UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name: Values & Ethics in Profession **Subject Code-HU301**
Year: 2nd Year **Semester: Third**

Module Number	Topics	Number of Lectures
1	**Introduction:**	**19L**
	Rapid Technological growth and depletion of resources, Reports of the Club of Rome. Limits of growth: Sustainable development	3
	EnergyCrisis: Renewable Energy Resources Environmental degradation and pollution. Eco-friendly Technologies. Environmental Regulations, Environmental Ethics	5
	Appropriate Technology Movement of Schumacher; later developments Technology and developing notions. Problems of Technology transfer, Technology assessment impact analysis.	6
	Human Operator in Engineering projects and industries. Problems of man, machine, interaction, Impact of assembly line and automation. Human centeredTechnology.	5
2	**EthicsofProfession:**	**9L**
	Engineering profession: Ethical issues in Engineering practice, Conflicts between business demands and professionalideals.	3
	Social and ethical responsibilities of Technologists. Codes of professional ethics. Whistle blowing and beyond.	6
	Profession and Human Values	**8L**
3.		
	Values Crisis in contemporary society Nature of values: Value Spectrum of a good life	3
	Psychological values: Integrated personality; mental health Societal values: The modern search for a good society, justice, democracy, secularism, rule of law, values in Indian Constitution. Aesthetic values: Perception and enjoyment of beauty, simplicity, clarity Moral and ethical values: Nature of moral judgements; canons of ethics; ethics of virtue; ethics of duty; ethics of responsibility.	5

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name: Numerical Methods **Subject Code-M(CS)301**
Year: 2nd Year **Semester: Third**

Module Number	Topics	Number of Lectures
1	**Approximation in numerical computation:**	**4L**
	Approximation of numbers	1
	Types of errors	2
	Calculation of errors	1
2	**Interpolation:**	**6L**
	Finite differences	1
	Newton forward/backward interpolation	2
	Lagrange's method	1
	Newton's divided difference Interpolation	2
3.	**Numerical integration:**	**3L**
	Trapezoidal rule	2
	Simpson's 1/3 rule	1
4	**Numerical solution of a system of linear equations:**	**6L**
	Gauss elimination method	1
	Matrix inversion	1
	LU Factorization method	2
	Gauss-Seidel iterative method	2
5	**Numerical solution of Algebraic equation:**	**5L**
	Bisection method	2
	Regula-Falsi method	1
	Newton-Raphson method	2
6	**Numerical solution of ordinary differential equation:**	**8L**
	Euler's method	2
	Runge-Kutta methods	2
	Predictor- Corrector methods	2
	FiniteDifference method	2

Assignment:

Module-1:

1. Find the relative error if 2/3 is approximated to 0.667.
2. Find the percentage error if 625.483 is approximated to three significant figures.
3. Find the relative error in taking ® = 3.141593 as 22/7.
4. The height of an observation tower was estimated to be 47 m, whereas its actual height was 45 m. calculate the percentage relative error in the measurement.

5. Two numbers are 3.5 and 47.279 both of which are correct to the significant figures given. Find their product.

Module-2:

1. Apply Newton's backward Interpolation to the data below, to obtain a polynomial of degree4 in x

x:	1	2	3	4	5
$f(x)$:	1	-1	1	-1	1

2. Using Newton's backward Interpolation, find the value of $f(2)$ from the following table:

x:	1	3	4	5	6	7
$f(x)$:	2.68	3.04	3.38	3.68	3.96	4.21

3. Using Newton's Forward Interpolation, the area A of a circle of diameter d.

d:	80	85	90	95	100
A:	5026	5674	6362	7088	7854

Calculate the area of a circle of diameter 105.

4. Estimate the value of $f(22)$ and $f(42)$ from the following available data:

x:	20	25	30	35	40	45
$f(x)$:	354	332	291	260	231	204

Using Newton's Forward Interpolation

5. Find $f(x)$ as a polynomial in x for the following data by Newton's divided difference method:

x:	-4	-1	0	2	5
$f(x)$:	1245	33	5	9	1335

6. Using Newton's divided difference method to find $f(x)$ from the following available data:

x:	0	1	2	4	5	6
$f(x)$:	1	14	15	5	6	19.

Module-3:

1. Apply trapezoidal rule to find the integral $I = \int_0^1 \sin^{®} x\,dx.$
2. Find, from the following table the area bounded by the curve and the x-axis from $x = 7.47$ to $x = 7.52$,
   $f(7.47) = 1.93, f(7.48) = 1.95, f(7.49) = 1.98, f(7.50) = 2.01,$
   $f(7.51) = 2.03, f(7.52) = 2.06.$
3. Evaluate $I = \int_0^1 \frac{1}{1+x^2}\,dx,$ correct to three decimal places and also find the approximate value of ®.
4. A solid of revolution is formed by rotating about the x-axis the area between the x-axis, the lines $x = 0$ and $x = 1$ and a curve through the points with the following coordinates:
   $(0,1),(0.25,0.9896)(0.5,0.9589),(0.75,0.9089),(1,0.8415).$

Module-4:

1. Solve the following system of equations:

$4x + y + z = 4$

$x + 4y - 2z = 4$,

$3x + 2y - 4 = 6$

by matrix-inversion method.

2. Solve the above system by matrix-inversion method:

   $x + y - z = 2$

   $2x + 3y + 5z = -3$

   $3x + 2y - 3z = 6$

3. The following system of equations are given:

   $4x + y + z = 4$

   $x + 4y - 2z = 4$, Solve the above system by LU decomposition method.

   $3x + 2y - 4 = 6$

4. Solve the given system of equations by LU decomposition method:

   $x + y - z = 2$

   $2x + 3y + 5z = -3$

   $3x + 2y - 3z = 6$

Module-5:

1. Find the root of the following equations correct three decimal places by the Regulafalsi method: $x^3 + x - 1 = 0$.
2. Using Regulafalsi method, compute the real root of the following equation correct to four decimal places: $xe^x = 2$.
3. Find the root of the following equations correct three decimal places by the Regulafalsi method: $x^6 - x^4 - x^3 - 1 = 0$.
4. Find the root of the following equations correct three decimal places by the bisection method : $x - e^x = 0$
5. Find the root of the following equations, using the bisection method correct three decimal places: $x - \cos x = 0$
6. Using the bisection method to find a root of the equation to four decimal places: $x^3 - 9x + 1 = 0$

Module-6:

1. Using Runge-kutta method of order 4, find $y(0.2)$ given that $\frac{dy}{dx} = 3x + \frac{1}{2}y, y(0) = 1$ taking $h = 0.1$.
2. Using Runge-kutta method of order 4, compute $y(0.2)$ and $y(0.4)$ from $10\frac{dy}{dx} = x^2 + y^2, y(0) = 1$ taking $h = 0.1$.
3. Using Milne's predictor-corrector method to obtain the solution of the equation $\frac{dy}{dx} = x - y^2$ at $x = 0.8$ given that $y(0) = 0.0000$, $y(2) = 0.0200$, $y(4) = 0.0795$ $y(6) = 0.1762$.

4. Given $2\frac{dy}{dx} = (1+x^2)y^2$ and $y(0)=1$, $y(0.1)=1.06$, $y(0.2)=1.12$, $y(0.3)=1.21$, evaluate $y(0.4)$ by Milne's predictor-corrector method.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name: Analog Electronic Circuits **Subject Code-EC(EE)301**
Year: 2nd Year **Semester: Third**

Module Number	Topics	Number of Lectures
1	**Introduction:**	**2L**
	1. Active & Passive Devices,overview of analog circuits.	1
	2. application ofanalog circuits-implementation etc.	1
2	**DIODE AND THEIR APPLICATIONS**	**3L**
	1. Characteristics of ideal & real diodes,diode circuits rectifiers	1
	2. clipping,clamping,special types of diodes & their applications systems	1
	3. schottky,varactor,photodiodes,LEDs	1
3.	**BIPOLAR JUNCTION TRANSISTOR AND APPLICATIONS**	**8L**
	1. Characteristics of BJT; Ebers-Moll equations and large signal models, inverse mode of operation, early effect; BJT as an amplifier and as a switch	2
	2. DC biasing of BJT amplifier circuits; small signal operations and models, Single state BJT amplifiers – CE, CB and CC amplifiers	2
	3. high frequency models and frequency response of BJT amplifiers, Basic design in discrete BJT amplifiers; complete design examples	**2**
	4. Basic BJT digital logic inverter, SPICE modeling of BJT and amplifier circuits	**2**
4	**MOSFETS AND CIRCUITS:**	**6L**
	1. MOSFET -operational Characteristics; PMOS, NMOS and CMOS current voltage characteristics, DC analysis; Constant Current Sources and Sinks	2
	2. MOSFET as an Amplifier and as a Switch; Biasing on MOS Amplifiers, Small Signal Operation of MOS amplifiers	2
	3. Common-source, common gate and source Follower Amplifiers, CMOS amplifiers; MOSFET Digital logic inverters, voltage transfer characteristics, SPICE modeling of MOSFET circuits	2
5	**VOLTAGE AND POWEWR AMPLIFIER:**	**4L**
	1. Classification amplifiers; Class A, Class B, Class AB Class C – Circuit operation, transfer characteristics, power dissipation, efficiency	2

	2. Practical BJT and MOS power transistors; thermal resistance, heat sink design; IC power amplifiers	2
6	**FEEDBACK IN AMPLIFIERS AND OSCILLATOR:**	**5L**
	1. Feedback concept and definition; Four basic feedback topologies; Analysis of Series-shunt, shunt-shunt and shunt-series feedback amplifiers; stability in feedback amplifiers	2
	2. frequency compensation; principle of sinusoidal oscillators and barkhausen criterion, Active-RC and Active-LC sinusoidal oscillators	1
	3. Wien Bridge; Phase-Shift; Quadrature Oscillators; Crystal Oscillators, application in voltage regulation	2
7,8	**DIFFERENTIAL AMPLIFIER:**	**3L**
	1. Advantages of differential amplifiers; MOS and BJT differential pair; Small signal and large signal operation of differential pairs	1
	2. Parameters and non-ideal characteristics of differential amplifiers	1
	3. differential amplifier with active load frequency response; spice simulation examples	1
	OPERATIONAL AMPLIFIER AND ITS APPLICATIONS:	**5L**
	1. Concept of operational amplifiers; Ideal operational amplifier parameters	1
	2. Inverting and non-inverting configurations; Common OPAMP IC	1
	3. Gain-frequency and Slew rate, SPICE modeling and simulation examples	1
	4. Instrumentation amplifiers; Integrators, Differentiators	**1**
	5. Logarithmic Amp; Multipliers; Comparators; Schmitt triggers	1
9	**FILTERS AND TUNED AMPLIFIERS:**	**4L**
	1. Filter characteristics and specifications	1
	2. First and Second Order Filter functions; First-order and second order filter network using OPAMPS	1
	3. Tuned Amplifiers; Basic principle; amplifiers with multiple tuned circuits	1
	4. Synchronous and Stagger tuning; RF amplifiers considerations	1
	WAVEFORM GENERATION AND	**4L**

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

10	**SHAPING CIRCUIT:**	
	1. Multivibrators – Astable, monostable and bistable circuits	1
	2. bistable circuit as memory element comparator generation of square, triangular waveform, standardized pulse using AMV and MMV	1
	3. Application of 555 timer	1
	4. VCO,PLL	1
Total Number Of Hours = 44		

Faculty In-Charge HOD, ECE Dept.

Assignment:

Module-1,2(DIODE CIRCUIT):

1. **A500µFcapacitor provides a load current of 200 mA at 8% ripple; calculate the pick rectified voltage obtained from the 60Hz supply and the dc voltage across the filter capacitor**.
2. **Calculate the size of the filter capacitor needed to obtained a filtered voltage with 7% ripple at a load of 200mA.The full wave rectified voltage is 30v, and the supply is 60Hz**
3. **Using necessary circuits and waveforms illustrate and explain positive and negative clamping of voltages.**
4. **Sketch the volt-ampere characteristics of zener diode. Indicate the knee on the curve and explain its significance. What happens when the current in zener decreases below the knee current?**

Module-3(BIPOLAR JUNCTION TRANSISTOR AND APPLICATIONS):

1. **Find the Q point of a self–bias transistor circuit with the following specification:- V_{cc}=22.5volt,R_L= 5.6KΩ,R_E= 1KΩ,R_1= 90KΩ, R_2=10KΩ V_{BE}=0.7 volt and β=55 Assume $I_b >> I_{c0}$**
2. **A particular BJT operating at Ic=2mA has Cµ=Ipf, Cπ=10 pf and β= 150.Whatare f_t & fβ for this situation?**
3. **Draw the circuits of the various transistor configurations. Why common emitter configuration is mostly used? Give its typical uses.**
4. **What is transistor biasing? What are the basic conditions which are to be necessarily fulfilled for achieving faithful amplification of input signal in transistor amplifiers?**
5. **What are the advantages of the FET over a conventional bipolar junction transistor? Define pinch off voltage, transconductance, amplification factor and drain resistance of a FET. Explain with the help of circuit diagram how an FET is used as a voltage amplifier and as voltage dependent resistor (VDR).**

Module-4,5 (MOSFETS AND VOLTAGE AND POWEWR AMPLIFIER):

1. **A CE-RC coupled amplifier uses transistors with the following h-parameters: hfe=50,hie=1100Ω, hoe=10*10-6 mhos, hre=2.5*10-4.The value of gmat the**

operating point is 200mhos.The biasing resistor R1&R2 may be neglected being large in comparison with RI. The load resistor Rc= 5KΩ. Let the total shunt capacitance C=200μf in the input Ckt. And the coupling capacitor Cc=7μf. Calculate for one stage of the amplifier(a) mid band current gain (b) mid band voltage gain (c) lower and higher 3db frequencies and (d)gain-bandwidth product.

2. The circuit of a common source FET amplifier is shown in the figure below. Find expressions for voltage gain Av and current gain Ai for the circuit in mid frequency region where Rs is bypassed by Cs. Find also the input resistance Rin for the amplifier. If RD = 3kΩ, RG = 500kΩ, μ = 60, rds = 30kΩ, compute the value of Av, Ai , and Rin.
3. Draw the circuit of h-parameter equivalent of a CE amplifier with un by-passed emitter resistor. Derive an expression for (i) its input impedance and (ii) voltage gain, using the equivalent circuit.
4. What is a 'multistage amplifier'? Give the requirements to be fulfilled for an ideal coupling network.
5. Draw a neat sketch to illustrate the structure of a N-channel E-MOSFET. Explain its operation.
6. What are the important characteristics of a cascade amplifier? Write the circuit of cascade amplifier and determine an expression for its voltage gain in terms of its circuit parameters
7. Write a neat sketch to shown the construction of a depletion-enhancement MOSFET and explain its operation in both the modes.
8. Draw the circuit of a RC-coupled amplifier. Explain its behaviour at low, mid and high frequencies by drawing separate equivalent circuit for each frequency region
9. Explain the classification of power amplifiers according to operational modes.

Module-6 (FEEDBACK IN AMPLIFIERS AND OSCILLATOR:):

1. Sketch the block diagram of a feedback amplifier and derive the expressions for gain (1) with positive feedback and (2) with negative feedback.
2. State the advantages of negative feedback.
3. An amplifier, without feedback, has a voltage gain of 400, lower cut-off frequency f1 = 50 Hz, upper cutoff frequency f2 = 200 KHz and a distortion of 10%. Determine the amplifier voltage gain, lower cut-off frequency and upper cut-off frequency and distortion, when a negative feedback is applied with feedback ratio of 0.01.
4. An amplifier, with feedback, has voltage gain of 100. When the gain without feedback changes by 20% and the gain with feedback should not vary more than 2%. If so, determine the values of open loop Gain A and feedback ratio.
5. Discuss Nyquist criterion for stability of feedback amplifier, with the help of Nyquist plot and Bode plot.
6. Explain the relevant information ,how the negative feedback improves stability reduce noise and increase input impedance?
7. Explain voltage shunt feedback amplifiers &current series feedback amplifiers?
8. Explain the classification of amplifiers? Explain the following in detail.
9. Explain current shunt and voltage shunt feedback amplifiers?
10. With simple diagrams explain the operation of negative resistance oscillator using tunnel diode?
11. Draw the wave form for negative resistance oscillator.
12. With simple diagrams explain the wein bridge oscillator and derive its frequency of oscillation.

13. **Explain RC phase shift oscillator?**
14. **Explain Clapp's oscillator and derive the expression for frequency of oscillation.**
15. **Also explain how frequency stability can be improved Clapp's oscillator?**
16. **Explain Hartley oscillator and derive the equation for oscillation?**
17. **Derive the frequency of oscillation for colpitts oscillator.**
18. **Explain pierce crystal oscillator and derive the equation for oscillation? Derive the frequency of oscillation.**
19. **State Barkhausen criterion for sustained oscillations in a sinusoidal oscillator. The capacitance values of the two capacitors C1 and C2 of the resonant circuit of a colpitt oscillator are C1 = 20pF and C2 = 70pF.The inductor has a value of 22μH. What is the operating frequency of oscillator?**

Module-7,8 (OPERATIONAL AMPLIFIER AND ITS APPLICATIONS):

1. **In the differential amplifier circuit shown below, the transistors have identical characteristics and their β=100.Determine the (i) output voltage (ii) the base currents and (iii) the base voltages taking into account the effect of the RB and VBE. Take VBE=0.7 Volts.**
2. **Suggest modification in the given circuit of Opamp to make it (i) inverting (ii) non inverting.**
3. **A differential amplifier has inputs Vs1=10mV, Vs2 = 9mV. It has a differential mode gain of 60 dB and CMRR is 80 dB. Find the percentage error in the output voltage and error voltage. Derive the formulae used**
4. **Write the circuit of current mirror used in a op-amp design and explain its operation.**
5. **Write the circuit diagram of a square wave generator using an opamp and explain its operation.**
6. **What is an integrator? Derive the formula for its output voltage. Explain its working with neat and clean waveform i) In case of square wave input ii) In case of sine wave input**
7. **Derive the formula for summing amplifier and on averaging amplifier in non Inverting configuration.**

Module-9 (FILTERS AND TUNED AMPLIFIERS):

1. **Explain in detail about single tuned amplifier**
2. **Explain in detail about double tuned amplifier**
3. **Explain in detail about stagger-tuned amplifier**
4. **Compare single tuned and double tuned amplifier**
5. **Explain the different types of neutralization?**
6. **What is the fundamental difference between audio amplifiers and tuned amplifiers? How is bandwidth related to resonant frequency (fr) and the quality factor (Q).**
7. **Draw the circuit diagram of a collector tuned amplifier and derive the expression for the voltage gain at the tuned frequency.**
8. **Explain the reasons for potential instability in tuned amplifiers.**

Module-10(WAVEFORM GENERATION AND SHAPING CIRCUIT):

1. **Explain how the timer IC 555 can be operated as an astable multivibrator, using timing diagrams.**
2. **Explain, using neat circuit diagram and waveforms, the application of timer IC555 as monostable multivibrator**
3. **Explain bistable Multivibrator and its types?**
4. **Explain about speedup capacitors or commutating capacitors.**

5. **Explain about Monostable Multivibrator.**
6. **Explain about collector coupled astable Multivibrator.**
7. **Explain emitter coupled astable Multivibrator.**
8. **Write in detail about Schmitt Trigger circuit?**

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name: Digital Electronic Circuit **Subject Code-EC(EE)302**
Year: 2nd Year **Semester: Third**

Module Number	Topics	Number of Lectures
1	**Number Systems and Codes**	**5L**
	1. Binary, octal and hexadecimal number systems, and conversion of number with one radix to another	2L
	2. Different binary codes	3L
2	**Logic Functions**	**7L**
	3. Boolean algebra and Boolean operators	2L
	4. Logic Functions	1L
	5. Minimization of logic functions using Karnaugh -map	3L
	6. Quine-McClausky method of minimization of logic functions	1L
3	**Combinational Circuits**	**8L**
	7. Introduction to combinational circuits, logic convention, and realization of simple combinational functions using gates	2L
	8. Implications of delay and hazard	1L
	9. Realization of adders and subtractors	2L
	10. Design of code converters, comparators, and decoders	2L
	11. Design of multiplexers, demultiplexers,	1L
4	**Analysis of Sequential Circuits**	**8L**
	12. Introduction to sequential circuits: Moore and Mealy machines	1L
	13. Introduction to flip-flops like SR, JK, D 8iT with truth tables, logic diagrams, and timing relationships	2L
	14. Conversion of Flip-Flops, Excitation table	2L
	15. State tables, and realization of state stables	3L
5	**A/D and D/A Converter**	**3L**
	Different types of converters	3L
6	**Logic Families**	**5L**
	16. Introduction to Logic families	1L
	17. TTL family	1L
	18. CMOS family	2L
	19. Electrical characteristics of logic families	1L

Faculty In-Charge HOD, ECE Dept.

