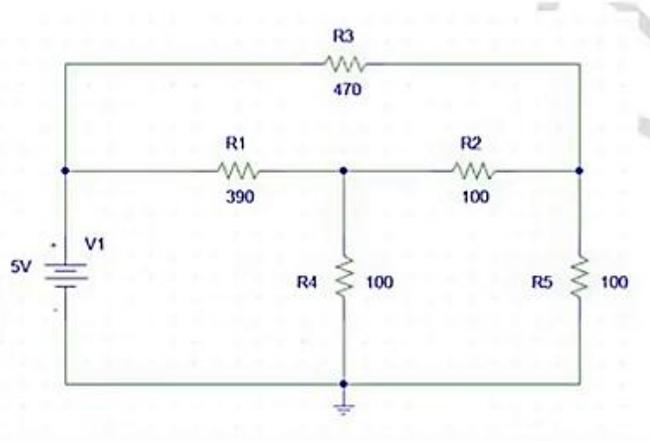	
	Refer to this introductory video, **Mesh Analysis tutorial**

	Follow Lab instructions, and complete the observation table below: a) Mesh Analysis _(table below)_ *Refer the introductory video and note down theoretical values for mesh currents

	Simulate and verify key electrical theorems using *Sequel* circuit simulation tool.
	
	   – Construct the given circuits for verifying network theorems - Nodal analysis.
	   – Simulate the given circuit to verify the circuit parameters like Nodal Voltages.

Supply Voltages.	Current Through 390Ω Resistor		Voltage across 390Ω Resistor	
	Calculated	Observed	Calculated	Observed
5V				

Follow Lab instructions, and complete the observation table below:

    b) Nodal Analysis

Supply Voltages.	Node Voltages (Calculated Values)		Node Voltages (Observed Values)	
	VA	VB	VA	VB
5V				

Result:

- Constructed Circuit using Sequel: Yes/No
- Able to simulate the circuit: Yes/No
- Observed and calculated values are matching in both cases: Yes/No

Faculty Signature & Date: _____

|	
Tasks Problem Statement **Procedure Observation Discussion, and Report**	**4. Construct the Given logic gates in Sequel and verify its truth table** - **AND GATE:** An AND gate has two or more inputs but only one output. The output assumes the logic 1 state only when each one of its inputs is at the logic 1 state. The output assumes a logic 0 state even if one of its inputs is at logic 0 state. AND gate is also called an „all or nothing" gate. The logic symbol & truth table of the two input AND gate are shown below respectively. The symbol for AND operation is ".". With input variables A & B the Boolean expression for output can be written as;

$$X = A.B$$
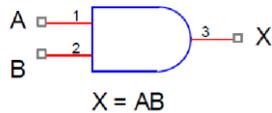
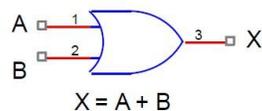| Input		Output
A	B	**X**
0	0	0
0	1	0
1	0	0
1	1	1

$X = AB$

Refer to this introductory video, AND_Gate_tutorial

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of AND Gate:

- **OR GATE:**

Like an AND gate, an OR gate may have two or more inputs but only one output. The output assumes the logic 1 state, even if one of its inputs is in logic 1 state. Its output assumes logic 0 state, only when each one of its inputs is in logic 0 state. OR gate is also called an "any or all" gate. It can also be called an inclusive OR gate because it includes the condition, "both the input can be present". The logic symbol & truth table of two input OR gate are shown in figure below. The symbol for OR operation is "+". With input variables A & B the Boolean expression for output can be written as;

$$X = A + B$$

$X = A + B$

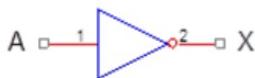| Input		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of OR Gate:

- **NOT GATE:**

A NOT gate is also known as an inverter, has only one input and only one output. It is a device whose output is always the complement of its input. That is the output of a not gate assumes the logic 1 state when its input is in logic 0 state and assumes the logic 0 state when its input is in logic 1 state. The logic symbol & truth table of NOT gate are shown in figure below.

The symbol for NOT operation is, " ‾(bar)". With input variable A, the Boolean expression for output can be written as:

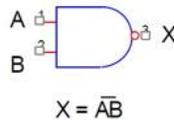$$X = \bar{A} \; ; \text{This is read as "X is equal to A bar".}$$

Input	Output
A	X
0	1
1	0

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of NOT Gate:

- **NAND GATE:**

NAND gate is universal gate. It can be used to perform all the basic logic functions. NAND means NOT AND that is, AND output is 'NOT'ed or complemented. So a NAND gate is a combination of an AND gate and a NOT gate. The output is logic 0 level, only when each of its inputs assumes a logic 1 level. For any other combination of inputs, the output is logic 1 level. NAND gate is equivalent to an OR gate with bubbled inputs. The logic symbol & truth table of two input NAND gate are shown in figure below.   With input variables A & B the Boolean expression for output can be written as:
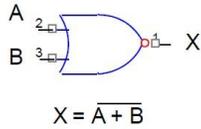
$$X = \overline{A \cdot B}$$

Input		Output
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

$X = \overline{AB}$

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of NAND Gate:

- **NOR GATE:**

NOR gate is a universal gate. It can be used to perform all the basic logic functions. NOR means NOT-OR that is, OR output is 'NOT'ed or complemented. So NOR gate is combination of an OR gate and a NOT gate. The output is logic 1 level, only when each of its inputs assumes a logic 0 level. For any other combination of inputs, the output is logic 0 level. NOR gate is equivalent to an AND gate with bubbled inputs. The logic symbol & truth table of two inputs NOR gate are shown in figure below. With input variables A & B the Boolean expression for output can be written as:

$$X = \overline{A + B}$$

Input		Output
A	**B**	**X**
0	0	1
0	1	0
1	0	0
1	1	0

$X = \overline{A + B}$

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of NOR Gate:

- **EXCLUSIVE-OR (X-OR) GATE:**

An X-OR gate is a two input, one output logic circuit, whose output assumes a logic 1 state when one and only one of its two inputs assumes a logic 1 state. Under the condition when both the inputs are same either 0 or 1, the output assumes a logic 0 state. Since an X-OR gate produces an output 1 only when the inputs are not equal, it is called as, an anti-coincidence gate or inequality detector. The output of an X-OR gate is the modulo sum of its two inputs.

