ADITYA COLLEGE OF ENGINEERING
(An Autonomous Institution)
www.acem.ac.in

DEPARTMENT OF
ELECTRICAL AND ELECTRONICS ENGINEERING

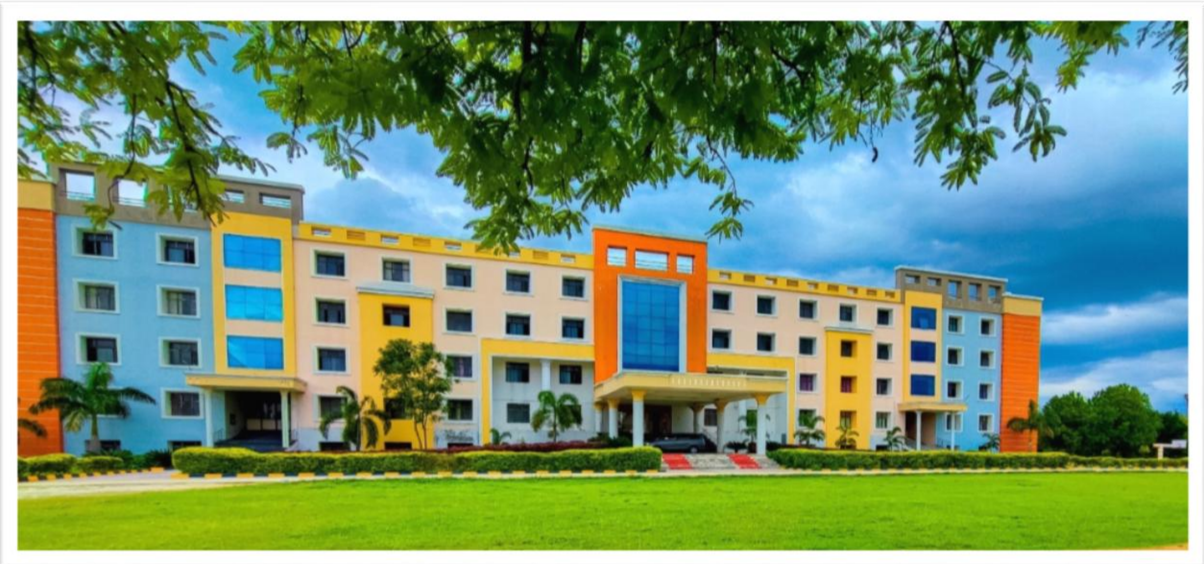
Course Structure
&
Detailed Syllabus

For the students admitted to

B. Tech. Regular Four Year Degree Programme from the Academic Year 2026-27

and

B. Tech. Lateral Entry Scheme from the Academic Year 2027-28



ADITYA COLLEGE OF ENGINEERING, MADANAPALLE

(UGC - Autonomous Institution & Accredited with NAAC A+ Grade)

Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapuramu

Madanapalle - 517325, Annamayya (Dist.), A. P.

www.acem.ac.in

ACEM - R26 REGULATION

ELECTRICAL AND ELECTRONICS ENGINEERING

COURSE STRUCTURE AND SYLLABUS

for

B. Tech. (Regular – Full Time)

(Effective for the Students Admitted into I Year from the Academic Year 2026 - 27 onwards)

and

B. Tech. (Lateral Entry Scheme)

(Effective for the Students getting Admitted into II Year through Later Entry Scheme from the Academic Year 2027 - 28 onwards)

Institute Vision:

To impart quality in engineering education to meet the technological advances and industrial requirements with global standards.

Institute Mission:

- Provide quality technical education through skill-based trainings and promote research and development, and consultancy services.
- Offer state-of-the-art-infrastructure for supporting technological advances.
- Develop disciplined, creative and globally competent engineers.
- Equip and empower the faculty at all levels to promote innovations and technical advancements in various domains of engineering.

Department Vision:

To produce accomplished engineers through eminence education and become a recognized center of excellence in academics, research and technological services in the area of Electrical and Electronics Engineering and related inter-disciplinary fields.

Department Mission:

- To facilitate the students to work with modern tools, state of art technologies, innovative research capabilities besides inculcating leadership abilities.
- To instigate students for propensity to research and innovation by exposing them to industry and societal needs to creating solutions for contemporary problems.
- To prepare the students professionally skilful, intellectually proficient and socially responsible.

Programme Outcomes (POs):

On Successful completion, the graduate will be able to,

- **PO1_Engineering Knowledge:** Apply knowledge of mathematics, natural sciences, engineering fundamentals, and an engineering specialization to a wide range of practical procedures and practices.
- **PO2_Problem Analysis:** Identify and analyse well-defined engineering problems and reach substantiated conclusions using codified methods of analysis specific to their field of activity.
- **PO3_Design/Development of Solutions:** Design solutions for well-defined technical problems and assist in the design of systems, components, or processes to meet specified needs, with appropriate consideration for public health and safety, as well as cultural, societal, and environmental factors.
- **PO4_Conduct Investigations of Complex Problems:** Conduct investigations of well-defined problems by locating and searching relevant codes and catalogues, and by conducting standard tests and measurements.
- **PO5_Engineering Tool Usage:** Apply appropriate techniques, resources, and modern computing, engineering and IT tools to well-defined engineering problems, while being aware of their limitations.

- **PO6_The Engineer and the World:** When solving well-defined engineering problems, evaluate sustainable development impacts to society, the economy, sustainability, health and safety, legal frameworks, and the environment.
- **PO7_Ethics:** Understand and commit to professional ethics and the norms of technician practice, including compliance with relevant laws. Demonstrate an understanding of the need for diversity and inclusion.
- **PO8_Individual and Collaborative Team Work:** Function effectively as an individual, and as a member or leader in diverse and inclusive teams, in multidisciplinary face-to-face/remote/distributed settings.
- **PO9_Communication:** Communicate effectively and inclusively on well-defined engineering activities with the engineering community and society at large by comprehending the work of others, documenting one's own work, and giving and receiving clear instructions.
- **PO10_Project Management and Finance:** Demonstrate awareness of engineering management principles as a member or leader of a technical team and apply them to manage projects in multidisciplinary environments.
- **PO11_Life-Long Learning:** Recognize the need for, and have the ability for, independent updating in the face of specialized technical knowledge.