Assignment

Module 1

Perform the following number system conversions:

(a) $(1110010)_2 = (\ ?\ )_{10}$

(b) $(10000111)_2 = (\ ?\ )_8$

(c) $(DACB6)_{16} = (?)_8$

Module 2

1. A system of logic is to be designed which has two outputs & three inputs. One output will be TRUE if an odd number of inputs are TRUE. The other output will be TRUE if only one input alone is TRUE. Draw the truth table & write the corresponding Boolean equation.
2. **(a)**Minimize the following using K-map and realize the simplified expression using basic gates only.

   $$Y = \sum(0,2,3,6,7,8,10,11,12,15)$$

   (b) What is the difference between combinational circuit and sequential circuit?
3. Prove that

   (a) $BCD + A\,\overline{C}\,\overline{D} + ABD = ABD + A\,\overline{C}\,\overline{D} + AB\,\overline{C}$

   (b) $(A+B)(\overline{A}\,\overline{C}+C)(\overline{\overline{B}+AC}) = \overline{A}\,B$
4. Plot the logical expression $ABCD + A\,\overline{B}\,\overline{C}\,\overline{D} + A\,\overline{B}\,C + AB$ on a 4-variable K-map; obtain the simplified expression from the map.
5. Minimize the function using K-map and realize the simplified expression using basic gates. $f(a,b,c,d)\sum m(9,10,12) + d(3,5,6,7,11,13,14,15)$
6. Minimize the function using K-map. $\Pi M\,(3,6,8,11,13,14) \bullet d\,(1,5,7,10)$.
7. Let f (A, B) = $\overline{A} + B$, then find the value of f(f(x+y , y) ,z).
8. Let X*Y = $\overline{x} + y$ and z = X*Y, then find the value of z*x.
9. Write the canonical SOP form of $f = A + \overline{B}C$ and implement using decoder

Module 3

1. Implement a Full Adder circuit using two Half Adder. Write the truth table for Half Subtractor.
2. Implement the following functions using 3:8 Decoders. F (A, B, C) =∑ m (0, 1, 4, 5, 7)
3. What is Multiplexer?
4. Why multiplexer is called as “data selector”?
5. Implement $f(A,B,C) = \sum m(1,2,3,5,6,7)$ using 4:1 MUX with AB as select line.
6. Write thetruth table of 4:1 MUX and implement the circuit using logic gates.
7. Design XOR and OR gate using 2:1 MUX.
8. Write the canonical SOP form of $f = A + \overline{B}C$ and implement using decoder.

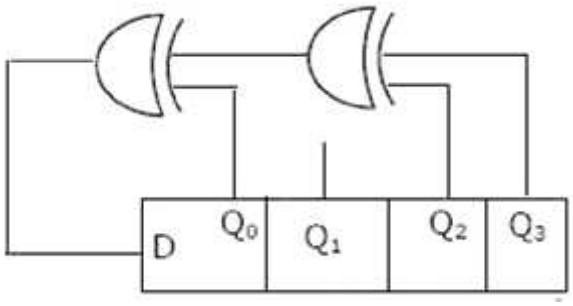
Module 4

1. Explain Ring counter. Draw the circuit diagram and wave form of Ring counter.
2. Preform the conversion from T flip-flop to JK flip-flop.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

3. Write down the difference between Synchronous and Asynchronous counter. What is modulus in a counter (explain with an example)? Design 3 bit asynchronous ripple counter (up counter) and draw the output waveform.
4. What is register?
5. What is shift register?
6. What are the types of shift register? Sketch the block diagrams of each type.
7. A 4 bit shift register, which shifts 1 bit to the right at every clock pulse, is initialized to values 1000 for ($Q_0Q_1Q_2Q_3$). The D input is derived from Q0, Q2 and Q3 through two XOR gates as shown in figure.Write the 4 bit values ($Q_0Q_1Q_2Q_3$) after each clock pulse till the pattern (1000) reappears on ($Q_0Q_1Q_2Q_3$).



8. Using proper truth table and logic diagram find the characteristic equation of SR Flip-Flop. Design a 2-bit up/down synchronous counter using SR flip flop. Use one directional control input M. For M=0, the counter will count up and for M=1, counter will count down.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name: Electric Circuit Theory **Subject Code-EE301**
Year: 2nd Year **Semester: Third**

Module Number	Topics	Number of Lectures
1	**Introduction:**	**4L**
	1. Introduction todifferent types of signals-continuous and discrete, different types of systems .	1
	2. Introduction to linear , non -linear, lumped, distributed, passive, active, lateral, bi-lateral elements , networks	1
	3. Assumptions made circuit theory and network and general explanation of KCL, KVL	1
	4. Explaining current divider, voltage divider rule, method of solving network using KCL, KVL	1
2	**Magnetically coupled circuits**	**3L**
	1. Magnetic coupling, Polarity of coils, Polarity of inducedvoltage, Concept of Self and Mutual inductance, Coefficient of coupling. Modeling of coupled circuits,	1
	2. Concept of self and mutual inductance, co-efficient of coupling	1
	3. Modelling of coupled circuits	1
3.	**Laplace transform:**	**5L**
	1. Significance of Laplace transform. 2. Analysis of Impulse, Step & Sinusoidal response of RL, RC, and RLC circuits with respect to Laplace transform.	2
	3. Transient analysis of different electrical circuits with and without initialconditions.	3
4	**Fourier method and waveform analysis**	**8L**
	1. Significance of Fourier series and Fourier transform	1
	2. Difference of Fourier and Laplace transform	1
	3. Application of Fourier series in different types signals.	3
	4. Application of Fourier transform to solve circuit theory problems	3
5	**Network theorems**	**9L**
	1. Formulation of network equations, Source transformation,Loop variable analysis, Node variable analysis. Assumptions made in solving Network problems	1
	2. Problems with DC & AC sources involving:	

	A. Superposition Theorem	1
	B. Thevenin and Norton theorem.	2
	C. Maximum Power transfer theorem.	1
	D. Millman theorem	1
	E. Tellegen theorem	1
	Additional problem solving involving Network theorems	2
6	**Graph theory:**	**5L**
	1. Concept of Tree, Branch, Tree link	1
	2. Incidence matrix, Tie-set matrix and loop currents	2
	3. Cut set matrix and node pair Potentials, network equilibrium equations	2
7	**Two port network analysis:**	5L
	1. Open circuit Impedance & Short circuit Admittance parameter, Z-parameter, Y-parameter	**1**
	2. Transmission line parameters	1
	3. Hybrid parameters	1
	4. Inter-relations between the parameters, Driving point impedance and admittance	2
8	**Filter Circuits**	**6L**
	1. Analysis and synthesis of filters, general properties and types of filters.	2
	1. Low pass, High pass, Band pass, Band 2. reject,	2
	3. Active filters	2
Total Number Of Hours = 45		

Faculty In-Charge HOD, EE Dept.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

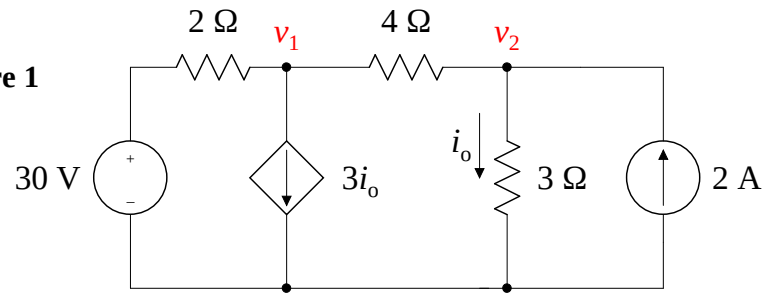
Lecture-wise-Plan

Assignment:1

Module-1

1. In the circuit of Figure 1, find the current i_o using nodal analysis.

Figure 1



2. **Explain invertible system. Why it is important to have an inverse of a system?**
3. **What are the conditions for a system to be a linear system?**

Module-2:

1. **In the following circuit as shown in Fig.2, k=1 find I_1 and I_2**

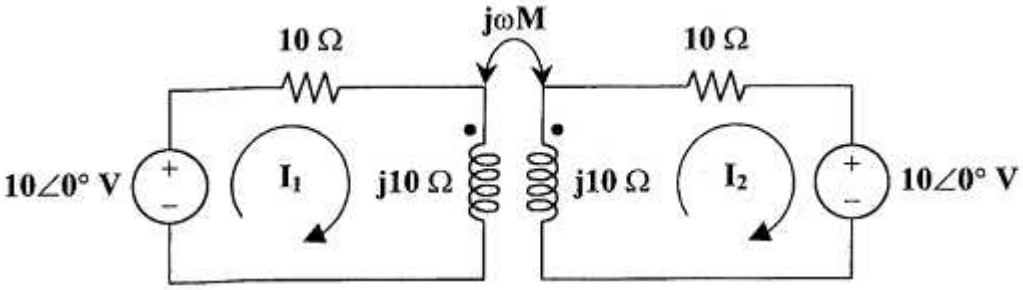


Figure 2

2. **For the three coupled coils (Figure 3) calculate the total inductance**

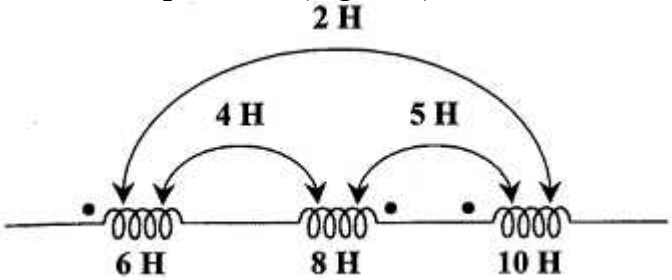
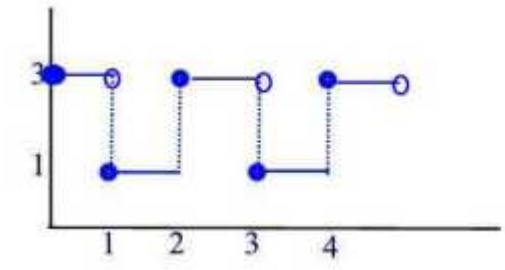


Figure 3

Module-3:

1. Find the Laplace transform of the given function , Fig.4



2. The circuit was initially in steady state with the switch in position 'a'. At t=0+ it goes from 'a' to 'b', find the expression for voltage $V_o(t)$ at t>0.

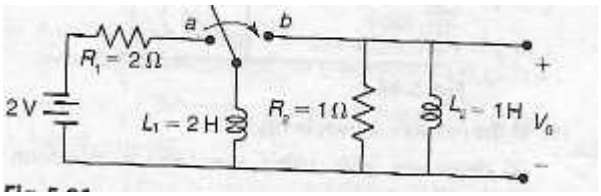


Fig.5

Module-4:

1. Find the Fourier for a train of pulses given by the equation
   V(t) = V; 0<t<T/2
   =0; T/2<t<T

2. When a complex wave is applied to a pure inductor, the current wave has lesser harmonics than the applied voltage . Explain, why?

Module-5:

1. Explain the applications and limitations of Millmans' theorem.
2. Mention the salient features of Tellegens' theorem.

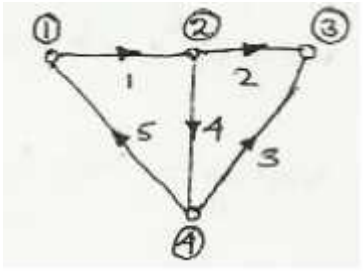
Module-6:

1. Draw the directed graph from the following incidence matrix

Branch

Node	1	2	3	4	5	6	7
1	-1	0	-1	1	0	0	1
2	0	-1	0	-1	0	-1	0
3	1	1	0	0	-1	1	0
4	0	0	1	0	1	0	-1

2. **Determine the tie-set matrix of the following graph. Also find the equation of the branch current and voltages.**



UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Module-7:

1. **Evaluate the condition of reciprocity for a two port network in terms of**
   (a) Z- parameters
   (b) Y-parameters
   (c) ABCD parameters
   (d) Hybrid parameters

Module-8:

1. **Design a band pass filter with cut off frequencies of 160 Hz and 8 kHz. The load for the circuit is 1 MΩ.**

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name:-Field Theory **Subject Code**:-EE302

Year: Second Year **Semester: -**Third

Module No.	Topics	Number of Lectures(H)
1.	**Introduction to field theory:-**	**3H**
	1. Physical significance of divergence and curl and definition of field.	1 H
	2. Introduction to different types of field.	2H
2.	**Electrostatic field:-**	**4H**
	3. Electrostatic field – Electric flux density, Gauss law, Divergence theorem.	2H
	4 Concept of electric scalar potential, derivation of electrostatic energy of a static charge. .	2H
3.	**Magnetostatic Field:-**	**5H**
	5. Magneto static field or field due to moving charge, concept of conductivity, resistance, continuity equation.	2H
	6. Concept of conductors and dielectrics.	1H
	7. Polarization of dielectrics and electric susceptibility.	2H
4.	**Characteristics of electrostatic field:-**	**3H**
	8.Capacitors.	1H
	9. Boundary conditions of electrostatic field.	2H
5.	**Magnetic field during stady current:-**	**6H**
	10.BiotSavart Law, Amperes' circuital law, Magnetic vector potential .	1H
	11.Magnetic materials, Concept of magnetization or magnetic polarization .	2H
	12.Concept of conduction current density and displacement current density .	1H
	13.Magnetic boundary conditions .	2H
	Time varying field	**4H**

6.	14.Magnetic forces, concept of inductance, Lorentz force, Faradays' law . 15. Time varying fields, Helmholtz theorem, Helmholtz equation, EM wave equation .	2H 2H
Module No.	**TOPICS**	**Planed Lectures**
7.	**Uniform plain wave**	**4h**
	16.Wave propagation through space, through dielectrics and through conductors 17.Propagation constant, attenuation constant and phase constant .	2H 2H
8.	**Transmission Lines**	**6H**
	18. Intrinsic impedance for good conductors, dielectrics – 1H 19 .Poynting theorem and Poynting vector -1H 20 . Concept of transmission line -1H 21 .Transmission line equations – 1H	2H 1H 2H 1H
TOTAL HOUR REQUIRED=35		

ASSIGNMENT:

MODULE1:

1. Define divergence, curl and gradient. Explain all these in respect of real fields.
2. Classify different vector fields with proper examples.
3. What is a solenoidal and irrotational vector field? Explain with example.

MODULE2:

1. What is the significance of Gauss's law?
2. Can charge exist inside a hollow metallic sphere? Explain
3. What is electric scalar potential? Explain.

MODULE3:

1. State and prove continuity equation
2. What is conductivity? What is the relation of conductivity and resistivity?
3. What is polarization? Explain displacement current.

MODULE4:

1. Explain different type of boundary conditions in electrostatic field.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lecture-wise-Plan

Subject Name:-Field Theory **Subject Code**:-EE302

Year: Second Year **Semester: -**Third

2. What is polarization? How will you generate right handed circular polarize waves using linearly polarized wave sources? Define axial ratio.

MODULE5:

1. State and explain Amperes' circuital law.

2. What is electric potential and electric potential difference? What is magnetic potential? How is it different from electric potential?
3. Explain various magnetic boundary conditions.

MODULE6:

1. Explain the significance ofHelmholtz theorem.

2. Derive an expression for the wave equation in terms of electric field intensity in a perfect dielectric with no absorption**.**

MODULE 7 & 8:

1. Explain propagation constant, attenuation constant and phase constant.

2. Explain the differences in wave propagation through space, through dielectrics and through conductors.

3. Show that a lossless x/8 length line terminated as open circuit, behaves as a capacitor.
4. Draw an equivalent circuit of a transmission line. What is distortion less line? How to achieve this?
5. State and prove Poynting theorem.

Faculty In-Charge HOD, EE Dept.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR
LAB MANUAL

Title of Course: Analog & Digital Electronic Circuit Lab
Course Code: EC(EE)391
L-T-P scheme: 0-0-3 **Course Credit: 2**

Objectives:
The objective of this course is to introduce the organization of acomputer and its principal components, viz, ALU, Control, Memory andInput/output. The course will also enable the student to understand thedesign components of a digital subsystem that required realizing variouscomponents such as ALU, Control, etc.

Learning Outcomes:
Upon successful completion of the Lab course, a student will be able to:
1. An ability to implement basic gates and their operations.
2. An ability to understand and implement Flip Flops
3. An ability to understand and implement Multiplexers
4. An ability to understand and implement shift registers andcounters
5. An ability to understand and implement Encoders and Decoders
6. An ability to understand and implement Half adder and FulladderMust be able to build a small 8 bit processor that supports reading frommemory (16 bytes), Execute 3 instructions, and add/subtract/stop. Alloperations are to be performed on set of 4 registers. Must implementprogram counter and decoder to fetch the next instruction.

Course Contents:
Exercises that must be done in this course are listed below:
1.Realization of basic gates using Universal logic gates
2.Code conversion circuits- BCD to Excess-3
3.One bit and two bit comparator circuits
4. Construction of simple Decoder and Multiplexer circuits using NAND gate.
5. Construction of simple arithmetic circuits-Adder, Subtractor.
6. Realization of RS-JK and D flip-flops using Universal logic gates.
7. Realization of Ring counter and Johnson's counter.
8.Study of Diode as clipper & clamper.
9.Study of Zener diode as a voltage regulator.
10.Study of ripple and regulation characteristics of full wave rectifier without and with capacitor filter.
11. Study of characteristics curves of B.J.T.
12.Study Inverting and Non –inverting Amplifier

Text Book:

Recommended Systems/Software Requirements:

Experiment No.: 01

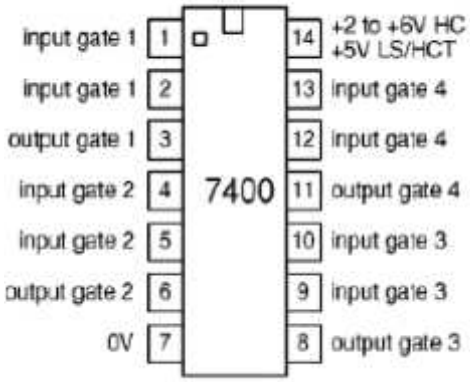
Experiment Name: Realization of basic gates using Universal logic gates

AIM: To construct logic gates NOT, AND, OR, EX-OR, EX-NOR of basic gates using NAND gate and verify their truth tables.

Apparatus Required: Digital trainer kit, patch cords, IC 7400.