The logic symbol & truth table of two input X-OR gate are shown in figure below.

The symbol for X-OR operation is "$\oplus$".

With input variables A & B the Boolean expression for output can be written as:
$$Y = A\overline{B} + \overline{A}B \quad \text{or written as X= A} \oplus \text{B}$$
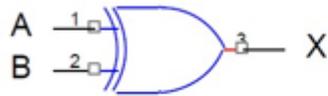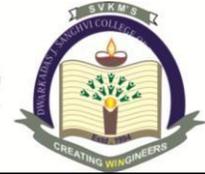
Input		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

$X = A \oplus B$

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of EX-OR Gate:

Result:
- Constructed Circuit using Sequel: Yes/No
- Able to simulate the circuit: Yes/No
- Observed and calculated and/or expected results match in each case above: Yes/No:

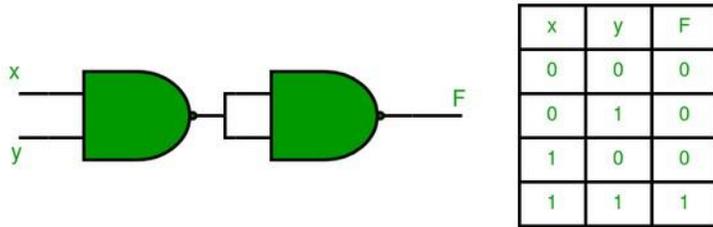**Faculty Signature & Date:** _____

| Tasks Problem Statement	**5. Realisation of the basic gates using universal gates**
	1. Implementation of **AND Gate** using Universal gates.
	a) Using NAND Gates
Procedure Observation Discussion, and Report	The AND gate can be implemented by using two NAND gates in the below fashion:

x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

b) Using NOR Gates

Implementation of AND gate using only NOR gates as shown below:
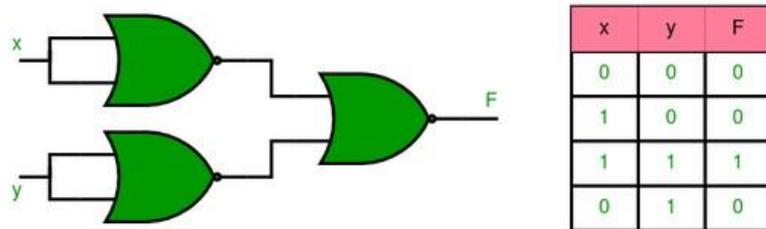
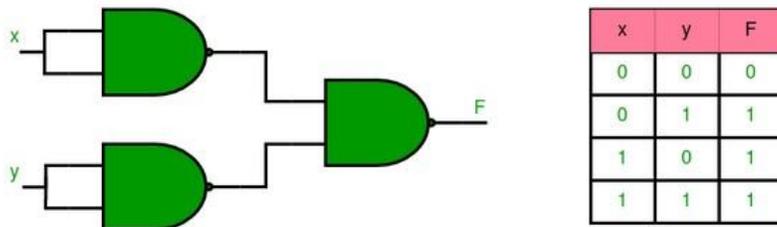| x	y	F
0	0	0
1	0	0
1	1	1
0	1	0

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of AND Gate implemented using NAND and NOR gates.

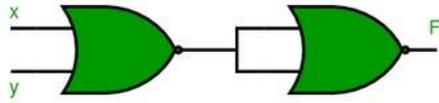2. Implementation of **OR Gate** using Universal gates.

a) Using NAND Gates

The OR gate can be implemented using the NAND gate as below:

x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

b) Using NOR Gates

Implementation of OR gate using two NOR gates as shown in the picture below:
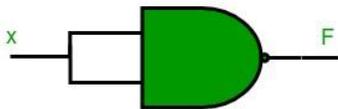
x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of OR Gate implemented using NAND and NOR gates.

3. Implementation of **NOT Gate** using Universal gates.

a) Using NAND Gates

x	F
0	1
1	0

b) Using NOR Gates

x	F
0	1
1	0

Follow Lab instructions, and paste here the input, output waveform, indicating truth table of NOT Gate implemented using NAND and NOR gates.
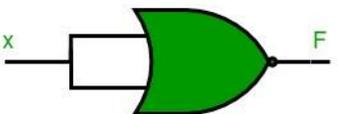
Results:	• Constructed Circuit using Sequel: Yes/No • Able to simulate the circuit: Yes/No • Observed and calculated and/or expected results match in each case above: Yes/No **Faculty Signature & Date:** _____
	Test your understanding questions: (oral answers to mentors) 1. What is SEQUEL, and what are its main features in circuit simulation? 2. What are the advantages of using simulation tools like SEQUEL before hardware implementation? 3. Define an RC series circuit. What components does it consist of, and how do they behave in series? 4. Explain the concept of frequency response. Why is it important in circuit analysis?
Conclusion:	_____ _____ _____ _____