Program Specific Outcomes (PSOs):

On Successful completion, the graduate (Electrical and Electronics Engineering) will be able to,

- **PSO1:** Graduates will exhibits demonstrable knowledge in the field of Electrical and Electronics Engineering.
- **PSO2:** Ability to transform and excel as solution providers for industry and society challenges.

Programme Educational Objectives (PEOs):

After few years of graduation, the graduates of Electrical and Electronics Engineering are expected to

- **PEO1:** The graduates exhibit strong knowledge in basic sciences and electrical engineering to enable them for higher learning and employment.
- **PEO2:** To prepare the graduates to design electrical, electronics and computing systems innovatively for mankind.
- **PEO3:** To inculcate the graduate professionalism, ethics, effective communication skills and capability to succeed in multi-disciplinary fields.
- **PEO4:** To motivate the graduates and faculty to do useful and application oriented research.

B.Tech. – Electrical and Electronics Engineering
Course Structure & Syllabus – ACEM R26 Regulations
(Applicable from the Academic Year 2026-27 onwards)

INDUCTION PROGRAMME

S.No.	Course Name	Category	L-T-P-C
1.	Physical Activities -- Sports, Yoga and Meditation, Plantation	MC	0-0-6-0
2.	Career Counselling	MC	2-0-2-0
3.	Orientation to all branches -- career options, tools, etc.	MC	3-0-0-0
4.	Orientation on admitted Branch -- corresponding labs, tools and platforms	EC	2-0-3-0
5.	Proficiency Modules & Productivity Tools	ES	2-1-2-0
6.	Assessment on basic aptitude and mathematical skills	MC	2-0-3-0
7.	Remedial Training in Foundation Courses	MC	2-1-2-0
8.	Human Values & Professional Ethics	MC	3-0-0-0
9.	Communication Skills -- focus on Listening, Speaking, Reading, Writing skills	BS	2-1-2-0
10.	Concepts of Programming	ES	2-0-2-0

Group - A

B.Tech. – I Year I Semester

S. No.	Category	Course Code	Name of the Course	L	T	Pr.	P	Credits
1.	BS	26BSHB101T	Linear Algebra & Calculus	3	0	2	0	4
2.	BS	26BSHB104T	Physics for Electrical Systems	2	1	0	0	3
3.	ES	26ES03101T	Engineering Graphics	1	0	4	0	3
4.	ES	26ES05101T	Computational Thinking & Problem Solving with Python	3	0	0	0	3
5.	ES	26ES31101T	AI Tools and Applications	1	0	4	0	3
6.	BS	26BSHB104P	Physics for Electrical Systems Lab	0	0	0	3	1.5
7.	ES	26ES05101P	Computational Thinking & Problem Solving with Python Lab	0	0	0	3	1.5
8.	ES	26ES03102P	Engineering Workshop	0	0	0	3	1.5
9.	HM	26HMHS103L	NSS / NCC / Scouts & Guides / Community Service	0	0	0	1	0.5
			Total	**10**	**1**	**10**	**10**	**21**

B. Tech. – I Year II Semester

S. No.	Category	Course Code	Name of the Course	L	T	Pr.	P	Credits
1.	BS	26BSHB102T	Differential Equations & Vector Calculus	3	0	2	0	4
2.	BS	26BSHB108T	Engineering Chemistry for Electrical Engineers	3	0	0	0	3
3.	HM	26HMHS101T	Communicative English	2	0	0	0	2
4.	ES	26ES03103T	Design Thinking	1	0	4	0	3
5.	PC	26PC02101T	Electromagnetic Fields	2	1	0	0	3
6.	BS	26BSHB108P	Engineering Chemistry for Electrical Engineers Lab	0	0	0	3	1.5
7.	HM	26HMHS101P	Communicative English Lab	0	0	0	2	1
8.	PC	26PC02102P	Electrical Engineering Workshop Practices	0	0	0	2	1
9.	HM	24HMHS102L	Health and Wellness, Yoga and Sports	0	0	0	1	0.5
10.	MC	26ES05102A	Cybersecurity Awareness for Engineers	2	0	0	0	0
			Total	**11+2**	**1**	**6**	**8**	**19**

B. Tech. – II Year I Semester

S. No.	Category	Course Code	Name of the Course	L	T	Pr.	P	Credits
1.	BS	26BSHB211T	Complex Variables and Probability Theory	2	1	0	0	3
2.	HM	26HMHS205T	Managerial Economics and Financial Analysis	2	0	0	0	2
3.	ES	26ES01201T	Sustainability and Green Engineering (AI Integrated)	2	0	0	0	2
4.	EC	26EC04201T	Analog Circuits	2	1	0	0	3
5.	PC	26PC02205T	AI Assisted Electrical Circuit Analysis	2	0	2	0	3
6.	PC	26PC02206T	DC Machines and Transformers with Intelligent Control	2	0	2	0	3
7.	EC	26EC04201P	Analog Circuits Lab	0	0	0	2	1
8.	PC	26PC02205P	AI Assisted Electrical Circuit Analysis Lab	0	0	0	3	1.5
9.	PC	26PC02206P	Intelligent Control of DC Machines and Transformers Lab	0	0	0	3	1.5
10.	SE	26SE05201S	C Programming and Data Structures	1	0	0	2	2
11.	MC	26BSHB214 A	Environmental Science	2	0	0	0	0
			Total	**13+2**	**2**	**4**	**10**	**22**

B. Tech. – II Year II Semester

S. No.	Category	Course Code	Name of the Course	L	T	Pr.	P	Credits
1.	BS	26BSHB213T	Biology for Engineers	2	0	0	0	2
2.	HM	26HMHS204T	Universal Human Values – Understanding Harmony & Ethical Human Conduct	3	0	0	0	3
3.	PC	26PC02207T	AI Assisted Control Systems	2	0	2	0	3
4.	PC	26PC02208T	AI-Assisted Control of Induction and Synchronous Machines	2	0	2	0	3
5.	PC	26PC02209T	Intelligent Power Systems – I	2	0	2	0	3
6.	PC	26PC04203T	AI Integrated Signal Processing and Systems	2	0	2	0	3
7.	PC	26PC02207P	AI Assisted Control Systems Lab	0	0	0	3	1.5
8.	SE	26PC02208P	AI-Assisted Control of Induction and Synchronous Machines Lab	0	0	0	3	1.5
9.	SE	26SE02201S	Electrical Simulation Tools	1	0	0	2	2
10.	MC	26HMHS206A	Technical Paper Writing and IPR	2	0	0	0	0
			Total	**14+2**	**0**	**8**	**8**	**22**
	Mandatory Community Service Project (26IPIN301L) of 08 Weeks duration during Summer Vacation							

B. Tech. – I Year
I Semester

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26BSHB101T	BS	**Linear Algebra & Calculus** (Common to all branches of Engineering)	3	0	2	0	4

Pre-Requisites: Basic algebra, trigonometry, coordinate geometry, differential and integral calculus, vectors, matrices, determinants, and functions of several variables.