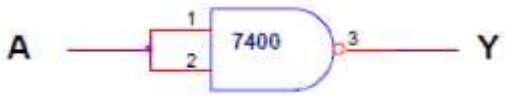
Pin Diagram:

IC 7400 NAND gate



Circuit Diagrams

Not gate

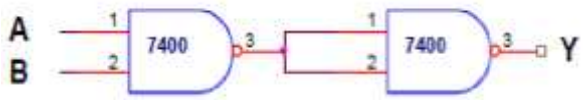


Truth Table

INPUT A	OUTPUT Y
0	1
1	0

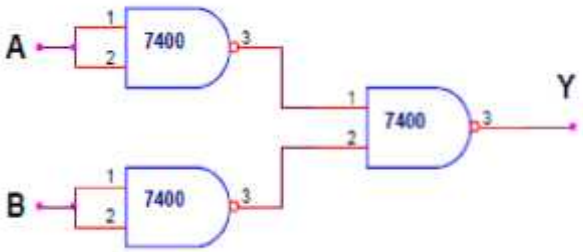
INPUT		OUTPUT Y
A	B	
0	0	0

AND gate



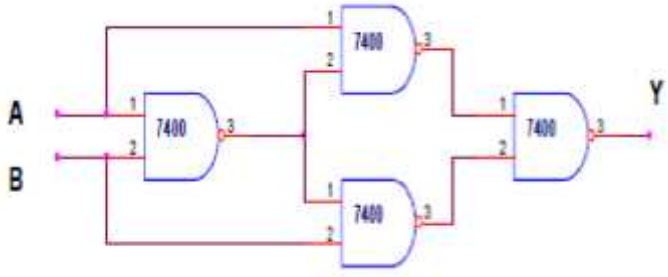
0	**1**	**0**
1	**0**	**0**
1	**1**	**1**

ORgate



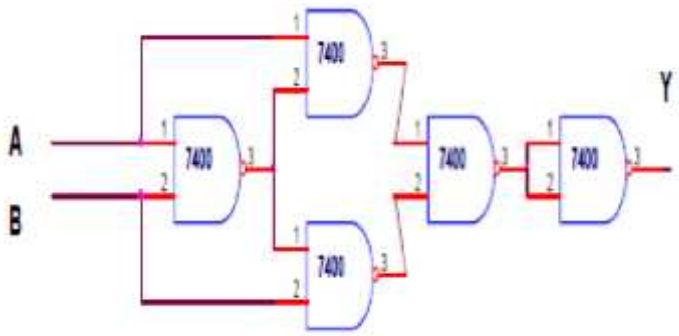
INPUT		**OUTPUT Y**
A	**B**	
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**1**

Ex-OR gate



INPUT		**OUTPUT Y**
A	**B**	
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

Ex-NOR gate



INPUT		OUTPUT Y
A	B	
0	0	1
0	1	0
1	0	0
1	1	1

Procedure:

1. Connect the logic gates as shown in the diagrams using IC 7400 NAND gate.
2. Feed the logic signals 0 or 1 from the logic input switches in different combinations at the inputs A & B.
3. Monitor the output using logic output LED indicators.
4. Repeat steps 1 to 3 for NOT, AND, OR, EX – OR & EX-NOR operations and compare the outputs with the truth tables.

Precautions:
1. All the connections should be made properly.
2. IC should not be reversed.

Result: Different logic gates are constructed using NAND gates and their truth tables are verified.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR
LAB MANUAL

Experiment No.: 02

Experiment Name: Code conversion circuits- BCD to Excess-3.

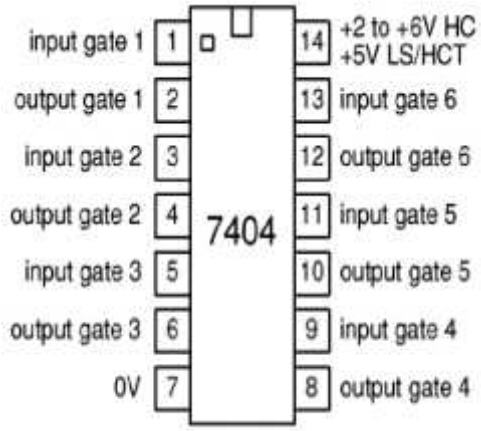
AIM: To design and implement 4 bit BCD to excess-3 converter

Apparatus Required: Digital trainer kit, patch cords, IC 7432, IC 7404, IC 7408, IC 7486.

Pin Diagram:

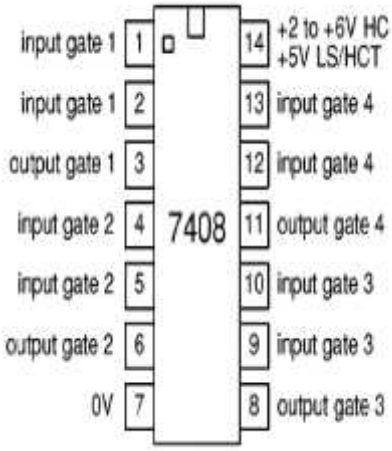
NOT gate

IC 7404



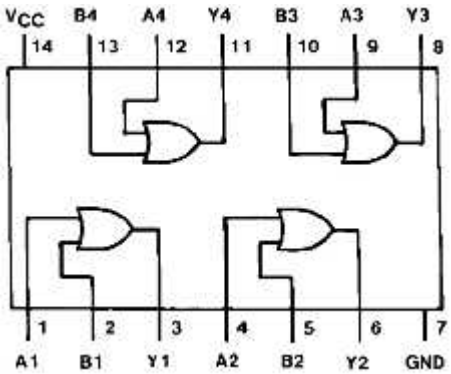
AND gate

IC 7408



OR gate

IC 7432

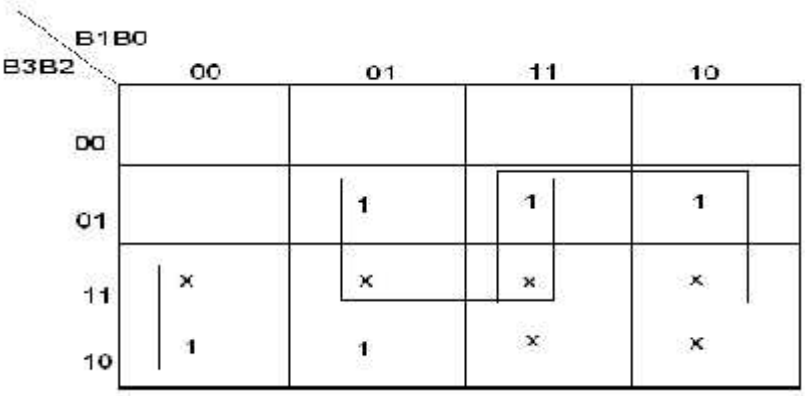


Theory:

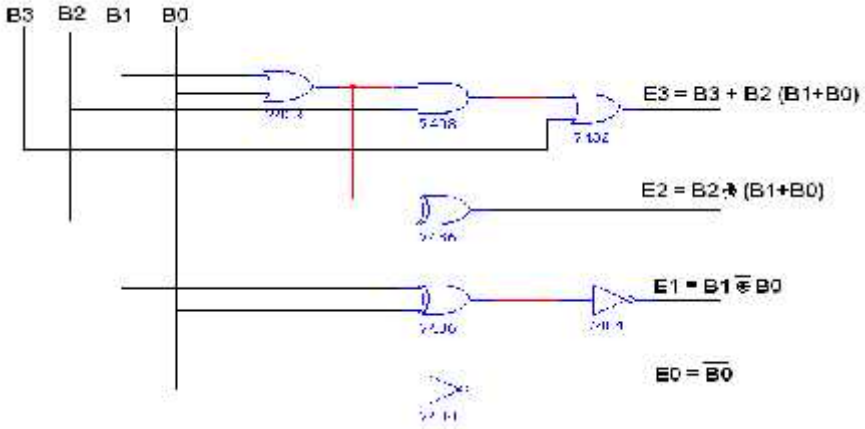
A code converter is a circuit that makes the two systems compatible even though each uses a different binary code. To convert from binary code to Excess-3 code, the input lines must supply the bit combination of elements as specified by code and the output lines generate the corresponding bit combination of code. Each one of the four maps represents one of the four outputs of the circuit as a function of the four input variables.

LOGIC DIAGRAM:

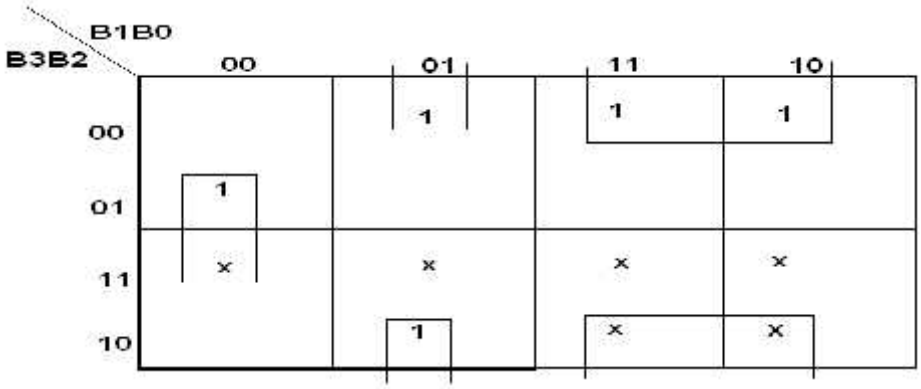
K-map for E3



E3 = B3 + B2 (B0 + B1)

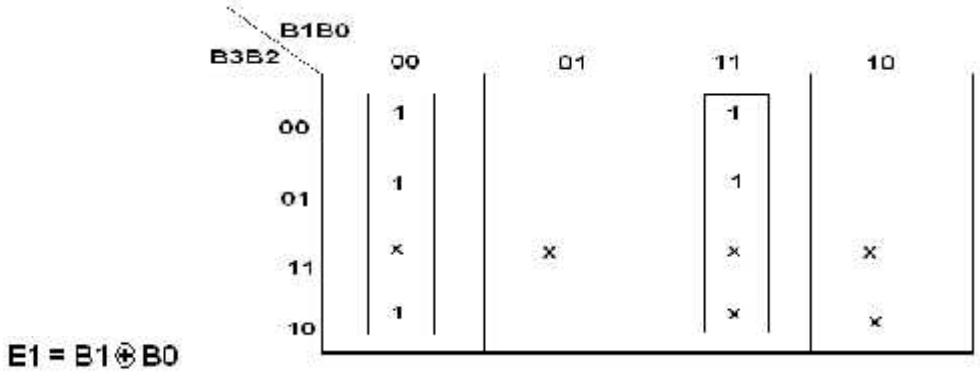


K-map for E2

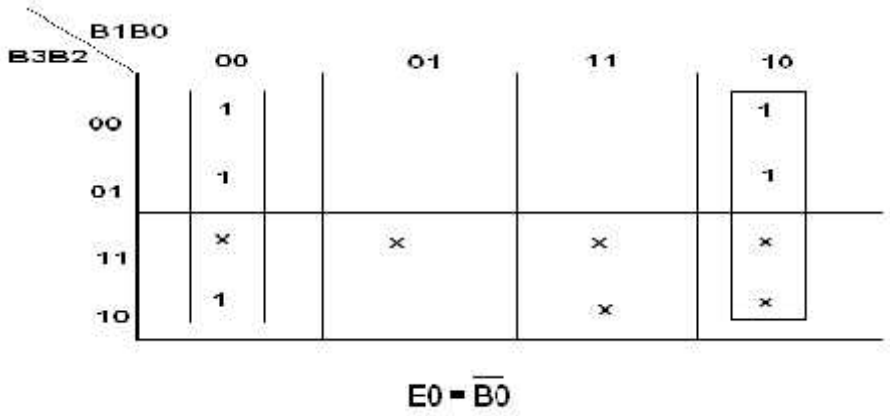


E2 = B2 ⊕ (B1 + B0)

K-map for E1



K-map for E0



TRUTH TABLE:

BCD Input				Excess-3 Output			
B3	**B2**	**B1**	**B0**	**E3**	**E2**	**E1**	**E0**
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0

1	**0**	**0**	**0**	**1**	**0**	**1**	**1**
1	**0**	**0**	**1**	**1**	**1**	**0**	**0**
1	**0**	**1**	**0**	**x**	**x**	**x**	**x**
1	**0**	**1**	**1**	**x**	**x**	**x**	**x**
1	**1**	**0**	**0**	**x**	**x**	**x**	**x**
1	**1**	**0**	**1**	**x**	**x**	**x**	**x**
1	**1**	**1**	**0**	**x**	**x**	**x**	**x**
1	**1**	**1**	**1**	**x**	**x**	**x**	**x**

PROCEDURE:

i. Connections were given as per circuit diagram.
ii. Logical inputs were given as per truth table
iii. Observe the logical output and verify with the truth tables.

Result:

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR
LAB MANUAL

Experiment No.: 03

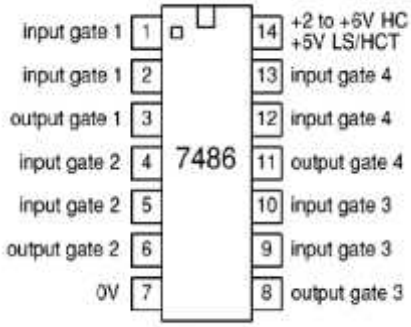
Experiment Name: One bit and two bit comparator circuits.

AIM: To design and implement 1 bit and 2 bit comparators circuits and verify its outputs.

Apparatus Required: Digital trainer kit, patch cords, IC 7404, IC 7408, IC 7486, IC 7432.

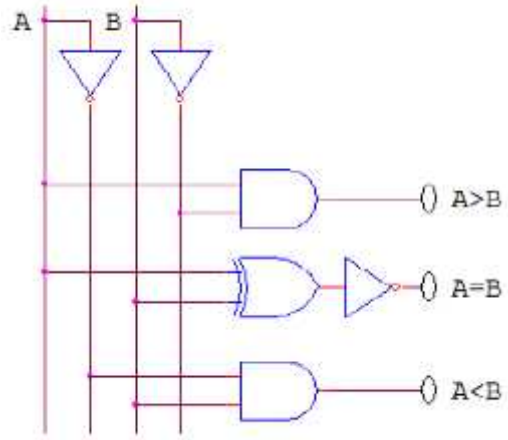
Pin Diagram:

Ex-OR gate



LOGIC DIAGRAM:

1bit Comparator



Truth Table

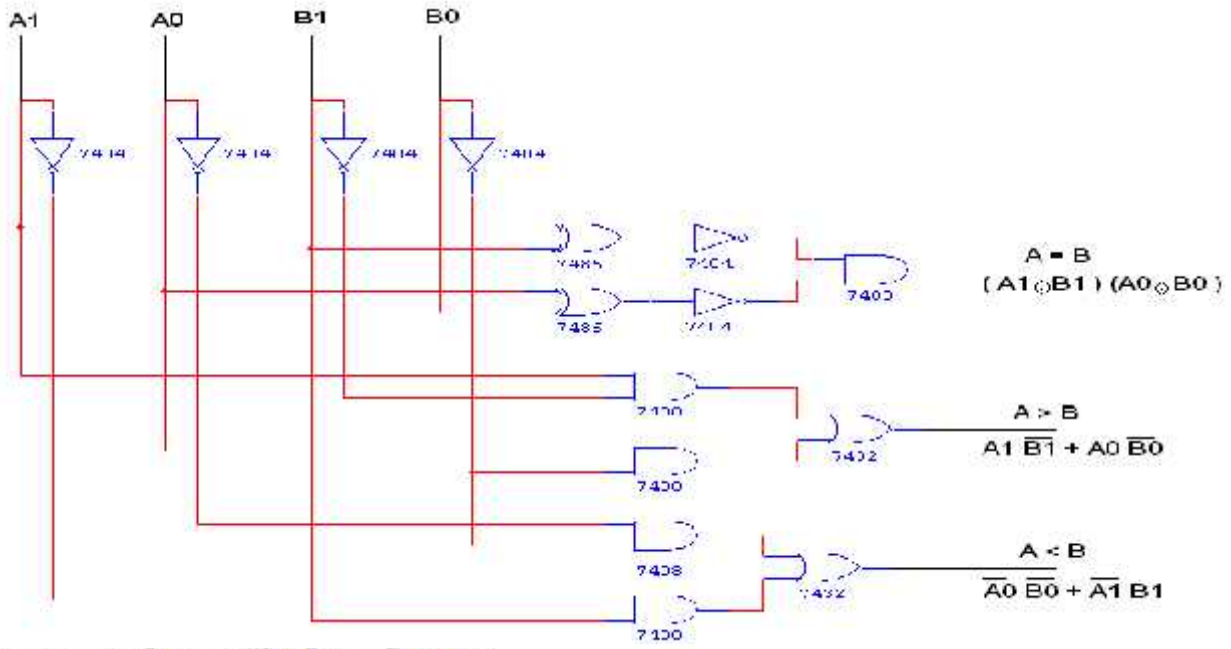
INPUTS		OUTPUTS		
A	B	A > B	A = B	A < B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$A>B = A\bar{B}$

$A<B = \bar{A}B$

$A=B = \bar{A}\,\bar{B} + AB$

2bit Comparator



$(A>B) = A1\,\bar{B}1 + A0\,B1\,B0 + B0A1A0$

$(A=B) = (A0 \oplus B0)\,(A1 \oplus B1)$

$(A<B) = B1\,\bar{A}1 + B0\,\bar{A}1\,\bar{A}0 + \bar{A}0\,B1\,B0$

Truth Table:

INPUTS				OUTPUTS		
A_1	A_0	B_1	B_0	A > B	A = B	A < B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

PROCEDURE:

- Check all the components for their working
- Insert the appropriate IC into the IC base
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT:

Experiment No.: 04

Experiment Name: Construction of simple Decoder and Multiplexer circuits using NAND gate.

AIM: To design and implement simple decoder and multiplexer circuits and verify its outputs.

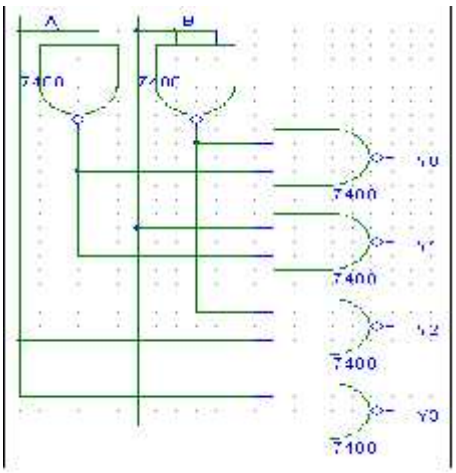
Apparatus Required: Digital trainer kit, patch cords, IC 7400.

Theory:

A decoder is a combinational circuit that connects the binary information from 'n' input lines to a maximum of 2^n unique output lines.
Multiplexers are very useful components in digital systems. They transfer a large number of information units over a smaller number of channels, (usually one channel) under the control of selection signals. Multiplexer means many to one. A multiplexer is a circuit with many inputs but only one output. By using control signals (select lines) we can select any input to the output. Multiplexer is also called as data selector because the output bit depends on the input data bit that is selected. The general multiplexer circuit has 2^n input signals, n control/select signals and 1 output signal.

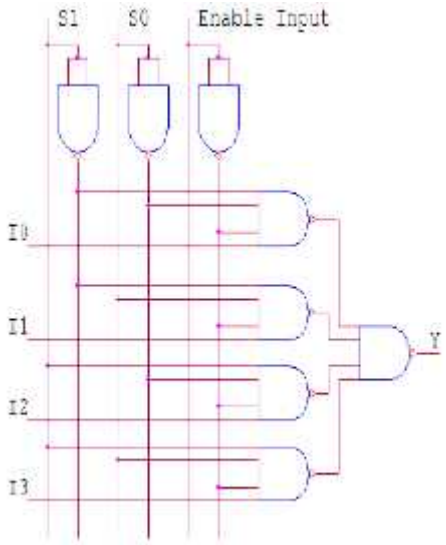
Logic Diagram:

2:4 DECODER: **Truth Table:**



INPUT		OUTPUT			
A	B	Y0	Y1	Y2	Y3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

4:1 Multiplexer **Truth Table**



Select Inputs		Enable Input	Inputs				Out puts
S_1	S_0	E	I_0	I_1	I_2	I_3	Y
X	X	1	X	X	X	X	0
0	0	0	0	X	X	X	0
0	0	0	1	X	X	X	1
0	1	0	X	0	X	X	0
0	1	0	X	1	X	X	1
1	0	0	X	X	0	X	0
1	0	0	X	X	1	X	1
1	1	0	X	X	X	0	0
1	1	0	X	X	X	1	1

Y = D0 S1' S0' + D1 S1' S0 + D2 S1 S0' + D3 S1 S0

PROCEDURE:

- Check all the components for their working
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT:

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

LAB MANUAL

Experiment No.: 05

Experiment Name: Construction of simple arithmetic circuits-Adder, Subtractor.