Course Objectives:

- To build strong foundations in linear algebra and multivariable calculus with direct engineering relevance.
- To integrate analytical methods with AI-assisted computational tools for visualization, simulation, and verification.
- To develop outcome-oriented problem-solving skills aligned with NEP 2020 and modern engineering applications.
- To apply matrix methods, eigenanalysis, and multivariable calculus to solve engineering problems.
- To use computational tools for approximation, optimization, differentiation, and integration.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply matrix algebra techniques to solve systems of linear equations and analyze matrix properties relevant to engineering models. (L3)

CO2: Analyze eigenvalue problems, diagonalization, and quadratic forms to interpret linear transformations and engineering systems. (L4)

CO3: Apply mean value theorems, Taylor and Maclaurin series, and curvature concepts to approximate and interpret engineering functions. (L3)

CO4: Analyze multivariable functions using partial derivatives, Jacobians, gradients, and optimization techniques. (L4)

CO5: Evaluate areas, volumes, and coordinate transformations using multiple integrals in engineering contexts. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	1		1				2	2	1
CO2	3	3	3	2	1		1				2	2	2
CO3	3	2	2	2	1		1				2	2	1
CO4	3	3	3	3	2		1				2	2	2
CO5	3	3	2	3	3		1				2	2	1

Unit I: Matrix Algebra and Linear Systems:

Core Content: Matrices, operations, transpose, inverse, types of matrices, symmetric and skew-symmetric matrices; orthogonal, Hermitian, skew-Hermitian and unitary matrices. determinant properties; rank of a matrix; echelon and normal forms; system of linear equations; consistency conditions; Gauss elimination and Gauss-Jordan methods; iterative methods: Jacobi and Gauss-Seidel;

AI Integration: Python-based solution of large linear systems using NumPy; visualization of convergence in iterative methods; AI-assisted debugging of matrix algorithms; using ChatGPT/Copilot to explain row operations and rank reduction.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; SymPy for symbolic matrix operations; Jupyter Notebook; Scilab.

Unit II: Eigenvalues, Eigenvectors, Diagonalization & Quadratic Forms:

Core Content: Eigenvalues and eigenvectors; Properties, characteristic equation; Cayley-Hamilton theorem; diagonalization; similarity transformation; powers and inverse of matrices using Cayley-Hamilton theorem; quadratic forms; canonical form; reduction by orthogonal transformation; rank, index, signature; Nature of Quadratic form;

AI Integration: AI-assisted eigenvalue computation and verification; Python/SciPy visualization of eigen modes and quadratic surfaces; use of AI tools to explain diagonalization and matrix classification; numerical checking of Cayley-Hamilton theorem.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib); SymPy; GNU Octave; Google Colab.

Unit III: Single Variable Calculus and Series Expansions:

Core Content: Rolle’s theorem; Lagrange’s mean value theorem; Cauchy’s mean value theorem; Taylor’s and Maclaurin’s series with remainder; approximation of functions; error estimation; curvature of plane curves; radius and centre of curvature; circle of curvature; applications to engineering approximations and curve design.

AI Integration: SymPy-based Taylor and Maclaurin expansion generation; plotting truncation error and convergence; AI-supported derivation checking for mean value theorems; curvature plot visualization and interpretation.

Free/Open-Source AI Tools: Python (SymPy, NumPy, Matplotlib); Google Colab; Jupyter Notebook; WolframAlpha free tier for verification.

Unit IV: Multivariable Differentiation and Optimization:

Core Content: Functions of several variables; limits and continuity; partial derivatives; higher order partial derivatives; total derivative; chain rule; gradient; tangent plane and normal line; Jacobians; functional dependence; Taylor expansion of two-variable functions; maxima and minima; second derivative test; Lagrange multipliers and constrained optimization; divergence and curl.

AI Integration: Gradient field visualization using Python; AI-assisted Jacobian computation and verification; constrained optimization using SciPy; use of AI tools to interpret saddle points, extrema, and optimization constraints.

Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy, Matplotlib); Google Colab; Jupyter Notebook; SciPy optimize module.

Unit V: Multiple Integrals and Coordinate Transformations:

Core Content: Double integrals; evaluation over type I and type II regions; change of order of integration; triple integrals; change of variables; polar, cylindrical and spherical coordinates; area and volume computation; mass, centroid and moment calculations; engineering applications of multiple integration.

AI Integration: Numerical evaluation of double and triple integrals using SciPy; 2D and 3D region visualization; AI-assisted generation of integration code; interpreting Jacobian as geometric scaling factor in transformed coordinates.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib); Google Colab; SymPy; Jupyter Notebook; 3D plotting tools.

Text Books:

1. B. S. Grewal, Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, Wiley, 2018.
3. G. B. Thomas, M. D. Weir, J. Hass, Thomas’ Calculus, 14th Edition, Pearson, 2018.

4. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5th Edition, Alpha Science International, 2021.

Reference Books:

1. Gilbert Strang, Introduction to Linear Algebra, 5th Edition, Wellesley-Cambridge Press, 2016.
2. David C. Lay, Steven R. Lay, Judi J. McDonald, Linear Algebra and Its Applications, 5th Edition, Pearson, 2015.
3. George B. Arfken, Hans J. Weber, Frank E. Harris, Mathematical Methods for Physicists, 7th Edition, Academic Press, 2012.
4. Dennis G. Zill and Warren S. Wright, Advanced Engineering Mathematics, 6th Edition, Jones & Bartlett, 2018.
5. E. Kreyszig, Elementary Linear Algebra, 2nd Edition, Wiley, 1983.