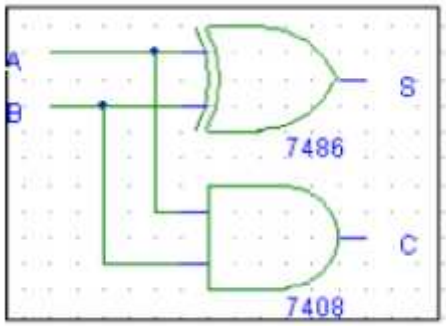
AIM: To design and implement simple adder and subtractor circuits and verify its outputs.

Apparatus Required: Digital trainer kit, patch cords, IC 7400, IC 7408, IC 7486, IC 7432

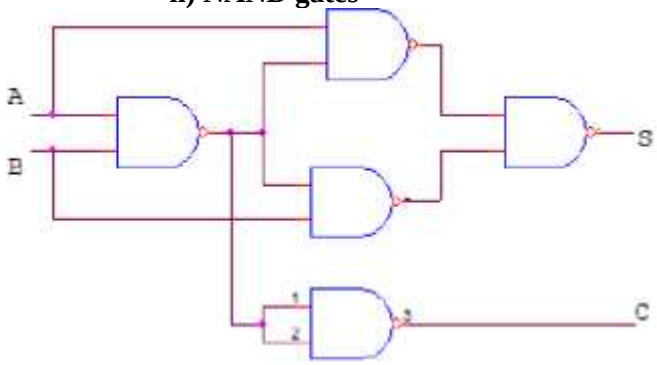
I. To Realize Half Adder

Logic Diagram

i) Basic gates



ii) NAND gates



Truth Table:

INPUTS		OUTPUTS	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Boolean Expression

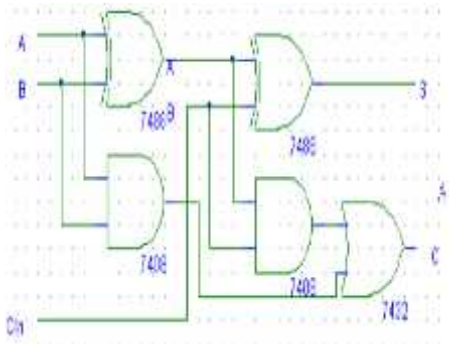
$S = A \oplus B$

$C = A\ B$

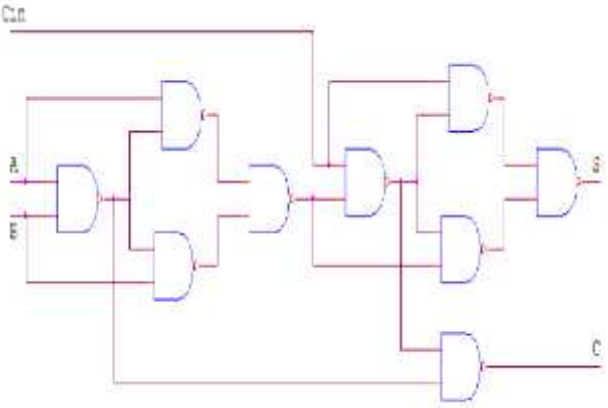
II. To Realize Full Adder

Logic Diagram

i) Basic gates



ii) NAND gates



Truth Table:

TRUTH TABLE

INPUTS			OUTPUTS	
A	B	Cin	S	C
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

Boolean Expression

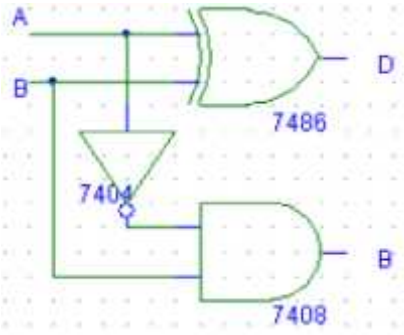
$$S = A \oplus B \oplus C$$

$$C = A\,B + B\,Cin + A\,Cin$$

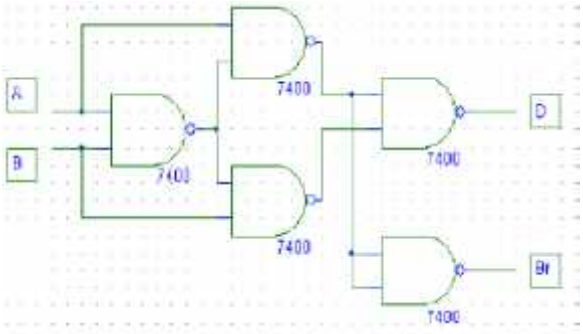
III. Half Subtractor

Logic Diagram

ii) Basic gates



ii) NAND gates



Truth Table:

INPUTS		OUTPUTS	
A	B	D	Br
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Boolean Expression:

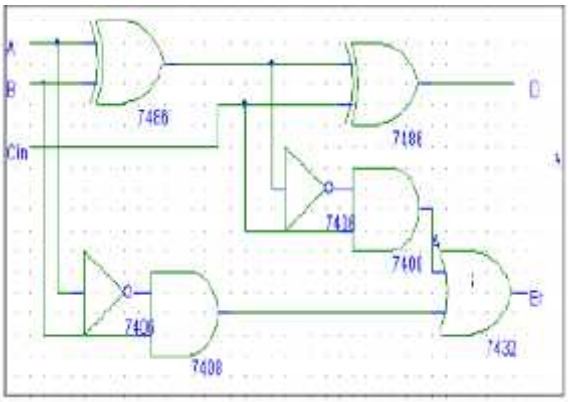
$$D = A \oplus B$$

$$Br = \bar{A}B$$

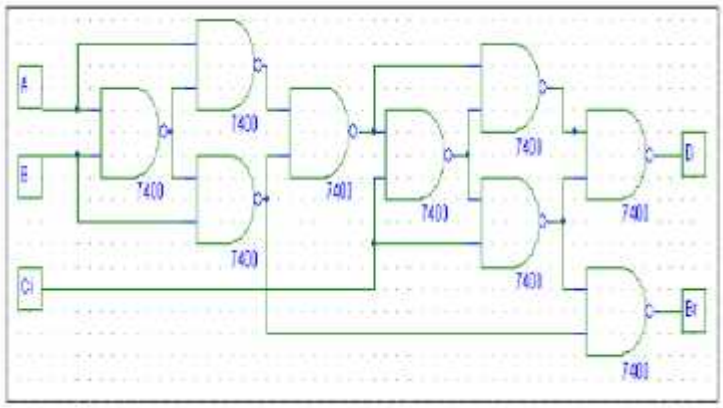
IV. Full Subtractor

Logic Diagram

iii) Basic gates



ii) NAND gates



Truth Table:

INPUTS			OUTPUTS	
A	B	Cin	D	Br
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Boolean Expression:

$$D = A \oplus B \oplus C$$

$$Br = \bar{A}\,B + B\,Cin + \bar{A}\,Cin$$

PROCEDURE:

- Check all the components for their working
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT:

Experiment No.: 06

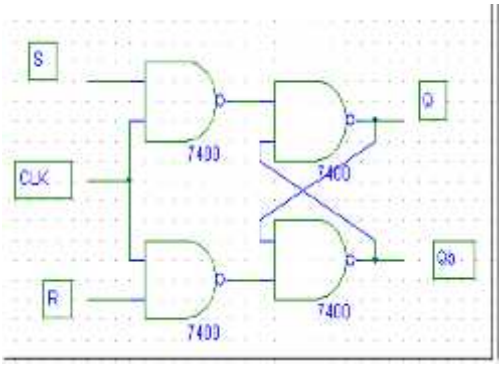
Experiment Name: Realization of RS-JK and D flip-flops using Universal logic gates.

AIM: To design and implement RS-JK and D flip-flops using Universal logic gates and verify its outputs.

Apparatus Required: Digital trainer kit, patch cords, IC 7400, IC 7404, IC 7402, IC 7410, IC 7432

SR Flip Flop

Logic Diagram:



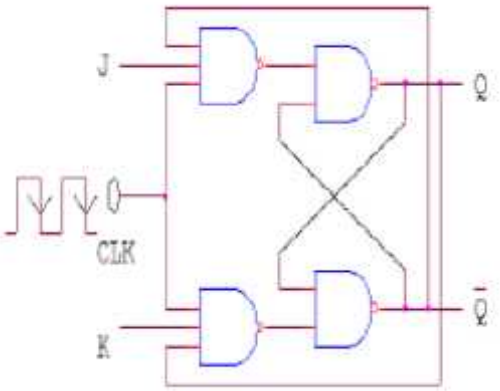
Truth Table

S	R	Q
0	0	No Change
0	1	0
1	0	1
1	1	Forbidden

JK Flip Flop

Logic Diagram

Truth Table



J	K	Q
0	0	No Change
0	1	0
1	0	1
1	1	Race around

D Flip Flop

Logic Diagram **Truth Table**

PROCEDURE:

• Check all the components for their working
• Make connections as shown in the circuit diagram.
• Verify the Truth Table and observe the outputs.

RESULT

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR
LAB MANUAL

Experiment No.: 07

Experiment Name: Realization of Ring counter and Johnson's counter.

AIM: To design and implement Ring counter and Johnson's counter and verify its outputs.

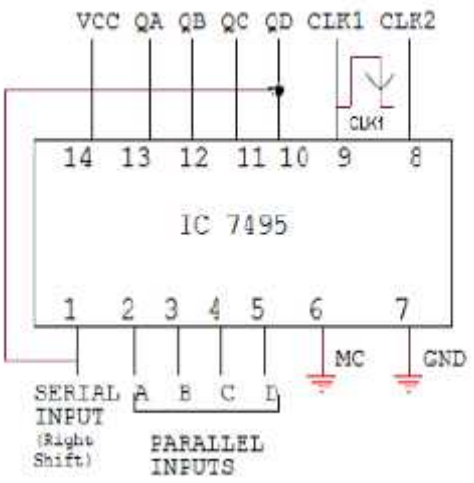
Apparatus Required: Digital trainer kit, patch cords, IC 7404, IC 7495.

Theory:

Ring counter is a basic register with direct feedback such that the contents of the register simply circulate around the register when the clock is running. Here the last output that is QD in a shift register is connected back to the serial input.
A basic ring counter can be slightly modified to produce another type of shift register counter called Johnson counter. Here complement of last output is connected back to the not gate input and not gate output is connected back to serial input. A four bit Johnson counter gives 8 state output.

Ring Counter

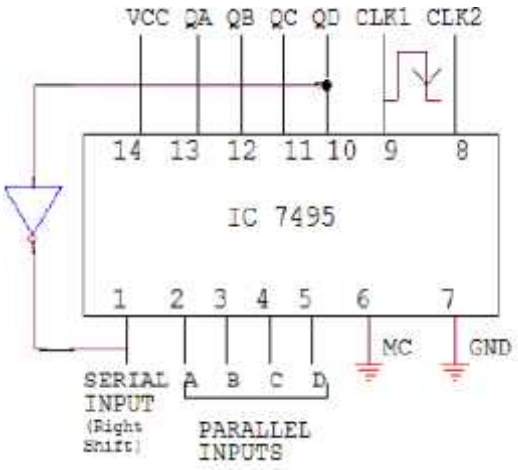
Logic Diagram: **Truth Table**



Clock pulses	Q_A	Q_B	Q_C	Q_D
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	0	0
5	0	1	0	0
6	0	0	1	0
7	0	0	0	1
8	1	0	0	0

Johnson Counter

Logic Diagram: **Truth Table**



Clock pulses	Q_A	Q_B	Q_C	Q_D
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0

PROCEDURE:

- Check all the components for their working
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT

AIM: Study of Diode as clipper & clamper

APPARATUS REQUIRED:

AC Supply (-12V-0V-12V), PN Diodes-1N4007, Capacitor-470μF,Connecting Wires, Resistor-10 KΩ, Breadboard, Multimeter

THEORY:

There are a variety of diode networks called clippers that have the ability to "clip" off a portion of the input signal without distorting the remaining part of the alternating waveform. The half-wave rectifier is an example of the simplest form of diode clipper—one resistor and diode. Depending on the orientation of the diode, the positive or negative region of the input signal is "clipped" off.
There are two general categories of clippers: series and shunt. The series configuration is defined as one where the diode is in series with the load, while the shunt variety has the diode in a branch parallel to the load.

The clamping network is one that will "clamp" a signal to a different dc level. Thenetwork must have a capacitor, a diode, and a resistive element, but it can also employan independent dc supply to introduce an additional shift. The magnitude of Rand C must be chosen such that the time constant,τ= RC is large enough to ensurethat the voltage across the capacitor does not discharge significantly during the intervalthe diode is non-conducting.

CLIPPER'SCIRCUIT DIAGRAM:

Positive Series Clipper

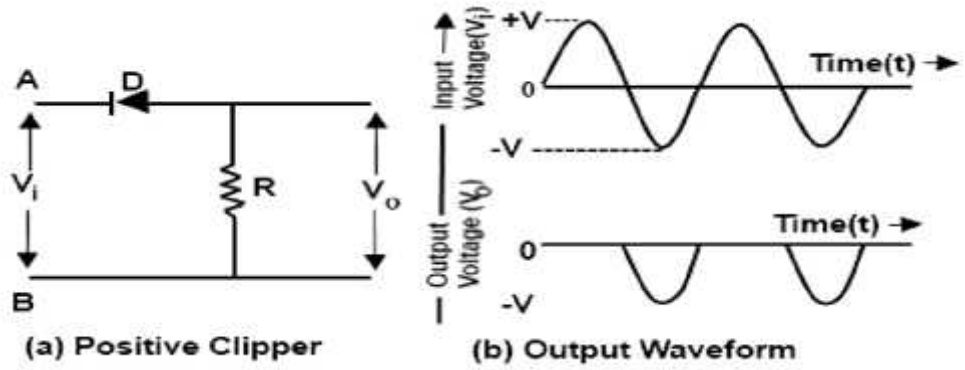


Fig1: Simple positive Series Clipper

Negative Series Clipper

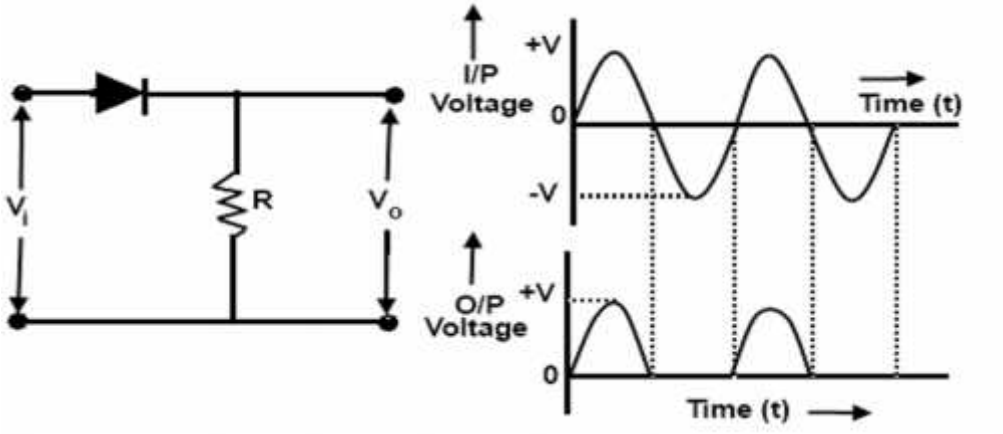


Fig.2: Simple negative series clipper

Biased Positive Series Clipper

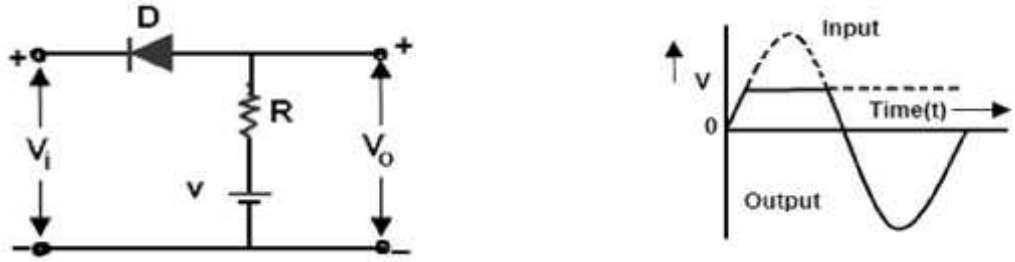


Fig.3: Biased Positive Series Clipper

Biased Negative Series Clipper

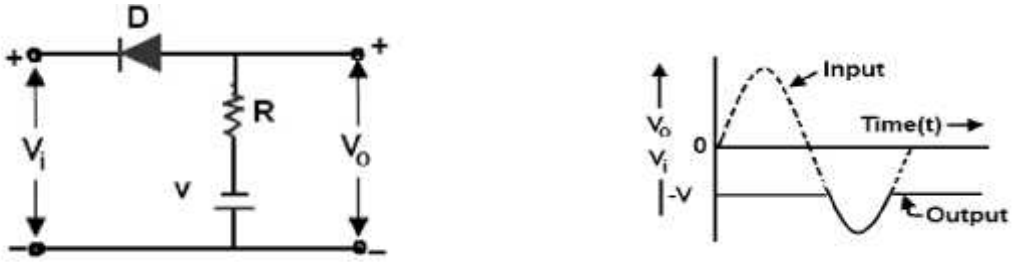


Fig.4: Biased Negative Series Clipper

Positive shunt Clipper

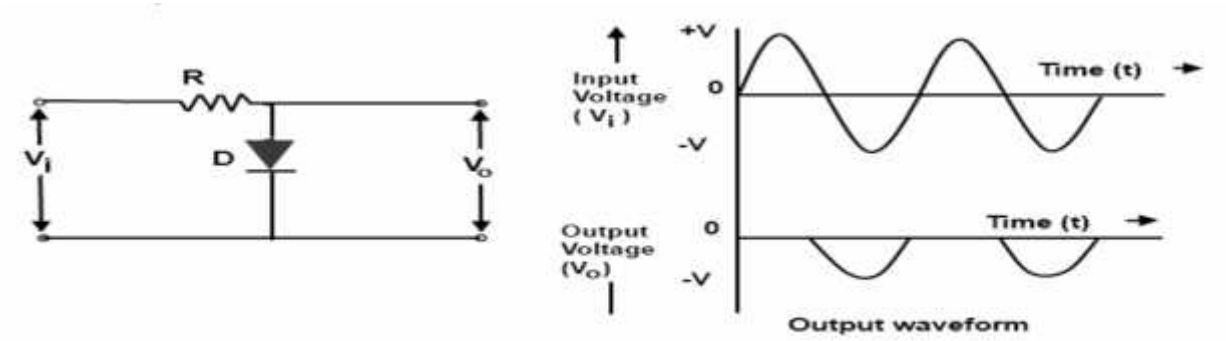


Figure 5: Shunt (parallel) positive clipper

Negative shunt Clipper

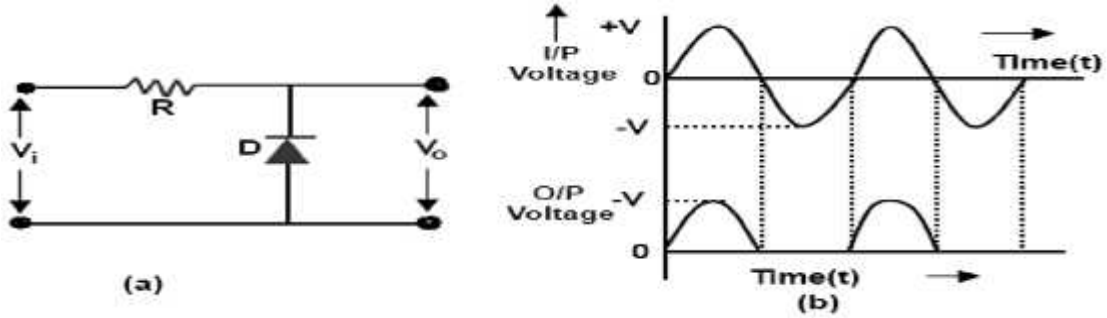


Fig. 6: Shunt (parallel) negative clipper

Biased Positive and negative shunt Clipper

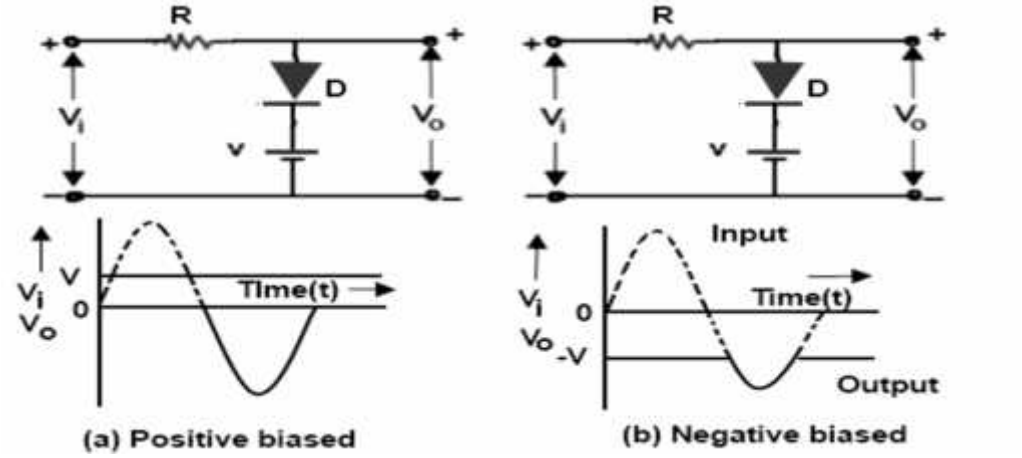
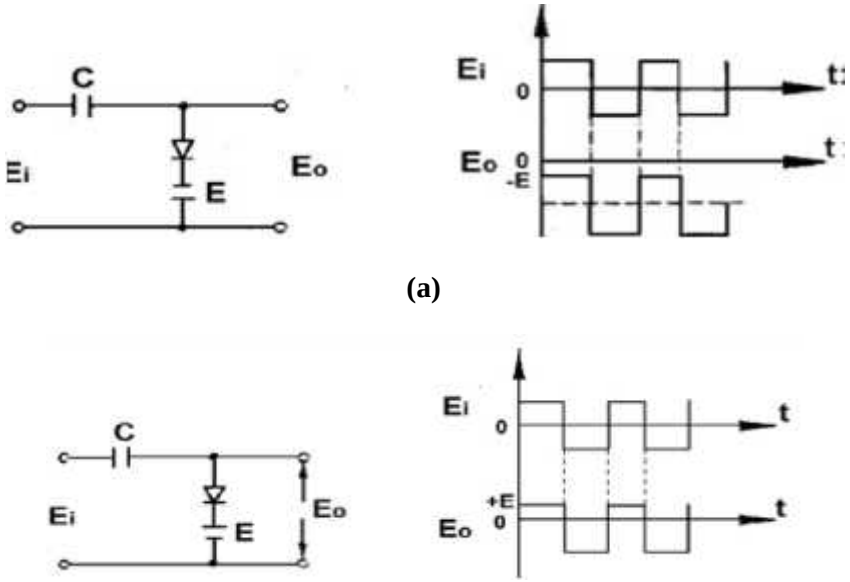


Fig. 7: Biased Positive and negative shunt Clipper

Clamper Circuit Diagrams

Negative Biased Clamping Circuit

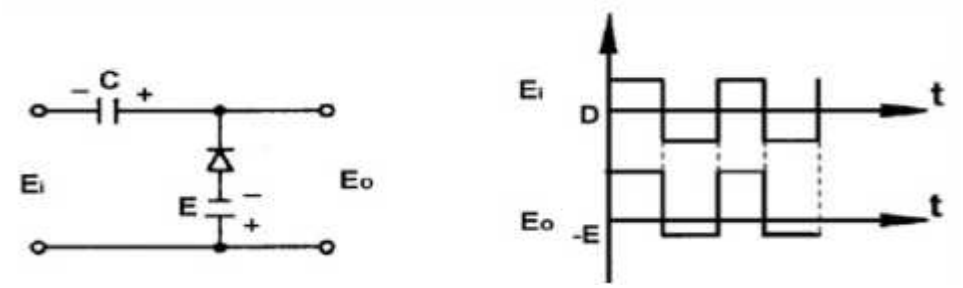


(a)

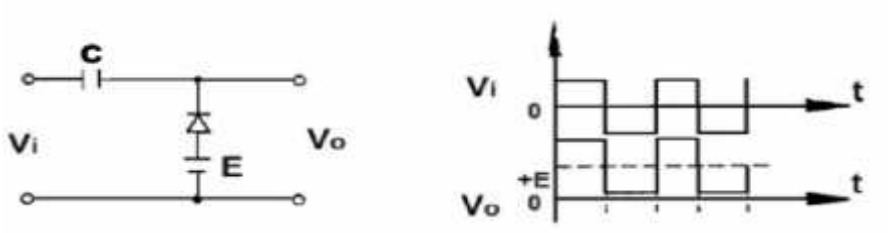
(b)

Fig. 8: Negative Biased Clamping Circuit

Positive Biased Clamping Circuit



(a)



(b)

Fig.9: Positive Biased Clamping Circuit

Experiment No.: 09

AIM: Study of Zener diode as a voltage regulator

APPARATUS:
Zener diode – ECZ5V1, Regulated Power Supply (0-15V), Voltmeter, Ammeter, Resistor(1KΩ), Breadboard, Connecting wires

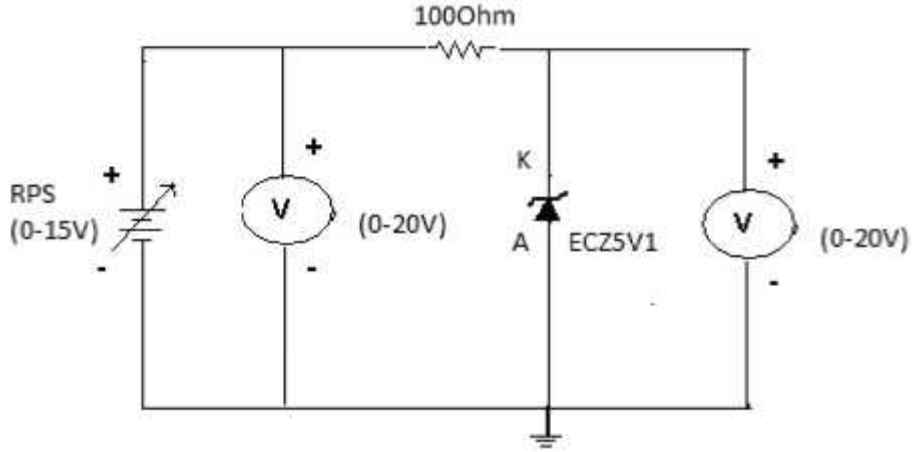
THEORY:
A zener diode is heavily doped p-n junction diode, specially made to operate in thebreak down region. A p-n junction diode normally does not conduct when reverse biased. Butif the reverse bias is increased, at a particular voltage it starts conducting heavily. Thisvoltage is called Break down Voltage. High current through the diode can permanentlydamage the device. To avoid high current, we connect a resistor in series with zenerdiode.Once the diode starts conducting it maintains almost constant voltage across the terminalswhatever may be the current through it, i.e., it has very low dynamic resistance. It is used involtage regulators.

CIRCUIT DIAGRAM:

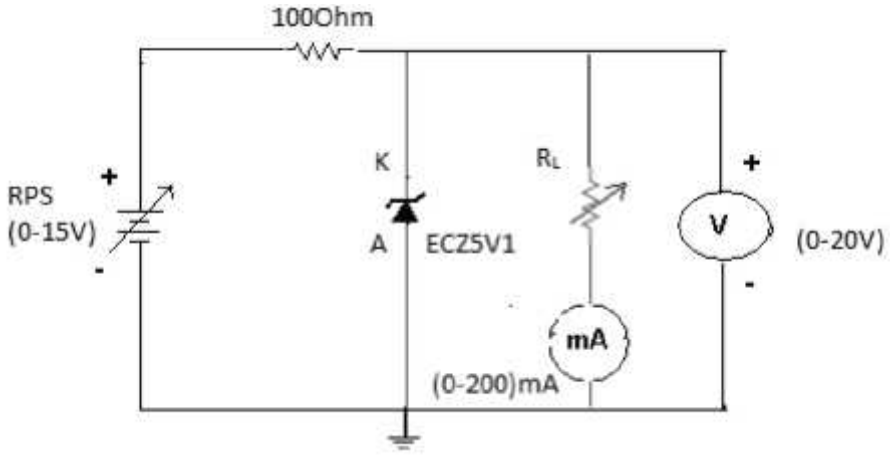
SUPPLY SIDE:

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

LAB MANUAL



LOAD SIDE:



PROCEDURE:

SUPPLY SIDE:
1. Connections are made as per the circuit diagram.
2. The Regulated power supply voltage is increased in steps.
3. For different input voltages (Vi) corresponding output voltages (Vo) are observed and then noted in the tabular form.
4. A graph is plotted between input voltage (Vi) and the output voltage (Vo).

LOAD SIDE:
1. Connection are made as per the circuit diagram
2. The load is placed in full load condition and the output voltage (Vo), load current (IL) are measured.
3. The above step is repeated by decreasing the value of the load in steps.
4. All the readings are tabulated and a graph is plotted between load current (IL) and the output voltage (Vo).

OBSERVATIONS:
SUPPLY SIDE:-

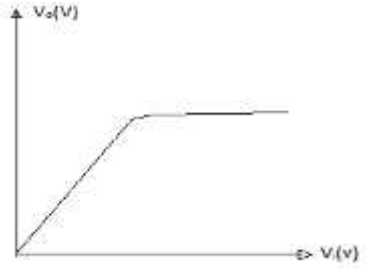
S.NO	Vi (V)	Vo (V)

LOAD SIDE:-

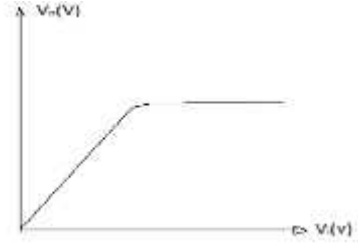
S.NO	IL (V)	Vo(V)

MODEL GRAPH:

SUPPLY SIDE:



LOAD SIDE:



RESULT: Regulator characteristics of zener diode are obtained and graphs are plotted for load and supply side.

Experiment No.: 10

AIM: Study of ripple and regulation characteristics of full wave rectifier without and withcapacitor filter

APPARATUS:
AC Supply (12V-0-12V), PN Diodes 1N4007, Capacitor 470μF, Connecting Wires, Variable resistor (0-10) KΩ, Breadboard, Multimeter

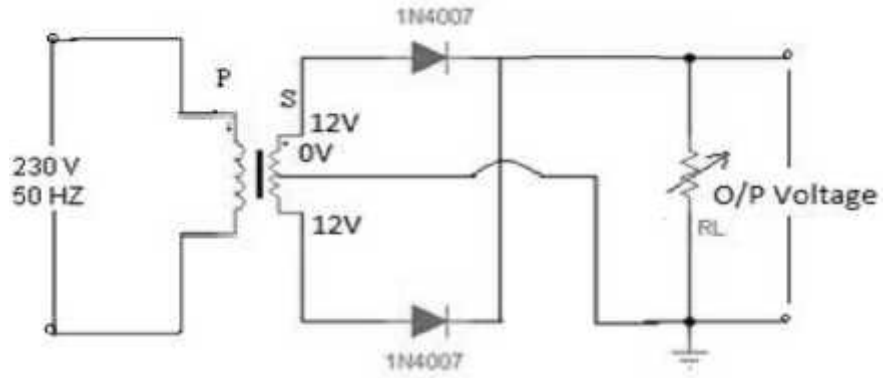
THEORY:

The circuit of a center-tapped full wave rectifier uses two diodes D1&D2. Duringpositive half cycle of secondary voltage (input voltage), the diode D1 is forward biased andD2is reverse biased. The diode D1 conducts and current flows through load resistor RL.During negative half cycle, diode D2 becomes forward biased and D1 reverse biased. Now,D2 conducts and current flows through the load resistor RL

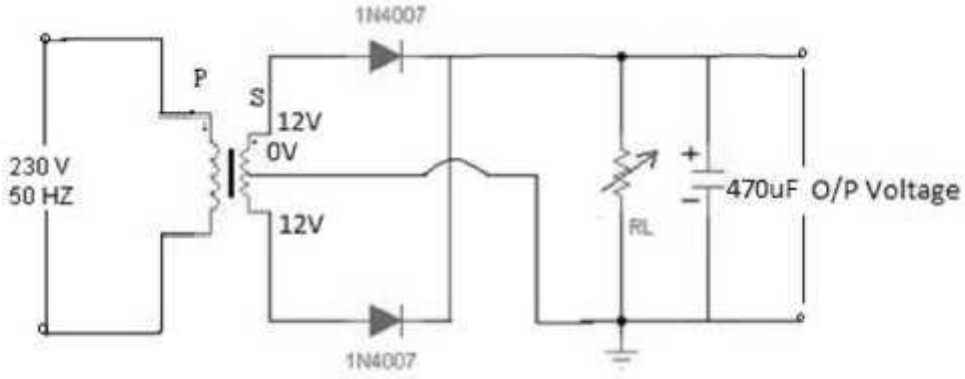
in the same direction. There is acontinuous current flow through the load resistor RL, during both the half cycles and will getunidirectional current as show in the model graph. The difference between full wave and halfwave rectification is that a full wave rectifier allows unidirectional (one way) current to theload during the entire 360 degrees of the input signal and half-wave rectifier allows this onlyduring one half cycle (180 degree).

CIRCUIT DIAGRAM:

WITHOUT FILTER



WITH FILTER



PROCEDURE:

1. Connections are made as per the circuit diagram.
2. Connect the ac mains to the primary side of the transformer and the secondary side to the rectifier.
3. Measure the ac voltage at the input side of the rectifier.
4. Measure both ac and dc voltages at the output side the rectifier.
5. Find the theoretical value of the dc voltage by using the formula $V_{dc}=2V_m/\pi$
6. Connect the filter capacitor across the load resistor and measure the values of Vac and

Vdc at the output.
7. The theoretical values of Ripple factors with and without capacitor are calculated.
8. From the values of Vac and Vdc practical values of Ripple factors are calculated. The practical values are compared with theoretical values.

OBSERVATIONS:

WITHOUT FILTER:

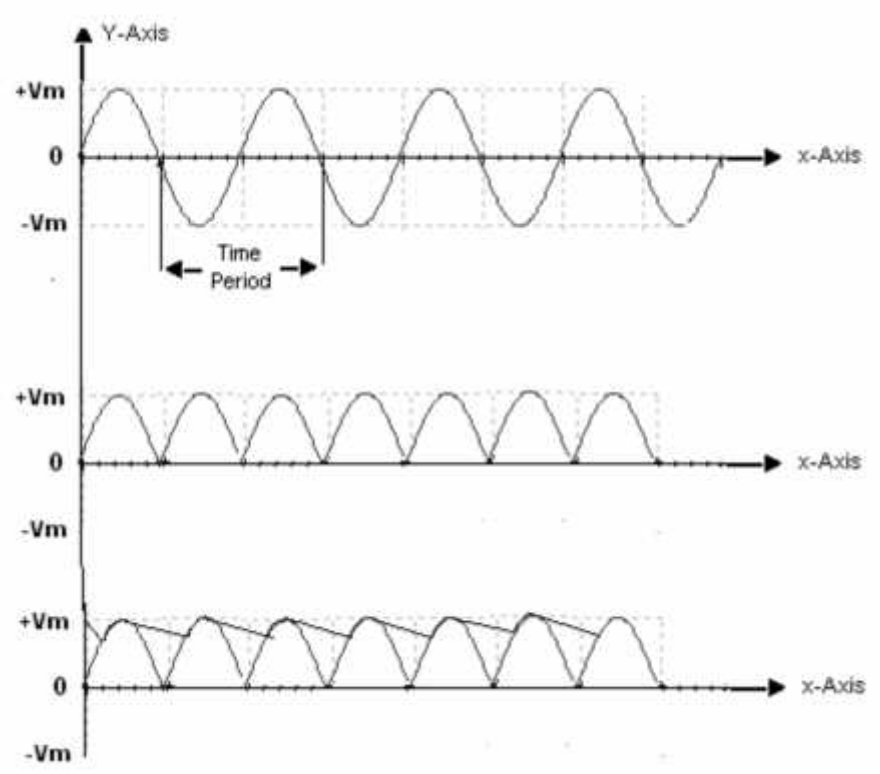
R_L(Ohms)	V_{ac}(Volts)	V_{dc}(Volts)	Ripple Factor = V_{ac}/ V_{dc}	% Regulation (V_{NL}-V_{FL})/V_{FL} *100

WITH FILTER:

RL (Ohms)	Vac (Volts)	Vdc (Volts)	Ripple Factor = Vac/ Vdc	% Regulation (VNL-VFL)/VFL *100

MODEL GRAPHS:

FULLWAVE RECTIFIER (WITH & WITHOUT FILTER):



RESULT:

The ripple factor of the Full-wave rectifier (with filter and without filter) is calculated.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR
LAB MANUAL

Experiment No.: 11

AIM: Study of characteristics curves of B.J.T

APPARATUS:

NPN-Transistor (BC107), Regulated Power Supply (0-15V), Voltmeters (0-20V), Ammeters (0-200μA), (0-200mA), Resistors 1KΩ, Breadboard, Connecting wires, JFET (BFW11)

THEORY:

A transistor is a three terminal device. The terminals are emitter, base, collector. Incommon emitter configuration, input voltage is applied between base and emitter terminalsand output is taken across the collector and emitter terminals. Therefore the emitter terminalis common to both input and output.

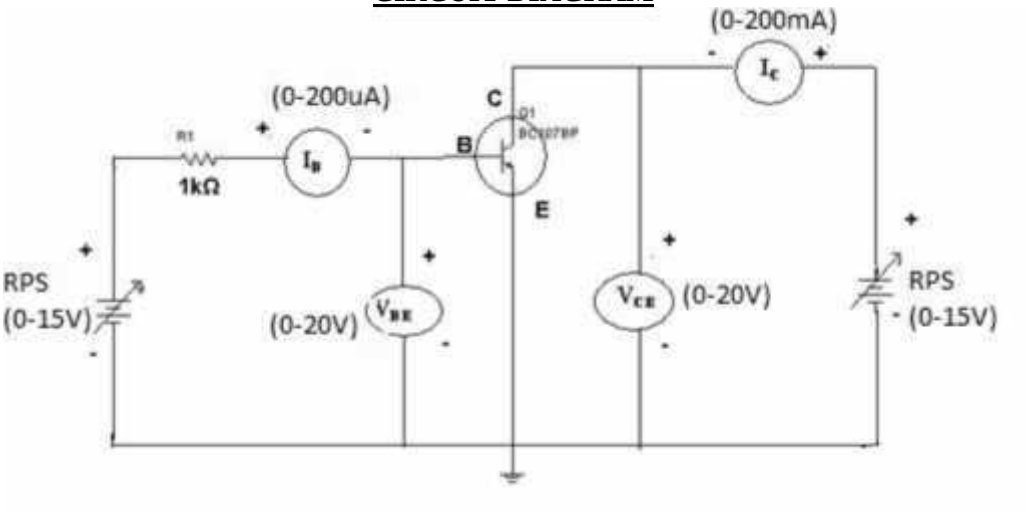
The input characteristics resemble that of a forward biased diode curve. This isexpected since the Base-Emitter junction of the transistor is forward biased. As compared toCB arrangement I_B increases less rapidly with V_{BE}. Therefore input resistance of CE circuitis higher than that of CB circuit.

The output characteristics are drawn between Ic and V_{CE} at constant I_B. the collectorcurrent varies with V_{CE} unto few volts only. After this the collector current becomes almost constant, and independent of V_{CE}. The value of V_{CE} up to which the collector currentchanges with V_{CE} is known as Knee voltage. The transistor always operated in the region above Knee voltage, I_C is always constant and is approximately equal to I_B.