Web Resources & NPTEL Links:

1. NPTEL: Linear Algebra – https://nptel.ac.in
2. MIT OpenCourseWare – Linear Algebra: https://ocw.mit.edu
3. MIT OpenCourseWare – Multivariable Calculus: https://ocw.mit.edu
4. Python SymPy Documentation: https://docs.sympy.org
5. Google Colab for Python: https://colab.research.google.com
6. SciPy Documentation: https://docs.scipy.org
7. NumPy Documentation: https://numpy.org/doc

Recommended Free & Open-Source AI/Computational Tools:

1. Python (NumPy, SciPy, Matplotlib) – FREE on Google Colab for matrix computation, visualization, and numerical analysis.
2. SymPy (free, open-source) – For symbolic algebra, calculus, matrix operations, and verification.
3. scikit-learn (free, open-source) – For regression, classification, and exploratory predictive modeling.
4. Jupyter Notebook (free, open-source) – For interactive computation and lab documentation.
5. OpenCV (free, open-source) – For image-based experimentation and visualization support where needed.
6. GNU Octave (free, open-source) – MATLAB-like numerical computing for matrix and calculus workflows.
7. TensorFlow / Keras (free, open-source) – For optional AI demonstrations in regression and classification tasks.

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26BSHB104T	BS	Physics for Electrical System	2	1	0	0	3

Pre-Requisites: Engineering Mathematics, Basic Electrical Engineering, and Engineering Physics Fundamentals

Course Objectives:

- To provide knowledge on wave optics concepts and their applications with basic AI-assisted optical analysis and sensing techniques.
- To introduce the electrical properties of metals and semiconductors with simple AI-assisted conductivity and material analysis methods.
- To equip students with the fundamentals of quantum mechanics and quantum computing with basic AI-assisted secure communication concepts.
- To familiarize students with dielectric and magnetic materials used in power systems along with basic AI-assisted monitoring and predictive analysis techniques.
- To impart knowledge on electromagnetic field theory and its applications in power systems with simple AI-assisted EMI prediction and shielding concepts.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the principles of interference, diffraction, and polarization with basic AI-assisted optical analysis in engineering applications. (L3)

CO2: Analyze the electrical properties of metals and semiconductors using basic AI-assisted material characterization and conductivity analysis techniques. (L4)

CO3: Apply the concepts of quantum mechanics and quantum computing with basic AI-assisted secure communication applications. (L3)

CO4: Analyse dielectric and magnetic material behaviour in power systems using basic AI-assisted monitoring and predictive maintenance techniques. (L4)

CO5: Apply and analyze Maxwell's equations and electromagnetic field concepts for energy transport, AI-based EMI detection and mitigation in power electronic and smart grid systems. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	3		1				2	2	1
CO2	3	3	2	2	3		1				2	2	2
CO3	3	3	2	2	2		1				2	1	2
CO4	3	3	3	2	3		1				2	2	2
CO5	3	3	2	3	3		1				2	2	2

Unit I: Applied Wave Optics:

Core Content: Interference: principle of superposition and coherence, types of interference, interference in thin films (reflection geometry) and Newton's rings, application — determination of wavelength and refractive index. Diffraction: introduction to diffraction, Fresnel and Fraunhofer diffraction, Fraunhofer diffraction due to single, double and N-slits (diffraction grating), dispersive power and resolving power of grating, application — LiDAR for autonomous sensing in electric vehicles (EV). Polarization: types of polarization, Nicol's prism, half-wave and quarter-wave plates.

AI Integration: AI-powered LiDAR integration for autonomous EV sensing and navigation; Python-based fringe pattern analysis for semiconductor wafer surface profiling.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn / basic DL models.

Unit II: Electrical Properties of Metals and Semiconductors:

Core Content: Classical free electron theory — merits and demerits; quantum free electron theory — equation for electrical conductivity; Fermi-Dirac statistics; Fermi level and variation with temperature; classification of solids — metals, insulators and semiconductors. Semiconductors: introduction, Fermi energy and Fermi level in intrinsic and extrinsic semiconductors, expression for concentration of electrons in conduction band and holes in valence band, electrical conductivity, expressions for drift and diffusion currents, Hall effect, expression for Hall coefficient and its applications.

AI Integration: AI-assisted prediction of electrical conductivity in metals and semiconductors; AI-assisted carrier concentration and conductivity analysis in semiconductors.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

Unit III: Introduction to Quantum Mechanics & Quantum Computing:

Core Content: Introduction to quantum mechanics, de-Broglie concept (matter waves), Heisenberg's Uncertainty Principle, wave function, postulates of quantum mechanics, Schrodinger's time-independent wave equation, particle in a one-dimensional potential well, quantum tunnelling (basic idea). Introduction to quantum computing: classical bits vs qubits, concept of superposition and entanglement, basic quantum operations — Hadamard gate (idea of superposition), Pauli-X gate, basic idea of quantum measurement. Applications (qualitative only): AI-assisted tunnel diodes and cold emission of electrons, basic idea of quantum cryptography and secure communication.

AI Integration: AI-assisted analysis of tunnel diode characteristics using quantum tunnelling concepts; AI-assisted quantum key distribution for secure communication.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

Unit IV: Dielectrics and Magnetic Materials for Power Systems:

Core Content: Dielectric materials: introduction, basic definitions, types of polarizations, internal fields in solids and the Clausius-Mossotti relation, frequency dependence of the polarisation, dielectric loss; application (qualitative only) — AI-assisted monitoring of solid, liquid and gaseous dielectrics in high voltage applications. Magnetic materials and energy conversion: introduction and classification (dia, para, ferro, ferri, and anti-ferro); hysteresis and eddy current losses in electrical machines; skin effect; soft & hard magnetic materials; applications (basic idea) — transformer cores, armatures, high-frequency inductors and permanent magnets for EV motors and wind generators.

AI Integration: AI-based hysteresis loop analysis and eddy current loss estimation; AI-driven design of permanent magnets for EV motors.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

Unit V: Electromagnetic Fields in Power Systems:

Core Content: Introduction, wave equation in differential form in free space, vector analysis, divergence and curl of electric field and magnetic field (static), Gauss' divergence theorem and Stokes' theorem; derivations of the four Maxwell equations and their significance; Maxwell's equations in vacuum, energy transport and the Poynting vector. Applications (qualitative only): role of Maxwell's equations in electromagnetic interference (EMI), EMI in power converters (switching noise in IGBT/MOSFET-based inverters).

AI Integration: AI-assisted EMI prediction in IGBT/MOSFET-based power converters; AI-optimised electromagnetic shielding for power electronics.

Free/Open-Source AI Tools: Google Colab, Python and scikit-learn.

Text Books:

1. Avadhanulu, Kshirsagar & Arun Murthy, Engineering Physics, S. Chand, 2019.

2. D. K. Bhattacharya & Poonam Tandon, Engineering Physics, Oxford Press, 2015.

Reference Books:

1. Shatendra Sharma & Jyotsna Sharma, Engineering Physics, Pearson, 2018.
2. S. O. Pillai, Solid State Physics, New Age International.
3. Nielsen & Chuang, Quantum Computation & Quantum Information, Cambridge.
4. Streetman & Banerjee, Solid State Electronic Devices, Pearson.
5. C. Kittel, Introduction to Solid State Physics, Wiley.
6. A. Von Hippel, Dielectrics and Waves, Wiley.
7. D. J. Griffiths, Introduction to Electrodynamics, Cambridge.
8. W. H. Hayt & J. A. Buck, Engineering Electromagnetics, McGraw Hill.

Web Resources & NPTEL Links:

1. NPTEL – nptel.ac.in
2. MIT OpenCourseWare – ocw.mit.edu
3. HyperPhysics – hyperphysics.phy-astr.gsu.edu
4. Feynman Lectures – feynmanlectures.caltech.edu
5. PhET Simulations – phet.colorado.edu
6. Library of Congress – loc.gov/rr/scitech/selected-internet/physics.html

Recommended Free & Open-Source AI/Computational Tools:

1. Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and EM field analysis.
2. IBM Qiskit (free, open-source) – for quantum computing and quantum cryptography simulations on Google Colab.
3. scikit-learn (free, open-source) – for semiconductor property prediction and EMI data analysis.
4. OpenEMS (free, open-source) – for 3D electromagnetic field simulation and power system analysis.
5. PhET Interactive Simulations (free) – for wave optics, quantum mechanics and semiconductors.

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES03101T	ES	**Engineering Graphics** (Common to all branches of Engineering)	1	0	4	0	3

Pre-Requisites: Basic Geometry, and Fundamental Python Programming

Course Objectives:

- To familiarise with the BIS conventions, geometric construction standards, and engineering scales.
- To develop visualization skills to interpret and project points, lines, and regular planes into 2D drawing sheets.
- To impart foundational knowledge on projecting regular 3D solids and reading directional geometric orientations.
- To enable the spatial reconstruction of sliced solids and flat unfolded developments of various structural shells.
- To introduce CAD tools combined with generative AI software to construct multi-view isometric representations from 3D models.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Construct engineering curves and scales using traditional drafting geometry validated by automated algorithmic paths. (L3)

CO2: Solve multi-planar projection instances of linear points, segments, and standard regular planar silhouettes. (L3)

CO3: Render projections of standard 3D solids and identify variations using structural boundary algorithms. (L4)

CO4: Draft complex cross-sectional geometries and mapped surface extensions of sliced geometric forms. (L4)

CO5: Produce composite isometric layouts alongside generative text-to-CAD system alternatives for dynamic prototyping. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1			2	1				1	1	1
CO2	3	3	2			2	1				1	1	1
CO3	3	3	2	1		2	1				1	1	2
CO4	3	3	3	1	2	2	1				2	2	2
CO5	3	2	2		3	2	1				2	2	2

Unit 1: Introduction to Engineering Graphics, Curves, and Scales:

Core Content:

Introduction: Principles of Engineering Graphics, signboards, BIS conventions, and lettering.

Curves: Construction of Polygons. Conic sections: Ellipse, Parabola, and Hyperbola using eccentricity/general methods. Cycloidal curves and Involutes.

Scales: Plain, Diagonal, and Vernier scales.

Hands-On Experience: Manual draft generation of an ellipse and diagonal scale, paired with a software verification loop.

AI Integration Module: AI prompt engineering for generating geometric profiles; algorithmic generation of conic curves using mathematical code blocks; parsing hand-drawn geometric sketches into clean vector shapes using Computer Vision (CV) edge detection (Canny Edge/Hough Transform).

Unit II: Orthographic Projections of Points, Lines:

Core Content: Principles of orthographic projections, projection of points in all four quadrants. Projection of straight lines inclined to one or both reference planes, traces of lines, determination of true lengths and true inclinations.

Projections of Planes: regular planes Perpendicular to both reference planes, parallel to one reference plane and inclined to the other reference plane; plane inclined to both the reference planes.

Hands-On Experience & Application:

Project: Construct a transformation script where a user inputting 2D projection endpoint automatically outputs the 3D line representation, true spatial length, and inclination angles without manual geometric tracking.

Application: Calibration setups for stereoscopic robotic arms tracking trajectories across multiple camera planes.

AI Integration Module: Machine learning pipelines for 3D coordinate estimation from 2D views; multi-view structural tracking; training neural models to predict true line length and orientation angles based solely on coordinate projections.

Unit III: Projection of Regular Solids:

Core Content:

Projection of Regular Solids: Types of solids: Polyhedra and Solids of revolution. Projections of solids in simple positions: Axis perpendicular to horizontal plane, Axis perpendicular to vertical plane and Axis parallel to both the reference planes, Projection of Solids with axis inclined to one reference plane and parallel to another plane.

Hands-On Experience & Application:

Project: Set up a classification notebook using scikit-learn or a simple convolutional model to automatically identify the type of solid (e.g., hexagonal prism vs. cone) present within an orthographic engineering layout image.

Application: Automated component sorting inside factory staging bays and conveyor streams.

AI Integration Module: Object recognition models (e.g., YOLO / Mobile Net) trained to identify standard solid geometries; automated multi view projection generation from 3D CAD meshes using programmatic render layers.

Unit IV: Sections of Solids, Development of Surfaces, and AI Generative Design:

Core Content:

Sections of Solids: Perpendicular and inclined section planes, Sectional views and True shape of section, Sections of solids in simple position only.

Development of Surfaces: Methods of Development: Parallel line development and radial line development. Development of a cube, prism, cylinder, pyramid and cone.