The current amplification factor of CE configuration is given by

$$\beta = \Delta I_C / \Delta I_B$$

CIRCUIT DIAGRAM



PROCEDURE:

INPUT CHARECTERSTICS:

1. Connect the circuit as per the circuit diagram.

2. For plotting the input characteristics the output voltage V_{CE} is kept constant at 1Vand for different values of V_{BE}. Note down the values of I_C
3. Repeat the above step by keeping V_{CE} at 2V and 4V.
4. Tabulate all the readings.
5. Plot the graph between V_{BE} and I_B for constant V_{CE}

OUTPUT CHARACTERSTICS:

1. Connect the circuit as per the circuit diagram.
2. For plotting the output characteristics the input current I_B is kept constant at 10μAand for different values of V_{CE} note down the values of I_C
3. Repeat the above step by keeping I_B at 75 μA, 100 μA
4. Tabulate the all the readings
5. Plot the graph between V_{CE} and I_C for constant I_B.

OBSERVATIONS:

INPUT CHARACTERISTICS:

S.NO	V_{CE} = 1V		V_{CE} = 2V		V_{CE} = 4V	
	V_{BE}(V)	I_B(μA)	V_{BE}(V)	I_B(μA)	V_{BE}(V)	I_B(μA)

OUTPUT CHARACTERISTICS:

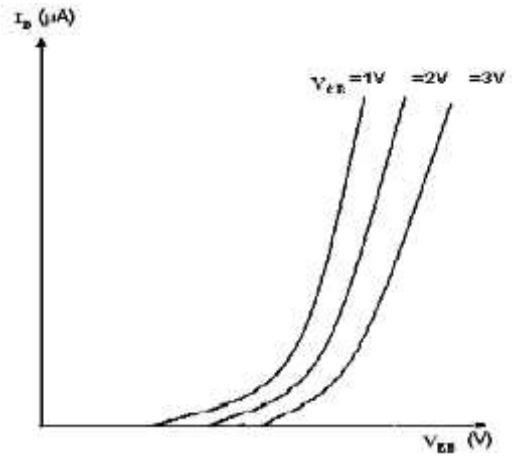
S.NO	IB = 50 μA		IB = 75 μA		IB = 100 μA	
	VCE(V)	IC(mA)	VCE(V)	IC(mA)	VCE(V)	IC(mA)

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

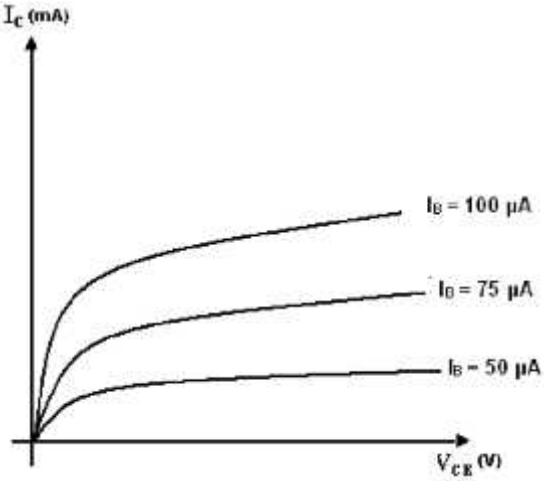
LAB MANUAL

MODEL GRAPHS:

INPUT CHARACTERSTICS:



OUTPUT CHARECTERSTICS:



EXPERIMENT-12

AIM: To study the following linear applications of op-amp

1. Invertingamplifier
2. Non – invertingamplifier

APPARATUS REQUIRED:

1. Op – Amp IC 741
2. Dual Power Supply 15V,
3. Resistors
4. Capacitors
5. Function Generator
6. Cathode Ray Oscilloscope
7. Multimeter
8. Breadboard and Connecting Wires

THEORY:

Inverting Amplifier:

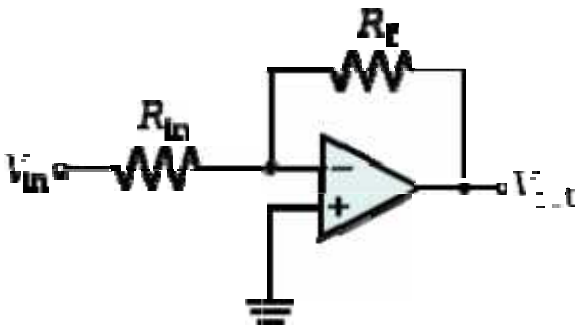
This is the most widely used of all the Op-amp circuits. The output V_0 is fed back to the inverting input through the R_f – R_{in} network as shown in figure where R_f is the feedback resistor. The input signal $V_i$is applied to the inverting input terminal through R_{in} and non-inverting input terminal of Op-amp is grounded.The output V_0 is given by

$$V_0 = V_i(-R_f / R_{in})$$

Where, the gain of amplifier is - R_f / R_{in}

The negative sign indicates a phase-shift of 180 degrees between $V_i$and V_0. The effective input impedance is R_i. An inverting amplifier uses negative feedback to invert and amplify a voltage. The R_{in},R_f resistor network allows some of the output signal to be returned to the input. Since the output is 180° out of phase, this amount is effectively subtracted from the input, thereby reducing the input into the operational amplifier. This reduces the overall gain of the amplifier and is dubbed negative feedback.

CIRCUIT DIAGRAM:



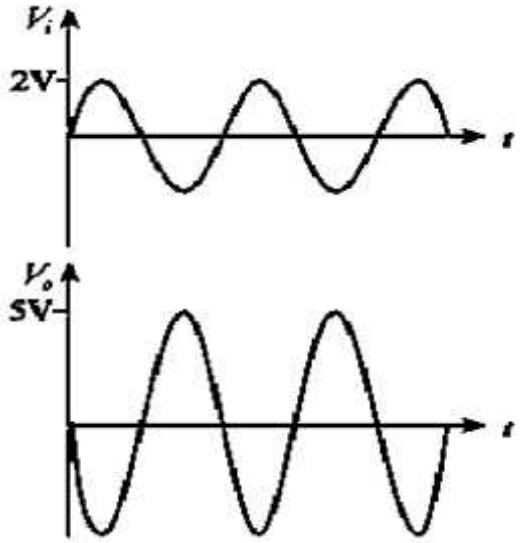
TABULATION:

V_i=----------

R_{in} =----------

S.No	R_f	**Observed** V_0	**Calculated** V_0 $V_0 = V_i(-R_f / R_{in})$

MODEL GRAPH:



Non – inverting amplifier

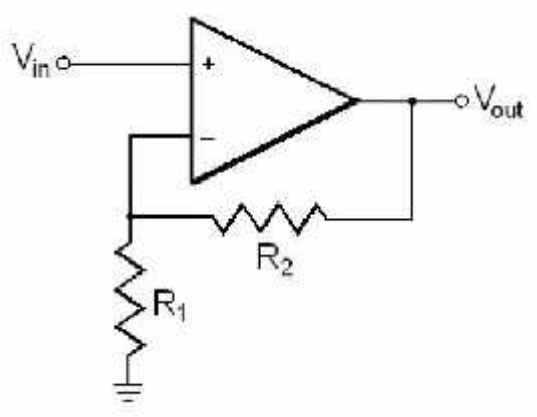
The circuit diagram of non – inverting amplifer is shown in figure. Here, the signal is applied to the non – inverting input terminal and feedback is given to inverting terminal. The circuit amplifiers the input signal without inverting it. The output Voutis given by

$$V_0 = (1 + \frac{R_f}{R_1})V_i$$

CIRCUIT DIAGRAM:

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

LAB MANUAL



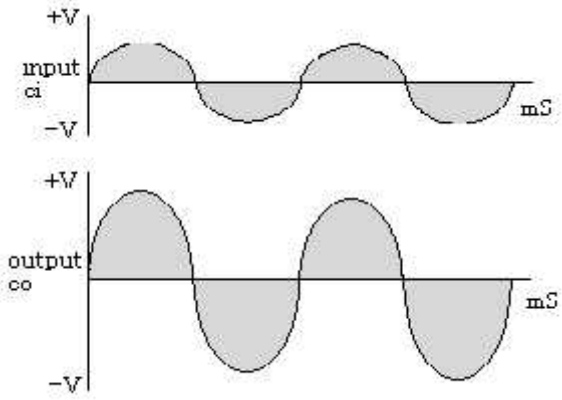
TABULATION:

V_i=----------

R_{in} =----------

S.No	R_f	**Observed** V_0	**Calculated** V_0 $V_0 = V_i(-R_f / R_{in})$

MODEL GRAPH:



PROCEDURE:

Inverting & Non – inverting amplifier

1. Make connections as given in fig 1 & fig 2 for inverting and non inverting amplifiers respectively.
2. Give sinewave input of $V_i$volts using AFO with the frequency of 1KHZ.
3. The output voltage V_0 observed on a CRO. A dual channel CRO to be used to see V_i&V_o.
4. Vary R_f and measure the corresponding V_0 and observe the phase of V_0 with respect to V_0.
5. Tabulate the readings and verify with theoreticalvalues.

RESULT:

The linear applications of 741 op amp were studied experimentally.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Lab Manual

Title of Course: Numerical Methods Lab
Course Code: M(CS)391
L-T-P scheme: 0-0-3 **Course Credit: 2**

Objectives:

1. To give an overview of *what* can be done
2. To give insight into *how* it can be done
3. To give the confidence to tackle numerical solutions

An understanding of how a method works aids in choosing a method. It can also provide an indication of what can and will go wrong, and of the accuracy which may be obtained.To gain insight into the underlying physics.
"The aim of this course is to introduce numerical techniques that can be used on computers, rather than to provide a detailed treatment of accuracy or stability"

Learning Outcomes:
On completion of this course, the student will be able to:

1. Demonstrate skills in using computer programming tools for engineering calculations;
2. Demonstrate ability to construct simple computer algorithms using a programming tool;
3. Apply simple numerical methods to solve mathematical problems with relevance to civil engineering;
4. Appreciate the limitations and the applicability of the numerical methods;
5. Apply computer-based numerical methods for the solution of engineering problems.

Course Contents:
Exercises that must be done in this course are listed below:

1. Assignments on Newton forward /backward, Lagrange' s interpolation.
2. Assignments on numerical integration using Trapezoidal rule, Simpson' s 1/3 rule.
3. Assignments on numerical solution of a system of linear equations using Gauss elimination and Gauss-Seidel iterations.
4. Assignments on numerical solution of Algebraic Equation by Regular-falsi and Newton Raphson methods.
5. Assignments on ordinary differential equation: Euler' s and Runga-Kutta methods.

Text Book:

1. Introductory method of numerical analysis, Sastry S.S
2. Computer Programming in fortran 77, Rajaraman V
3. Numerical methods: for scientific and engineering computation,Mahinder Kumar Jain

.
Recommended Systems/Software Requirements:

1. Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
2. Turbo C or TC3 complier in Windows XP or Linux Operating System.

Experiment No: 1(a) Newton forward interpolation
Aim: Write a C program to implement the Newton forward interpolation.
Description:
Interpolation is the process of finding the values of y corresponding to the any value of x between x0 and xn for the given values of y=f(x) for a set of values of x. Out of the many techniques of interpolation, Newton's Forward and Backward Interpolation are two very widely used formulas. In this tutorial, we're going to discuss a C program for Newton Forward Interpolation along with its sample output.
Both of Newton's formulas are based on finite difference calculus. These formulas are very often used in engineering and related science fields. Before going through the source code for Newton Forward Interpolation, let's go through the forward interpolation formula and the variables used in the C program.
Newton's forward interpolation formula contains y0 and the forward differences of y0. This formula is used for interpolating the values of y near the beginning of a set of tabulated values and

$$P(x) = y_0 + q\Delta y_0 + \frac{q(q-1)}{2!}\Delta^2 y_{-1} + \frac{(q+1)q(q-1)}{3!}\Delta^3 y_{-1} +$$
$$+ \frac{(q+1)q(q-1)(q-2)}{1!}\Delta^4 y_{-2} + \frac{(q+2)(q+1)q(q-1)(q-2)}{5!}\Delta^5 y_{-2} + \ldots$$
$$\ldots + \frac{(q-n-1)\ldots(q-n+1)}{(2n-1)!}\Delta^{2n-1} y_{-(n-1)} + \frac{(q+n-1)\ldots(q-n)}{(2n)!}\Delta^{2n} y_{-n},$$

Compared to forward interpolation, the backward interpolation formula contains yn and the backward differences of yn. This formula is used for interpolating the values of y near the end of a set of tabulated values and also for extrapolating the values of y a little ahead (i.e. to the right) of yn.

Algorithm:

1. Function NFI ()
2. Read n, x
3. For I = 1 to n by 1 do
4. Read x[i], y[i]
5. End for
6. If ((x < x[i] or (x > x[n]))
7. Print “Value lies out of boundary”
8. Exit
9. End if
10. //Calculating p
11. p = (x – x [1]) / (x [2]-x [1])
12. // Forward diff table
13. For j = 1 to (n-1) by 1 do
14. For i =1 to (n - j) by 1 do
15. If (j=1) Then
16. d[i][j] = y [i+1] – y[i]
17. Else
18. d[i][j] = d[i+1][j-1] – d[i][j-1]
19. End if
20. End For
21. End For
22. // Applying Formula
23. Sum =y [1]
24. For I = 1 to (n-1) by 1 do
25. Prod = 1
26. For j =0 to (i-1) by 1 do
27. Prod = prod * (p-j)
28. End for
29. m = fact(i)
30. Sum = sum + (d[1][i] * prod) / m
31. End For
32. Print “Ans is”, Sum
33. End Function

/* Program to implement Newton’s forward interpolation*/

```
1       #include<stdio.h>
2       #include<conio.h>
3       #include<math.h>
4       #include<stdlib.h>
5       main()
6       {
7         float x[20],y[20],f,s,h,d,p;
8         int j,i,n;
9         printf("enter the value of n :");
10        scanf("%d",&n);
11        printf("enter the elements of x:");
```

```
13      {
14          scanf("%f",&x[i]);
15           }
16                  printf("enter the elements of y:");
17               for(i=1;i<=n;i++)
18               {
19              scanf("%f",&y[i]);
20                          }
21       h=x[2]-x[1];
22       printf("Enter the value of f:");
23       scanf("%f",&f);
24     s=(f-x[1])/h;
25     p=1;
26     d=y[1];
27     for(i=1;i<=(n-1);i++)
28      {
29                  for(j=1;j<=(n-i);j++)
30                   {
31                        y[j]=y[j+1]-y[j];
32
33                   }
34                   p=p*(s-i+1)/i;
35                   d=d+p*y[1];
36      }
37     printf("For the value of x=%6.5f THe value is %6.5f",f,d);
38      getch();
39     }
```

OUTPUT:

```
how many record you will be enter: 5
enter the value of x0: 2.5
enter the value of f(x0): 9.75
enter the value of x1: 3
enter the value of f(x1): 12.45
enter the value of x2: 3.5
enter the value of f(x2): 15.70
enter the value of x3: 4
enter the value of f(x3): 19.52
enter the value of x4: 4.5
enter the value of f(x4): 23.75
Enter X for finding f(x): 4.25
u = -0.500
 f(4.25) = 21.583750
```

Experiment No: 1(b) Newton backward interpolation
Aim: Write a C program to implement Newton backward interpolation.
Algorithm:

1. Function NBI ()
2. Read n, x
3. For I = 1 to n by 1 do
4. Read x[i], y[i]
5. End for
6. If ((x < x[i] or (x > x[n]))
7. Print “Value lies out of boundary”
8. Exit

10. //Calculating p
11. p = (x – x [1]) / (x [2]-x [1])
12. // Forward diff table
13. For j = 1 to (n-1) by 1 do
14. For i =1 to (n - j) by 1 do
15. If (j=1) Then
16. d[i][j] = y [i+1] – y[i]
17. Else
18. d[i][j] = d[i+1][j-1] – d[i][j-1]
19. End if
20. End For
21. End For
22. // Applying Formula
23. Sum =y [n]
24. For I = 1 to (n-1) by 1 do
25. Prod = 1
26. For j =0 to (i-1) by 1 do
27. Prod = prod * (p+j)
28. End for
29. m = fact(i)
30. Sum = sum + (d[n-1][i] * prod) / m
31. End For
32. Print “Ans is”, Sum
33. End Function

/* Program to implement Newton’s forward interpolation */

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#include<stdlib.h>
main()
{
    float x[20],y[20],f,s,d,h,p;
    int j,i,k,n;
    printf(“enter the value of the elements :”);
    scanf(“%d”,&n);
    printf(“enter the value of x: \n\n”);
    for(i=1;i<=n;i++)
    {
                scanf(“%f”,&x[i]);
                }
        printf(“enter the value of y: \n\n”);
    for(i=1;i<=n;i++)
    {
                scanf(“%f”,&y[i]);
                }
                h=x[2]-x[1];
       printf(“enter the searching point f:”);
scanf(“%f”,&f);
s=(f-x[n])/h;
d=y[n];
p=1;
for(i=n,k=1;i>=1,k<n;i–,k++)
{
            for(j=n;j>=1;j–)
            {
                        y[j]=y[j]-y[j-1];
```

```
                    p=p*(s+k-1)/k;
                    d=d+p*y[n];
}
printf("for f=%f ,ans is=%f",f,d);
getch();
}
```

OUT PUT:

```
how many record you will be enter: 5
enter the value of x0: 2.5
enter the value of f(x0): 9.75
enter the value of x1: 3
enter the value of f(x1): 12.45
enter the value of x2: 3.5
enter the value of f(x2): 15.70
enter the value of x3: 4
enter the value of f(x3): 19.52
enter the value of x4: 4.5
enter the value of f(x4): 23.75
Enter X for finding f(x): 4.25
```

x(i)	y(i)	y1(i)	y2(i)	y3(i)	y4(i)
2.500	9.750				
3.000	12.450	2.700			
3.500	15.700	3.250	0.550		
4.000	19.520	3.820	0.570	0.020	
4.500	23.750	4.230	0.410	-0.160	-0.180

```
u = -0.500
f(4.25) = 21.583750 -
```

Experiment No: 1(c)Lagrange' s interpolation
Aim: Write a C program to implement Lagrange' s interpolation.

Algorithm:

1. Input number of Observation n
2. For i = 1 to n
3. Input Xi
4. Input Yi
5. Next i
6. Input xp at which yp to be computed
7. Initialize yp = 0
8. For i = 1 to n
9. t = 1
10. For j = 1 to n
11. If j ≠ i
12. t = t * (xp - Xj)/(Xi - Xj)
13. End If
14. Next j

16. Next i
17. Print yp as output
18. Stop

/* Program to implement Lagrange' s interpolation*/

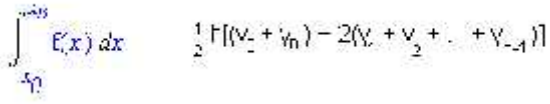
```
#include<stdio.h>
#include<conio.h>
#include<math.h>
int main()
{
  float x[10],y[10],temp=1,f[10],sum,p;
  int i,n,j,k=0,c;

  printf("\nhow many record you will be enter: ");
  scanf("%d",&n);
  for(i=0; i<n; i++)
  {
   printf("\n\nenter the value of x%d: ",i);
   scanf("%f",&x[i]);
   printf("\n\nenter the value of f(x%d): ",i);
   scanf("%f",&y[i]);
  }
  printf("\n\nEnter X for finding f(x): ");
  scanf("%f",&p);

  for(i=0;i<n;i++)
  {
    temp = 1;
    k = i;
    for(j=0;j<n;j++)
    {
      if(k==j)
      {
        continue;
      }
      else
      {
        temp = temp * ((p-x[j])/(x[k]-x[j]));
      }
    }
    f[i]=y[i]*temp;
  }

  for(i=0;i<n;i++)
  {
     sum = sum + f[i];
  }
  printf("\n\n f(%.1f) = %f ",p,sum);
  getch();
}
```

OUTPUT:

```
enter the value of n 4
enter the value to be found 2.5
enter the values for xi's & fi's
1 1
2 8
```

4 64
X = 2.500000
sum = 15.625000

Experiment No:2.Trapezoidal rule
Aim: Write a C program to implement Trapezoidal rule.
Description:
A number of definite integrals need to be solved in applied mathematics, physics and engineering. The manual analytical solution of definite integrals is quite cumbersome and time consuming. So, in this post I have presented source code in C program for Trapezoidal method as one of the computer-programming-based solutions of definite integrals. The techniques of numerical methods are used to solve this equation; it involves a number of calculations and efforts have been made to minimize error in the program.
The trapezium or trapezoidal rule can be used as a way of estimating the area under a curve because the area under a curve is given by integration. So, the trapezoidal rule gives a method of estimating integrals. This is useful when you come across integrals that you don't know how to evaluate. So, the program for trapezoidal method in C given here is applicable to calculate finite integral or area under a curve.
$h=(x_n - x_0)/n$
After that, the C source code for trapezoidal method uses the following formula to calculate the value of definite integral:



Algorithm:

1. Read x1, x2, e {x1 and x2 are the two end points of the internal the allowed error in integral is e}
2. h = x2 - x1
3. SI = (f(x1) + f(x2))/2;
4. I = h - si
5. i = 1 Repeat
6. x = x1 + h/2
7. for J= 1 to I do
8. SI = SI + f(x)
9. x = x + h
10. End for
11. i = 21
12. h = h/2 {Note that the internal has been halved above and the number of points where the function has to be computed is doubled}
13. i0 = i1
14. i1 = h.si. until / I1 - i0 / <= c./i1/
15. Write I1, h, i
16. Stop.

/* Program to to implement Trapezoidal rule */

```
#include<stdio.h>
#include<math.h>
main()
{
 float h, a, b, n, x[20], y[20], sum = 0, integral;
 int i;
 clrscr();
 printf("enter the value of a, b, n:");
 scanf("%f %f %f", &a, &b, &n);
 printf("enter the values of x:");
```

```
 {
  scanf("%f", &x[i]);
 }
 printf("\n enter the values of y:");
 for(i = 0; i <= (n-1); i++)
 {
  scanf("%f", &y[i]);
 }
 h = (b-a)/n;
 x[0] = a;
 for(i = 1; i <= n-1; i++)
 {
  x[i] = x[i-1] + h;
  sum = sum + 2 * y[i];
 }
 sum = sum + y[b];
 integral = sum * (h/2);
 printf("approximate integral value is: %f", integral);
 getch();
}
```

OUTPUT :

enter the values of a, b, n
123
enter the values of x:
123
enter the values of y:
123
approximate integral value is 2.166667

Experiment No:2(a)Simpson' s 1/3 rule

AIM: Write a C Program to implement Simpson' s 1/3 rule.

Description:

n the source code below, a function *f(x) = 1/(1+x)* has been defined. The calculation using **Simpson 1/3 rule in C** is based on the fact that the small portion between any two points is a parabola. The program follows the following steps for calculation of the integral.

- As the program gets executed, first of all it asks for the value of lower boundary value of x i.e. x_0, upper boundary value of x i.e. x_n and width of the strip, h.
- Then the program finds the value of number of strip as n=($x_n - x_0$)/h and checks whether it is even or odd. If the value of 'n' is odd, the program refines the value of 'h' so that the value of 'n' comes to be even.
- After that, this C program calculates value of f(x) i.e 'y' at different intermediate values of 'x' and displays values of all intermediate values of 'y'.
- After the calculation of values of 'c', the program uses the following formula to calculate the value of integral in loop.
  Integral = $*((y_0 + y_n) + 4(y_1 + y_3 + \ldots\ldots\ldots + y_{n-1}) + 2(y_2 + y_4 + \ldots\ldots\ldots + y_{n-2}))$
- Finally, it prints the values of integral which is stored as 'ans' in the program.

If f(x) represents the length, the value of integral will be area, and if f(x) is area, the output of Simpson 1/3 rule C program will be volume. Hence, numerical integration can be carried out using the program below; it is very easy to use, simple to understand, and gives reliable and accurate results.

f(x) = 1/(1+x)

.

Algorithm:

1. Read x1, x2, e
2. h = (x2 - x1)/2
3. i = 2

5. s2 = 0
6. s4 = f(x1 + h)
7. I0 = 0
8. In = (s + 4s4).(h/3)
9. Repeat
10. s2 = s2 + s4 { s2 stores already computed functional value and s4 the value computed in the new nitration }
11. s4 = 0
12. x = x1 + h/2
13. for j = 1 to I do
14. s4 = s4 + f(x)
15. x = x + h
16. h = h/2
17. i = 2i
18. io = in
19. in = (s1 + 2s2 + 4s4) . (h/3)
20. until |In-Io|≤e. /in
21. Write In, h, i
22. STOP

/* Program to to implement Simpson' s 1/3 rule. */

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
main()
{
 float h, a, b, n, x[20], y[20], sum = 0, itgl;
 int i;
 clrscr();
 printf("enter the values of a, b, n");
 scanf("%f%f%f", &a, &b, &n);
 printf("enter the values of x");
 for(i = 0; i <= n; i++)
 {
  scanf("%f", &x[i]);
 }
 printf("\n enter the values of y");
 for(i = 0; i <= n; i++)
 {
  scanf("%f", &y[i]);
 }
 h = (b - a)/n;
 a = x[0];
 b = x[n];
 for(i = 0; i <= (n-2); i++)
 {
  x[i] = x[i] + h;
  if(i % 2 == 0)
  {
   sum = sum + 4 * y[i];
  }
  else
  {
   sum = sum + 2 * y[i];
  }
 }
```

```
 printf("integral value%f", itgl);
 getch();
}
```

OUTPUT :

```
enter the values of a, b, n
123
enter the value of x
4567
enter the values of y
8912
integral value is 5.555556
```

Experiment No: 3(a)Gauss elimination.

AIM: Write a C Program to implement Gauss elimination method.

Description:

let us first consider the following three equations:

a1x + b1y + c1z = d1
a2x + b2y + c2z = d2
a3x + b3y + c3z = d3

Assuming a1 ≠ 0, x is eliminated from the second equation by subtracting (a2/ a1) times the first equation from the second equation. In the same way, the C code presented here eliminates x from third equation by subtracting (a3/a1) times the first equation from the third equation.

Then we get the new equations as:

a1x + b1y + c1z = d1
b'2y + c'2z = d'2
c''3z = d''3

The elimination procedure is continued until only one unknown remains in the last equation. After its value is determined, the procedure is stopped. Now, Gauss Elimination in C uses back substitution to get the values of x, y and z as:

z= d''3 / c''3
y=(d'2 – c'2z) / b'2
x=(d1- c1z- b1y)/ a1.

Algorithm:

1. Start
2. Declare the variables and read the order of the matrix n.
3. Take the coefficients of the linear equation as:
   Do for k=1 to n
   Do for j=1 to n+1
   Read a[k][j]
   End for j
   End for k
4. Do for k=1 to n-1
   Do for i=k+1 to n

a[i][j] = a[i][j] – a[i][k] /a[k][k] * a[k][j]

End for j

End for i

End for k

5. Compute x[n] = a[n][n+1]/a[n][n]
6. Do for k=n-1 to 1

   sum = 0

   Do for j=k+1 to n

   sum = sum + a[k][j] * x[j]

   End for j

   x[k] = 1/a[k][k] * (a[k][n+1] – sum)

   End for k
7. Display the result x[k]
8. Stop

/* Program to to implement Gauss elimination method */

```
#include<stdio.h>
int main()
{
   int i,j,k,n;
   float A[20][20],c,x[10],sum=0.0;
   printf("\nEnter the order of matrix: ");
   scanf("%d",&n);
   printf("\nEnter the elements of augmented matrix row-wise:\n\n");
   for(i=1; i<=n; i++)
   {
      for(j=1; j<=(n+1); j++)
      {
         printf("A[%d][%d] : ", i,j);
         scanf("%f",&A[i][j]);
      }
   }
   for(j=1; j<=n; j++) /* loop for the generation of upper triangular matrix*/
   {
      for(i=1; i<=n; i++)
      {
         if(i>j)
         {
            c=A[i][j]/A[j][j];
            for(k=1; k<=n+1; k++)
            {
               A[i][k]=A[i][k]-c*A[j][k];
            }
         }
      }
   }
   x[n]=A[n][n+1]/A[n][n];
   /* this loop is for backward substitution*/
   for(i=n-1; i>=1; i--)
   {
```

```
    for(j=i+1; j<=n; j++)
    {
       sum=sum+A[i][j]*x[j];
    }
    x[i]=(A[i][n+1]-sum)/A[i][i];
  }
  printf("\nThe solution is: \n");
  for(i=1; i<=n; i++)
  {
    printf("\nx%d=%f\t",i,x[i]); /* x1, x2, x3 are the required solutions*/
  }
  return(0);
```

OUTPUT :

```
No of Equtions : 3
Enter Coefficients of Eqution
4 3 -2
1 1 1
3 -2 1
Enter Constant value
5 3 2
Eliminated matrix as :-
4.00 3.00 -2.00 5.00
0.00 0.25 1.50 1.75
0.00 0.00 28.00 28.00
Solution :
X3 = 1.00
X2 = 1.00
X1 = 1.00
```

Experiment No:3(b) Gauss-Seidel iterations.
AIM: Write a C Program to implement Gauss-Seidel iterations method.

Description:
The program for Gauss-Seidel method in C works by following the steps listed below:

When the program is executed, first of all it asks for the value of elements of the augmented matrix row wise.
Then, the program asks for allowed error and maximum number of iteration to which the calculations are to be done. The number of iterations required depends upon the degree of accuracy.
The program assumes initial or approximate solution as y=0 and z=0 and new value of x which is used to calculate new values of y and z using the following expressions:
x= 1/a1 (d1-b1y-c1z)
y=1/b2 (d2-a2x-c2z)
z=1/c3 (d3-a3x-b3y)

Algorithm:

1. Start
2. Declare the variables and read the order of the matrix n
3. Read the stopping criteria er
4. Read the coefficients aim as

   Do for i=1 to n

   Do for j=1 to n

Repeat for j

Repeat for i

5. Read the coefficients b[i] for i=1 to n
6. Initialize x0[i] = 0 for i=1 to n
7. Set key=0
8. For i=1 to n

   Set sum = b[i]

   For j=1 to n

   If (j not equal to i)

   Set sum = sum – a[i][j] * x0[j]

   Repeat j

   x[i] = sum/a[i][i]

   If absolute value of ((x[i] – x0[i]) / x[i]) > er, then

   Set key = 1

   Set x0[i] = x[i]

   Repeat i
9. If key = 1, then

   Goto step 6

   Otherwise print results

.

/* Program to implement Gauss-Seidel iterations method. */

```
#include<stdio.h>
#include<math.h>
#define X 2
main()
{
   float x[X][X+1],a[X], ae, max,t,s,e;
   int i,j,r,mxit;
   for(i=0;i<X;i++) a[i]=0;
   puts(" Eneter the elemrnts of augmented matrix rowwise\n");
   for(i=0;i<X;i++)
   {
   for(j=0;j<X+1;j++)
   {
   scanf("%f",&x[i][j]);
   }
   }
   printf(" Eneter the allowed error and maximum number of iteration: ");
   scanf("%f%d",&ae,&mxit);
   printf("Iteration\tx[1]\tx[2]\n");
   for(r=1;r<=mxit;r++)
   {
      max=0;
      for(i=0;i<X;i++)
      {
         s=0;
```

```
            if(j!=i) s+=x[i][j]*a[j];
            t=(x[i][X]-s)/x[i][i];
            e=fabs(a[i]-t);
            a[i]=t;
        }
        printf(" %5d\t",r);
        for(i=0;i<X;i++)
        printf(" %9.4f\t",a[i]);
        printf("\n");
        if(max<ae)
        {
            printf(" Converses in %3d iteration\n", r);
            for(i=0;i<X;i++)
            printf("a[%3d]=%7.4f\n", i+1,a[i]);
            return 0;
        }

        }
    }
```

OUTPUT :

```
Enter the number of equations: 3
Enter the co-efficients of the equations:
a[1][1]= 2
a[1][2]= 1
a[1][3]= 1
a[1][4]= 5
a[2][1]= 3
a[2][2]= 5
a[2][3]= 2
a[2][4]= 15
a[3][1]= 2
a[3][2]= 1
a[3][3]= 4
a[3][4]= 8

x[1] =2.500000
x[2] =1.500000
x[3] =0.375000

x[1] =1.562500
x[2] =1.912500
x[3] =0.740625

x[1] =1.173437
x[2] =1.999688
x[3] =0.913359

x[1] =1.043477
x[2] =2.008570
x[3] =0.976119

x[1] =1.007655
x[2] =2.004959
x[3] =0.994933

x[1] =1.000054
```

x[3] =0.999474

converges to solution
x[1]=1.000054
x[2]=2.001995
x[3]=0.999474

Experiment No: 4(a)Regular-falsi

AIM: Write a program to implement Regular-falsi method.

Description:
The C Program for regula falsi method requires two initial guesses of opposite nature. Like the secant method, interpolation is done to find the new values for successive iterations, but in this method one interval always remains constant.

The programming effort for Regula Falsi or False Position Method in C language is simple and easy. The convergence is of first order and it is guaranteed. In manual approach, the method of false position may be slow, but it is found superior to the bisection method.

- tr – a counter which keeps track of the no. of iterations performed
- maxmitr – maximum number of iterations to be performed
- x0, x1 – the limits within which the root lies
- x2 – the value of root at nth iteration
- x3 – the value of root at (n+1)th iteration
- allerr – allowed error
- x – value of root at nth iteration in the regula function
- f(x0), f(x1) – the values of f(x) at x0 and x1 respectively

f(x) = cos(x) – x*e^x

Algorithm:

1. Start
2. Read values of x0, x1 and e

   *Here x0 and x1 are the two initial guesses

   e is the degree of accuracy or the absolute error i.e. the stopping criteria*
3. Computer function values f(x0) and f(x1)
4. Check whether the product of f(x0) and f(x1) is negative or not.

   If it is positive take another initial guesses.

   If it is negative then goto step 5.
5. Determine:

   x = [x0*f(x1) – x1*f(x0)] / (f(x1) – f(x0))
6. Check whether the product of f(x1) and f(x) is negative or not.

   If it is negative, then assign x0 = x;

   If it is positive, assign x1 = x;
7. Check whether the value of f(x) is greater than 0.00001 or not.

   If yes, goto step 5.

   If no, goto step 8.

   *Here the value 0.00001 is the desired degree of accuracy, and hence the stopping

8. Display the root as x.
9. Stop
10.

/* Program to implement Regular-falsi method */

```
#include<stdio.h>
#include<math.h>
float f(float x)
{
   return cos(x) - x*exp(x);
}
void regula (float *x, float x0, float x1, float fx0, float fx1, int *itr)
{
   *x = x0 - ((x1 - x0) / (fx1 - fx0))*fx0;
   ++(*itr);
   printf("Iteration no. %3d X = %7.5f \n", *itr, *x);
}
void main ()
{
   int itr = 0, maxmitr;
   float x0,x1,x2,x3,allerr;
   printf("\nEnter the values of x0, x1, allowed error and maximum iterations:\n");
   scanf("%f %f %f %d", &x0, &x1, &allerr, &maxmitr);
   regula (&x2, x0, x1, f(x0), f(x1), &itr);
   do
   {
      if (f(x0)*f(x2) < 0)
         x1=x2;
      else
         x0=x2;
      regula (&x3, x0, x1, f(x0), f(x1), &itr);
      if (fabs(x3-x2) < allerr)
      {
         printf("After %d iterations, root = %6.4f\n", itr, x3);
         return 0;
      }
      x2=x3;
   }
   while (itr<maxmitr);
   printf("Solution does not converge or iterations not sufficient:\n");
   return 1;
}
```

OUTPUT :

```
Enter the value of x0: -1

Enter the value of x1: 1

_____________________________________________________________

   x0        x1        x2        f0      f1      f2
_____________________________________________________________

-1.000000 1.000000 0.513434 -4.540302 1.459698 -0.330761
0.513434  1.000000 0.603320 -0.330761 1.459698 -0.013497
0.603320  1.000000 0.606954 -0.013497 1.459698 -0.000527
0.606954  1.000000 0.607096 -0.000527 1.459698 -0.000021
```

App.root = 0.607096

Experiment No: 4(b) Newton Raphson methods
AIM: Write a program to implement Newton Raphson methods.

Algorithm:

1. Start
2. Read x, e, n, d
   *x is the initial guess
   e is the absolute error i.e the desired degree of accuracy
   n is for operating loop
   d is for checking slope*
3. Do for i =1 to n in step of 2
4. f = f(x)
5. f1 = f'(x)
6. If ([f1] < d), then display too small slope and goto 11.
   [ ] is used as modulus sign
7. x1 = x – f/f1
8. If ([(x1 – x)/x1] < e), the display the root as x1 and goto 11.
   [ ] is used as modulus sign
9. x = x1 and end loop
10. Display method does not converge due to oscillation.
11. Stop

/* Program to implement Newton Raphson methods */

```
#include<stdio.h>
#include<math.h>
float f(float x)
{
   return x*log10(x) - 1.2;
}
float df (float x)
{
   return log10(x) + 0.43429;
}
void main()
{
   int itr, maxmitr;
   float h, x0, x1, allerr;
   printf("\nEnter x0, allowed error and maximum iterations\n");
   scanf("%f %f %d", &x0, &allerr, &maxmitr);
```

```
        h=f(x0)/df(x0);
        x1=x0-h;
        printf(" At Iteration no. %3d, x = %9.6f\n", itr, x1);
        if (fabs(h) < allerr)
        {
           printf("After %3d iterations, root = %8.6f\n", itr, x1);
           return 0;
        }
        x0=x1;
    }
    printf(" The required solution does not converge or iterations are insufficient\n");
    return 1;
}
```

OUTPUT :

```
ENTER THE TOTAL NO. OF POWER:::: 3

    x^0::-3

    x^1::-1

    x^2::0

    x^3::1

    THE POLYNOMIAL IS :::  1x^3 0x^2 -1x^1 -3x^0
    INTIAL X1---->3
 ITERATION    X1    FX1    F'X1
  1        2.192  21.000 26.000
 2        1.794  5.344  13.419
 3        1.681  0.980  8.656
 4        1.672  0.068  7.475
 5        1.672  0.000  7.384
THE ROOT OF EQUATION IS 1.671700
```

Experiment No:5(a)Euler' s methods

AIM: Write a program to simulate Euler' s method.

Description:

Solving an ordinary differential equation or initial value problem means finding a clear expression for y in terms of a finite number of elementary functions of x. Euler's method is one of the simplest method for the numerical solution of such equation or problem. This **C program for Euler's method** considers an ordinary differential equations, and the initial values of x and y are known. Mathematically, here, the curve of solution is approximated by a sequence of short lines i.e. by the tangent line in each interval. Using these information, the value of the value of 'y_n' corresponding to the value of 'x_n' is to determined by dividing the length $(x_n – x)$ into n strips.

Therefore, strip width= $(x_n – x)/n$ and $x_n=x_0+ nh$.

Again, if m be the slope of the curve at point, $y_{1=} y_{0 +} m(x_0 , y_o)h$.

Similarly, values of all the intermediate y can be found out.

Below is a source code for **Euler's method in C** to solve the ordinary differential equation **dy/dx = x+y**. It asks for the value of of x_0 , y_0 , x_n and h. The value of slope at different points is calculated using the function 'fun'.