Hands-on Experience & Application:

Project: Use a nesting optimization script (e.g., using genetic algorithms or standard heuristics) to arrange several custom flat-surface solid developments onto a finite raw sheet grid, actively minimizing material margins and cut lines.

Application: Sheet metal stamping pattern automation in automotive and aerospace chassis manufacturing.

AI Integration Module: Generative Adversarial Networks (GANs) for generating flat-pattern configurations; algorithmic optimization of surface layouts to minimize sheet manufacturing scrap; predictive toolpath generation using physics-guided heuristic networks.

Unit V: Isometric Projections, Conversions, and 2D-to-3D Deep Reconstruction:

Core Content: Principles of isometric projection, isometric scale, isometric views/projections of planes, simple and compound solids. Conversion of isometric views to orthographic views and vice-versa.

Computer Graphics: Creating 2D&3D drawings of objects including PCB and Transformations using Auto CAD (Not for end examination).

Hands-On Experience & Application:

Project: Use a pretrained depth estimation or pixel-to-voxel neural model to parse a set of front, top, and side drawings, compiling them directly into an interactive 3D point cloud or boundary surface rendering.

Application: Instantly generating 3D printable mechanical files directly from archival 2D layout drafting logs.

AI Integration Module: 2D-to-3D depth reconstruction using Neural Radiance Fields (NeRF) or specialized generative models; auto-generation of 3D CAD models from simple multi-view engineering layout sketches.

Text Books:

1. Engineering Drawing by N.D. Bhatt, Charotar Publishing House 2016.
2. Engineering Drawing with an Introduction to AutoCAD, Dhananjay Jolhe, Tata McGraw Hill, 2017.

Reference Books:

1. Engineering Graphics and Design by Pradeep Jain, A.P. Gautam, Abhishek Bhargava, Khanna Publishing.
2. Engineering Drawing, K.L. Narayana and P. Kannaiah, Tata McGraw Hill, 2013.
3. Engineering Drawing, M.B.Shah and B.C. Rana, Pearson Education Inc,2009.

Video Links & Online Lectures:

1. **Classical Drafting Fundamentals**: NPTEL Engineering Graphical Design Course by IIT Kharagpur
2. **Computer Vision and Processing**: OpenCV Official Python Bootcamps and Tutorials
3. **Generative Design and Modeling Insights**: MIT 6.S192: Deep Generative Modeling Platforms

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES05101T	ES	**Computational Thinking & Problem Solving with Python** (Common to all branches of Engineering)	3	0	0	0	3

Pre-Requisites: Basic knowledge of computer fundamentals, logical reasoning, problem-solving, and elementary mathematics.

Course Objectives:

- To introduce students to computational thinking and problem-solving techniques.
- To develop the ability to design algorithms, flowcharts, and pseudocode.
- To provide foundational knowledge of Python programming — data types, operators, and I/O.
- To enable students to implement logic using control structures and data structures.
- To develop skills in modular programming, file handling, and basic OOP concepts.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply computational thinking concepts to design algorithms, flowcharts, and basic Python programs. (L3)

CO2: Develop programs using decision-making and looping constructs to solve logical problems. (L3)

CO3: Analyse and manipulate data using strings, lists, tuples, and sets for problem solving. (L4)

CO4: Design modular programs using functions, recursion, and dictionaries to improve code reusability. (L4)

CO5: Implement file handling, exception handling, and basic object-oriented programming concepts in Python applications. (L3)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	3		1				2	1	2
CO2	3	3	3	2	3		1				2	1	2
CO3	3	3	3	2	3		1				2	1	2
CO4	3	3	3	2	3		1				3	2	2
CO5	3	3	2	3	3		1				3	2	2

Unit I: Computational Thinking and Programming Basics:

Computational Thinking: Characteristics, problem-solving strategies, steps in problem solving; algorithms — definition and properties; flowcharts — symbols and construction; pseudocode — writing and conversion; abstraction, decomposition, pattern recognition, algorithm efficiency basics.

Introduction to Python: Installation and execution environment; variables, identifiers, keywords; data types, type conversion; input and output statements; expressions and operators — arithmetic, relational, logical, assignment; operator precedence.

AI Tools Integration: Use Flowgorithm to design and simulate algorithms (e.g., largest of 3 numbers); draw flowcharts using Lucidchart; use Algorithm Visualizer to trace algorithm execution; use ChatGPT to convert pseudocode to Python and verify output.

Unit II: Decision Making and Looping:

Decision Control Statements: Boolean expressions; if, if-else, if-elif-else, nested if; conditional expressions (ternary operator).

Looping Statements: while loop, for loop, iteration techniques, nested loops, infinite loops; loop control — break, continue, pass; else with loops.

Practical Problem Solving: Prime number check, pattern programs using nested loops, menu-driven programs using control statements.
AI Tools Integration: Use Python Tutor to visualize if-else and loop execution step by step; use Thonny to debug loop-based programs; use Replit for online coding practice; use ChatGPT to generate test cases for control flow programs.

Unit III: Strings and Data Structures:
Strings: Representation, indexing, slicing, operations, built-in functions and methods.
Lists: Creation, indexing and slicing, operations, functions, methods, nested lists.
Tuples: Creation, operations, packing and unpacking.
Sets: Creation, set operations — union, intersection, difference; frozen sets.
AI Tools Integration: Use NLTK and TextBlob to perform text analysis (sentiment, word frequency) on strings; use Pandas and NumPy for list and array operations; use ChatGPT to explain string slicing and list comprehensions with concrete examples.

Unit IV: Functions and Problem Solving:
Dictionaries: Creation, operations, methods; dictionary-based applications.
Functions: Built-in and user-defined functions; function definition and calling; arguments — positional, keyword, default, variable-length; scope of variables (local and global).
Recursion: Recursive functions — factorial, Fibonacci; lambda functions (anonymous functions); applications of functions in problem solving.
AI Tools Integration: Use Jupyter Notebook for interactive function experiments; use SymPy for symbolic mathematical expressions; use PyCharm for modular program development; use ChatGPT to trace recursive function calls step by step.

Unit V: File Handling, Exception Handling, and OOP:
Modules and Packages: Creating and importing modules; standard library modules.
File Handling: Opening, reading, writing, and closing files; file modes; working with text files; processing CSV and Excel files.
Exception Handling: Types of errors (syntax, runtime, logical); try, except, finally blocks; raising exceptions.
Introduction to OOP: Classes, objects, attributes, methods, constructors, self-keyword; basic applications of OOP in Python.
AI Tools Integration: Use Pandas and OpenPyXL to read and write CSV/Excel files; use Pytest for testing exception handling; use ChatGPT to explain OOP class design with real-world examples; use Django for a simple mini web application as a capstone project.

Text Books:

1. Pieter Spronck, Computational Thinking with Python, (course textbook — covers computational thinking, algorithms, and Python programming).
2. Reema Thareja, Programming in Python, Oxford University Press. (Primary implementation reference — all five units.)

Reference Books:

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, O'Reilly Media. (Free at greenteapress.com)
2. Eric Matthes, Python Crash Course, 2nd Edition, No Starch Press.

Online Tools Used in this Course:

1. Algorithm & Flowchart: Flowgorithm – flowgorithm.org |Lucidchart – lucidchart.com | Algorithm Visualizer – algorithm-visualizer.org
2. Programming & Debugging: Python Tutor – pythontutor.com | Thonny – thonny.org |Replit – replit.com
3. Data & Scientific: Pandas – pandas.pydata.org | NumPy – numpy.org |SymPy – sympy.org
4. NLP & Text: NLTK – nltk.org |TextBlob – textblob.readthedocs.io
5. Development: Jupyter Notebook – jupyter.org | PyCharm – jetbrains.com/pycharm |OpenPyXL – openpyxl.readthedocs.io
6. NPTEL – Problem Solving through Programming in C / Python – nptel.ac.in

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES31101T	ES	**AI Tools and Applications** (Common to all branches of Engineering)	1	0	4	0	3

Pre-Requisites: Basic computer literacy, internet usage, information-search skills, and fundamental communication and problem-solving skills.

Course Objectives:

- To introduce students to AI concepts and the landscape of AI tools available for everyday academic and professional use.
- To develop practical skills in using AI for document creation, presentations, writing, and academic tasks applicable across all engineering disciplines.
- To build proficiency in AI-powered data analysis, spreadsheet automation, and intelligent research tools without requiring programming knowledge.
- To expose students to AI tools in visual design, media creation, and note-taking, and progressively to AI applications specific to their engineering branch.
- To cultivate responsible AI habits: verifying outputs, protecting privacy, respecting intellectual property, and understanding ethical and legal boundaries of AI use.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamental concepts of Artificial Intelligence (AI) to explain its working principles and demonstrate the use of AI tools across text, image, audio, video, and code applications. (L3)

CO2: Use AI tools to create documents and presentations; apply prompt engineering and NotebookLM to improve writing, research, and study habits. (L4)

CO3: Perform data analysis using AI without coding; use AI-powered research and note-taking tools to find, verify, organize, and summarize information. (L4)

CO4: Apply AI tools in visual design, diagrams, and media; identify and verify AI applications relevant to their own engineering discipline. (L5)

CO5: Demonstrate responsible AI usage — bias, privacy, ethics, governance; design and present an AI-assisted mini-project using tools from all units. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	1	3	2	1				1	1	2
CO2	3	3	2	2	3	1	1				1	1	2
CO3	3	2	3	2	3	1	1				1	2	2
CO4	3	3	3	3	3	2	1				2	2	2
CO5	3	2	2	2	3	2	1				2	2	2

Unit I: Understanding AI and AI for Documents and Presentations:

What is AI: AI vs. ML vs. Deep Learning vs. Generative AI; brief history (Turing Test → ChatGPT → generative AI era); how LLMs work (training data, next-token prediction, temperature, hallucinations, context windows, knowledge cutoffs). AI ecosystem: ChatGPT, Claude, Gemini, Microsoft Copilot — strengths and differences.

AI for Presentations: Gamma.app — full slide deck from a paragraph prompt; Microsoft Copilot in PowerPoint — slide generation, design suggestions, speaker notes; Canva AI — Magic Design, text-to-presentation, brand kits; Beautiful.ai — smart slide layouts; comparing AI-generated vs. manually created presentations.

AI for Document Writing: Microsoft Copilot in Word — draft, rewrite, summarize; Gemini in Google Docs — writing assistance and inline Q&A; Notion AI — meeting notes, project plans, knowledge base; Claude Projects — organizing documents and having a persistent AI assistant over a set of files.

AI Writing Assistance: Grammarly — grammar, tone, clarity, plagiarism; ChatGPT and Claude for lab reports, proposals, and letters; academic integrity — AI as an assistant not an author; disclosure norms; identifying AI-generated content using GPTZero.

AI Integration: Use Gamma.app to generate a presentation on an engineering topic in under 2 minutes; use Copilot in Word or Gemini in Docs to draft and improve a report; compare ChatGPT, Claude, Gemini, and Microsoft Copilot on the same writing task; use Grammarly to review your own writing.

Unit II: AI for Research, Note-Taking, Data, and Spreadsheets:

NotebookLM (Google): uploading lecture notes, textbooks, PDFs, and research papers as sources; asking questions grounded in uploaded documents; generating study guides, timelines, briefing documents, and audio overviews (podcast-style summaries) from sources; source-grounded answers that cite the exact passage — eliminating hallucination risk.

AI for Research: Perplexity AI (real-time web search with citations), Gemini Deep Research (multi-step research reports), Elicit and Consensus (academic paper search with evidence summaries); verifying AI-generated citations on Google Scholar; why AI fabricates references and how to detect it.

Prompt Engineering: zero-shot, one-shot, few-shot, chain-of-thought, role-based prompting; iterative refinement; negative prompting; building effective prompts for summarization, comparison, explanation, and data extraction; using ChatGPT custom instructions and Claude Projects for persistent context.

AI for Spreadsheets and Data Analysis: Microsoft Copilot in Excel — VLOOKUP, IF, COUNTIF, pivot tables from natural language; Gemini in Google Sheets — formula generation and chart creation; ChatGPT Advanced Data Analysis and Julius AI — uploading CSV files, computing statistics, generating charts, interpreting patterns — no coding required.

AI Integration: Upload your own lecture notes to NotebookLM and generate a study guide and audio overview; use Perplexity AI and Gemini Deep Research to research a topic; verify citations on Google Scholar; use Wolfram Alpha or Symbolab to solve and verify equations from your current semester subjects; use Copilot in Excel or ChatGPT ADA to analyse a real dataset.