The values of y are calculated in while loop which runs till the initial value of x is not equal to the final value. All the values of 'y' at corresponding 'x' are shown in the output screen.

dy/dx = x+y

Algorithm:

2. Define function
3. Get the values of x0, y0, h and xn
   *Here x0 and y0 are the initial conditions
   h is the interval
   xn is the required value
4. n = (xn – x0)/h + 1
5. Start loop from i=1 to n
6. y = y0 + h*f(x0,y0)
   x = x + h
7. Print values of y0 and x0
8. Check if x < xn
   If yes, assign x0 = x and y0 = y
   If no, goto 9.
9. End loop i
10. Stop

/* Program to simulate Euler' s method */

```
#include<stdio.h>
float fun(float x,float y)
{
   float f;
   f=x+y;
   return f;
}
main()
{
   float a,b,x,y,h,t,k;
   printf("\nEnter x0,y0,h,xn: ");
   scanf("%f%f%f%f",&a,&b,&h,&t);
   x=a;
   y=b;
   printf("\n  x\t  y\n");
   while(x<=t)
   {
      k=h*fun(x,y);
      y=y+k;
      x=x+h;
      printf("%0.3f\t%0.3f\n",x,y);
   }
}
```

OUTPUT :

```
Enter the value of range: 1 1.5
Enter the value of y1: 5
Enter the h: 0.1
  y1 = 5.000

  x = 1.000 => y2 = 5.500

  x = 1.100 => y3 = 6.105
```

x = 1.300 => y5 = 7.726

x = 1.400 => y6 = 8.808

x = 1.500 => y7 = 10.129

Experiment No:5(b) Runga-Kutta methods
AIM: Write a program to simulate Runga-Kutta methods.

/* Program to simulate Runga-Kutta methods*/

```
#include<stdio.h>
#include<math.h>
float f(float x,float y);
int main()
{
   float x0,y0,m1,m2,m3,m4,m,y,x,h,xn;
   printf("Enter x0,y0,xn,h:");
   scanf("%f %f %f %f",&x0,&y0,&xn,&h);
   x=x0;
   y=y0;
   printf("\n\nX\t\tY\n");
   while(x<xn)
   {
      m1=f(x0,y0);
      m2=f((x0+h/2.0),(y0+m1*h/2.0));
      m3=f((x0+h/2.0),(y0+m2*h/2.0));
      m4=f((x0+h),(y0+m3*h));
      m=((m1+2*m2+2*m3+m4)/6);
      y=y+m*h;
      x=x+h;
      printf("%f\t%f\n",x,y);
   }
}
float f(float x,float y)
{
   float m;
   m=(x-y)/(x+y);
   return m;
}
```

OUTPUT:

```
Enter the value of x0: 0

Enter the value of y0: 2

Enter the value of h: 0.05

Enter the value of last point: 0.1
    k1 = 0.1000

    k2 = 0.1025

    y(0.0500) = 2.101

    k1 = 0.1026
```

y(0.1000) = 2.205

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR
LAB MANUAL

Title of Course: Electric Circuit Theory Lab
Course Code: EE391
L-T-P scheme: 0-0-3 **Course Credit: 2**

Objectives:
1. To learn and understand to design electrical circuit practically or through any simulation software.
2. To provide an understanding of the circuit designing aspects in bread board.
3. To provide a window to investigate and verify various laws, theories, and concepts regarding electrical circuits practically or virtually by simulation software.

Learning Outcomes: The students will have a detailed knowledge of electrical circuit design using different electrical elements and sources through bread board or by any simulation software. The students will also get the opportunity & better understanding of various concepts, laws, & theories applicable in any electrical circuit by investigating and verifying them in the practically designed circuit.Upon the completion of Operating Systems practical course, the student will be able to:

- **Understand** and implement electrical circuit design knowledge to realize any electrical circuit practically
- **Use** modern simulation software to recreate any practical circuit virtually**.**
- **Understand** the benefits of circuit design in bread board.
- **Analyze** designed circuit to see weather various laws, theories, and concepts regarding electrical circuits holds or not.
- **Simulate**electrical circuits through any simulation software to check weather various laws, theories, and concepts regarding electrical circuits they studied holds or not.

Course Contents:
Exercises that must be done in this course are listed below:
Exercise No.1: Verification of Thevenin's Theorem: Hardware/Simulation
Exercise No. 2: Verification of Norton's Theorem: Hardware/Simulation
Exercise No. 3: Verification of TheoremSuperposition Theorem: Hardware/Simulation
Exercise No. 4: Verification of Maximum Power Transfer: Hardware/Simulation
Exercise No. 5: Study of Z-parameters of any practical circuit treated as Two-port network: Hardware/Simulation
Exercise No. 6: Study of Y-parameters of any practical circuit treated as Two-port network: Hardware/Simulation
Exercise No. 7: Study Resonance of a series RLC circuit: Hardware

Text Book:
1. S.P.Ghosh&A.Chakraborty , "Circuit Theory & Networks", TMH
2. Muhammad H. Rashid, "Introduction to PSpice Using Orcad for circuits and Electronics", Pearson Education.

Recommended Systems/Software Requirements:
1. MATLAB
2. SPICE.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

LAB MANUAL

Aim: To Verify the Thevenin's Theorem in breadboard or through MATLAB/SPICE.

Description:

APPARATUS REQUIRED:-If Practically by circuit design

(i) Bread Board
(ii) Connecting Wire
(iii) Different values of resistances
(iv) A Dc power Source

If by any simulation software

(i) MATLAB/SPICE

THEORY:

Sometimes, we wish to determine the response in a single load resistance in a network. Thevenin Theorem enables us to replace the remainder of the network by a simple equivalent circuit. Determining response in the load resistance, then becomes easier. The use of Thevenin Theorem is specially very helpful and time saving when we wish to find the response for different values of load resistance. Thevenin Theorem states that current through a load resistance connected across any two points of an active network can be obtained by the formula:

$$I_L=V_{th}/(R_{th}+R_L)$$

Where V_{th} is the open circuit voltage at the terminals of R_L when R_L is disconnected and R_{th} is the equivalent resistance viewed from the load terminals when all the sources replaced by their internal resistance only(Deactivating all the sources).

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance and circuit are given in the lab.

CALCULATIONS:

Calculate the theoretical/simulation data's of the given circuit

OBSERVATION TABLE:

Values	V_{th}	R_{th}	I_L
Theoretical Value			
Practical Value			

Percentage Error= [(Observed-Calculated)/Calculated]*100

RESULT:

The percentage error is found to be__%.

DISCUSSION:

Experiment No: 2.Verification of Norton's Theorem experimentally
Aim: To Verify the Norton's Theorem in breadboard or through MATLAB/SPICE.

Description:

APPARATUS REQUIRED:-If Practically by circuit design

(v) Bread Board
(vi) Connecting Wire
(vii) Different values of resistances
(viii) A Dc power Source

If by any simulation software

(ii) MATLAB/SPICE

THEORY:

Sometimes, we wish to determine the response in a single load resistance in a network. Norton's Theorem enables us to replace the remainder of the network by a simple equivalent circuit. Determining response in the load resistance, then becomes easier. The use of Norton Theorem is specially very helpful and time saving when we wish to find the response for different values of load resistance. Thevenin Theorem states that current through a load resistance connected across any two points of an active network can be obtained by the formula:

$$I_L=(I_N*R_N)/(R_N+R_L)$$

Where I_N is the Short circuit current at the terminals of R_L when R_LShort circuited and wefind out the short circuit current through the short circuit terminal and R_N is the norton'sequivalent resistance viewed from the load terminals when all the sources are replaced by their internal resistance only(Deactivatingall the sources).

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance and circuit are given in the lab.

CALCULATIONS:

Calculate the theoretical/simulation data's of the given circuit

OBSERVATION TABLE:

Values	I_N	R_N	I_L
Theoretical Value			
Practical Value			

Percentage Error= [(Observed-Calculated)/Calculated]*100

RESULT:

The percentage error is found to be__%.

DISCUSSION:

Experiment No: 3.Verification of Superposition Theorem experimentally
Aim: To Verify the Superposition Theorem in breadboard or through MATLAB/SPICE.

Description:

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

LAB MANUAL

(i) Bread Board
(ii) Connecting Wire
(iii) Different values of resistances
(iv) A Dc power Source

If by any simulation software

(i) MATLAB/SPICE

THEORY:

Superposition theorem states that in a linear network containing several independent sources, the overall response at any point in the network equals the sum of responses due to each independent source considered separately with all other independently sources made inoperative(short circuited). To make a source inoperative, it is short circuited leaving behind its internal resistance if it is a voltage source, and it is open circuited leaving behind its internal resistance if it is a current source.

In most electrical circuit analysis problems, a circuit is energized by a single independent energy source. In such cases, it is quite easy to find the response (i.e., current, voltage, power) in a particular branch of the circuit using simple network reduction techniques (i.e., series parallel combination, star delta transformation, etc.).

However, in the presence of more than one independent source in the circuit, the response cannot be determined by direct application of network reduction techniques. In such a situation, the principle of superposition may be applied to a linear network, to find the resultant response due to all the sources acting simultaneously.

The superposition theorem is based on the principle of superposition. The principle of Superposition states that the response (a desired current or the voltage) at any point in the linear network having more than one independent source can be obtained as the sum of responses caused by the separate independent sources acting alone. The validity of principle of superposition means that the presence of one excitation sources does not affect the response due to other excitations.

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance and circuit are given in the lab.

CALCULATIONS:

Calculate the theoretical/simulation data's of the given circuit

OBSERVATION TABLE:

Values	I_3	I_3'	I_3''
Theoretical Value			
Practical Value			

I_3 = Current through the load terminal when we deactivate second source and consider for first source

I_3'= Current through the load terminal when we deactivate first source and consider for second source

I_3''= I_3+I_3'

Percentage Error= [(Observed-Calculated)/Calculated]*100

RESULT:

The percentage error is found to be__%.

DISCUSSION:

Experiment No: 4.Verification of Maximum Power Transfer Theorem experimentally
Aim: To Verify the Maximum Power Transfer Theorem in breadboard or through MATLAB/SPICE.

Description:

APPARATUS REQUIRED:-If Practically by circuit design

(ii) Connecting Wire
(iii) Different values of resistances
(iv) A Dc power Source

If by any simulation software

(i) MATLAB/SPICE

THEORY:

This theorem is applicable for analysing communication networks. According to this theorem" a resistive load will draw the maximum power from a network when the load resistance is equal to the resistance of the network as viewed from its output terminals, with all energy sources removed leaving behind their internal resistances." If R_L is the load resistance connected across terminals a and b which consist of variable DC supply and internal resistance is R_S, then according to this theorem, the load resistance will draw maximum power when it is equal to R_S i.e. $R_L = R_S$.

And the maximum power drawn= $\mathbf{V^2_{oc}/4\ R_L}$

Where, Voc is the open circuit voltage at the terminals from which R_L is disconnected.

The variable resistor taken should be larger than fixed resistor. Then only power can be calculated.

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance and circuit are given in the lab.

CALCULATIONS:

Calculate the theoretical/simulation data's of the given circuit

OBSERVATION TABLE:

S.No	Load Resistance(R_L)	I_L(Load Current)	Power($P=I_L^2*R_L$)

RESULT:

Plot a graph between load resistance and power and observe that the power will be maximum when (Load resistance= Internal Resistance)

DISCUSSION:

Experiment No: 5.Study of Z-parameters of a Two-port network experimentally
Aim: To Study Z-parameters of any practical circuit treated as Two-port network in breadboard or through MATLAB/SPICE.

Description:

APPARATUS REQUIRED:-If Practically by circuit design

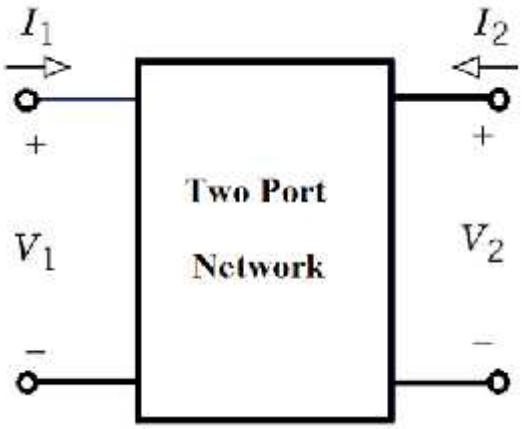
(iii) Different values of resistances
(iv) A Dc power Source

If by any simulation software

(i) MATLAB/SPICE

THEORY

A two-Port network basically consists in isolating either a complete circuit or part of it and finding its characteristics parameters. Once this is done, the isolated part of the circuit becomes a "black box" with a set of distinctive properties, enabling us to abstract away its specific build up, thus simplifying analysis.



Here,

V_1 = Input voltage

V_2 = Output voltage

I_1 = Input current

I_2 = output current

Z-model: In the Z-model or impedance model the two currents I_1 & I_2 are assumed to be known and the voltage V_1& V_2 can be found by:

$$\begin{pmatrix} V1 \\ V2 \end{pmatrix} = \begin{pmatrix} Z11 & Z12 \\ Z21 & Z22 \end{pmatrix} \begin{pmatrix} I1 \\ I2 \end{pmatrix}$$

Where,

Z11 = $V1/I1$ taking I2=0

Z12 = $V1/I2$ taking I1=0

Z21 = $V2/I1$ taking I2=0

Z22 = $V2/I2$ taking I1=0

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance and circuit are given in the lab.

CALCULATIONS:

Calculate the theoretical/simulation data's of the given circuit

OBSERVATION TABLE:

Sl. No.	When Output is open circuited (i.e. I2 =0)			When Input is open circuited (i.e. I1 =0)		
	V1	V2	I1	V1	V2	I2
1.						
2.						
3.						
4.						
5.						

Percentage Error= [(Observed-Calculated)/Calculated]*100

RESULT:

The percentage error is found to be__%.

DISCUSSION:

Experiment No: 6. Study of Y-parameters of a Two-port network experimentally
Aim: To Study Y-parameters of any practical circuit treated as Two-port network in breadboard or through MATLAB/SPICE.

Description:

APPARATUS REQUIRED:-If Practically by circuit design

(i) Bread Board
(ii) Connecting Wire
(iii) Different values of resistances

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

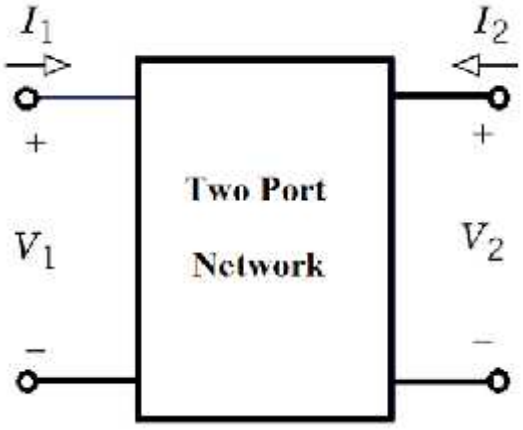
LAB MANUAL

If by any simulation software

(i) MATLAB/SPICE

THEORY

A two-Port network basically consists in isolating either a complete circuit or part of it and finding its characteristics parameters. Once this is done, the isolated part of the circuit becomes a "black box" with a set of distinctive properties, enabling us to abstract away its specific build up, thus simplifying analysis.



Here,

V_1 = Input voltage

V_2 = Output voltage

I_1 = Input current

I_2 = output current

Y-model: In the Y-model or admittance model, the two voltages V_1& V_2 are assumed to be known and the currents I_1 & I_2can be found by:

$$\begin{pmatrix} I1 \\ I2 \end{pmatrix} = \begin{pmatrix} Y11 & Y12 \\ Y21 & Y22 \end{pmatrix} \begin{pmatrix} V1 \\ V2 \end{pmatrix}$$

Where,

Y11 = $I1/V1$ taking V2=0

Y12 = $I1/V2$ taking V1=0

Y21 = $I2/V1$ taking V2=0

Y22 = $I2/V2$ taking V1=0

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance and circuit are given in the lab.

CALCULATIONS:

Calculate the theoretical/simulation data's of the given circuit

OBSERVATION TABLE:

Sl. No.	When Output is short circuited (i.e. V2 =0)			When Input is short circuited (i.e. V1 =0)		
	V1	I1	I2	V2	I1	I2
1.						
2.						
3.						
4.						
5.						

Percentage Error= [(Observed-Calculated)/Calculated]*100

RESULT:

The percentage error is found to be__%.

DISCUSSION:

Experiment No: 7. Study of Resonance experimentally
Aim: To Study Resonance of a series RLC circuit in breadboard

Description:

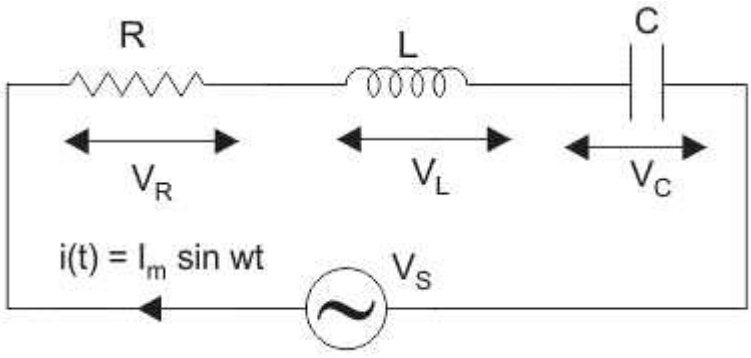
APPARATUS REQUIRED:-

(i) Bread Board
(ii) Connecting Wire
(iii) Different values of resistances
(iv) A Dc power Source

THEORY

Resonance occurs in an electrical circuit excited by AC source when the net inductive reactance (X_L) and net capacitive reactance (X_C) become equal either because for a fixed frequency as circuit's inductance (L) is equal to capacitance (C) or due to a particular frequency where $X_L = X_C$.

A simple series RLC circuit



When stated above condition arises in a circuit as given here the net impedance (Z_{net}) becomes minimum or equal to only resistance (R) of the circuit and the circuit starts operating in resonance with unity power factor. At resonance the value of net current is maximum as the net impedance is minimum.

CIRCUIT DIAGRAM:

Draw the circuit diagram as per the resistance, inductance & capacitancevalues as given in the lab.

CALCULATIONS:

Calculate the theoretical data's of the given circuit to find out the value of net impedance and current at resonance.

OBSERVATION TABLE:

1.	f_1	
2.	f_2	
3.	f_0	
4.	f_3	
5.	f_4	

Here, f_0 is resonance frequency and $f_1<f_2<f_0<f_3<f_4$

Percentage Error= [(Observed-Calculated)/Calculated]*100

RESULT:

The percentage error is found to be__%.

DISCUSSION:

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

LAB MANUAL

Title of Course: Technical Report Writing &Language Lab
Course Code: HU381
L-T-P scheme: 0-0-2 **Course Credit: 2**

Objectives:

1. To inculcate a sense of confidence in the students.
2. To help them become good communicators both socially and professionally.
3. To assist them to enhance their power of Technical Communication.

Learning Outcomes:

Course Contents:
Exercises that must be done in this course are listed below:
Exercise No.1: Report Types (Organizational/Commercial/Business/Project)

Exercise No. 2: Report Format &Organization of Writing Materials

Exercise No. 3: Report Writing (Practice Sessions &Workshops)

Exercise No. 4: Introductory Lecture to help the students geta clearidea of Technical Communication& the need of Language Laboratory Practice Sessions

Exercise No. 5: Conversation Practice Sessions: (To be done as real life interactions)
a) Training the students by using Language Lab Device/Recommended Texts/cassettes/cd to get their Listening Skill & Speaking killhoned
b) Introducing Role Play & honing overall Communicative Competence

Exercise No. 6: Group Discussion Sessions:
a) Teaching Strategies of Group Discussion
b) Introducing Different Models & Topics of Group Discussion
c) Exploring Live/Recorded GD Sessions for mending students' attitude/approach & fortaking remedial measure Interview Sessions;
d) Training students to face Job Interviews confidently and successfully
e) Arranging Mock Interviews and Practice Sessions for integrating Listening Skill with Speaking Skillin a formal situation for effective communication

Exercise No. 7: Presentation:
a) Teaching Presentation as a skill
b) Strategies and Standard Practices of Individual/Group Presentation
c) Media & Means of Presentation: OHP/POWERPOINT/Other Audio-Visual Aids

Exercise No. 8: Competitive Examination:
a) Making the studentsawareofProvincial/National/InternationalCompetitiveExaminations
b) Strategies/Tactics for success in Competitive Examinations
c) SWOT Analyse is and its Application in fixing Target

Text Book:
1. Nira Konar: English Language Laboratory: A Comprehensive Manual

D. Sudharani: Advanced Manual for Communication Laboratories& Technical Report Writing Pearson Education (W.B.edition), 2011PHI Learning,2011