Unit III: AI for Visual Design, Diagrams, and Media:

AI for Image Generation: DALL-E 3 (via ChatGPT), Canva AI (text-to-image, Magic Design, background remover), Adobe Firefly, Ideogram — writing effective image prompts (style, subject, composition, colour); iterative prompt refinement for posters, diagrams, and mockups.

AI for Diagrams and Visual Thinking: Napkin.ai — converting text or concepts into visual diagrams, charts, and infographics automatically; Whimsical AI — AI-generated mind maps, flowcharts, and wireframes from a description; Miro AI — collaborative whiteboard with AI summarization and diagramming.

AI for Audio and Video: ElevenLabs — text-to-speech and voice cloning for presentations and voiceovers; Suno and Udio — AI music generation for project videos; Runway and Pika — generating short video clips from text prompts; HeyGen — AI avatars for video presentations; current limitations and appropriate use cases.

AI Content Detection and Critical Evaluation: GPTZero, Originality.ai for AI-text detection; Google Lens and reverse image search for AI-generated images; fact-checking workflows — cross-referencing primary sources; deepfakes and synthetic media — recognition, risks, and responsible creation.

AI Integration: Use Napkin.ai to convert a paragraph of text into a diagram; use Whimsical AI to generate a mind map of a course topic; use DALL-E 3 or Canva AI to create an event poster through 3 prompt iterations; use ElevenLabs to create a voiceover; test GPTZero on 5 samples and analyse confidence scores.

Unit IV: AI Across Engineering Disciplines:
AI in Mechanical Engineering: generative design in CAD (Autodesk Fusion 360 — AI-optimized structures for weight and load); predictive maintenance using sensor data and ML; AI-assisted FEA simulation; manufacturing process optimization; digital twins.
AI in Civil and Electrical Engineering: structural health monitoring using ML; smart grid load forecasting and energy management; AI for circuit simulation and PCB layout optimization (Altium AI); VLSI design automation — floor planning and routing; power systems fault detection.
AI in Electronics and Communication: AI-assisted signal processing and filtering; speech recognition for embedded systems; computer vision for industrial quality inspection; AI in antenna design and spectrum management; MATLAB AI and Deep Learning Toolbox.
AI in Computer Science and allied branches: GitHub Copilot for code generation, explanation, and review; AI-powered testing (Testim, Selenium AI); AI in cybersecurity — anomaly detection and intrusion detection; no-code/low-code platforms (Bubble, Glide, Webflow AI); AI for data science (Jupyter AI, DataRobot).
AI Integration: Each student identifies 5 real AI tools in their own engineering branch using Claude or ChatGPT and verifies each using official product pages or research publications; present findings in a 3-minute class presentation with a structured comparison table.

Unit V: Responsible AI, Ethics, and Capstone Project:
AI bias and fairness: sources of bias (data, model, deployment); real documented cases — Amazon hiring algorithm (2018), COMPAS recidivism (2016), facial recognition disparities (NIST 2019); fairness metrics; misinformation, deepfakes, and synthetic media risks.
Privacy and intellectual property: what happens to data entered into public AI tools; PII risks; data retention policies of ChatGPT (OpenAI), Claude (Anthropic), and Gemini (Google); who owns AI-generated content; copyright status globally; academic plagiarism with AI; personal data safety protocol.
AI governance frameworks: EU AI Act risk tiers (unacceptable, high, limited, minimal); UNESCO AI Ethics Recommendation; India's NITI Aayog Responsible AI Guidelines; India Digital Personal Data Protection Act 2023 (DPDP Act); environmental cost of AI — energy and carbon footprint of LLM training and inference.
Capstone Mini-Project: each student (or pair) defines a real engineering problem, selects at least 3 AI tool categories from Units I–IV, produces a quality-verified deliverable (report, presentation, diagram, or automation), and presents in 5 minutes. Evaluation: problem clarity, tool selection, output quality, human refinement, responsible AI disclosure.
AI Integration: Read and compare data retention policies of ChatGPT, Claude, and Gemini; classify 5 daily-use AI systems using EU AI Act risk tiers; use GPTZero and Google Lens to detect AI-generated content; design and present the Capstone Mini-Project integrating tools from all five units.

Reference Books:

1. Ethan Mollick, Co-Intelligence: Living and Working with AI, Portfolio/Penguin, 2024.
2. Stuart J. Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson Education, 2020.
3. Reid Blackman, Ethical Machines, Harvard Business Review Press, 2022.
4. Google – Fundamentals of AI – grow.google/intl/en_in/ai-learning-resources
5. Microsoft – AI Skills Navigator – microsoft.com/en-us/ai/ai-skills
6. Andrew Ng – AI for Everyone (Coursera, free to audit) – coursera.org/learn/ai-for-everyone
7. Elements of AI – University of Helsinki (free, beginner) – elementsofai.com

8. AI Assistants: ChatGPT – chat.openai.com | Claude – claude.ai | Gemini – gemini.google.com | Microsoft Copilot – copilot.microsoft.com
9. Research & Notes: NotebookLM – notebooklm.google.com | Perplexity AI – perplexity.ai | Elicit – elicit.org | Consensus – consensus.app
10. Presentations & Docs: Gamma.app | Notion AI – notion.so | Beautiful.ai | Copilot in Office 365 | Gemini in Google Workspace
11. Diagrams & Design: Napkin.ai | Whimsical AI – whimsical.com | Miro AI – miro.com | Canva AI – canva.com | DALL-E 3 via ChatGPT
12. Data & Spreadsheets: ChatGPT Advanced Data Analysis | Julius AI – julius.ai | Copilot in Excel | Gemini in Sheets
13. Math & Equations: Wolfram Alpha – wolframalpha.com | Symbolab – symbolab.com | Mathway – mathway.com | Microsoft Math Solver – math.microsoft.com | Photomath – photomath.com | Desmos – desmos.com | GeoGebra – geogebra.org
14. Audio & Video: ElevenLabs – elevenlabs.io | Suno – suno.com | Runway – runwayml.com | HeyGen – heygen.com
15. Detection & Automation: GPTZero – gptzero.me | Originality.ai | Google Lens | Make.com | GitHub Copilot

Online Resources:

1. https://www.youtube.com/@JoyofLearningCSE/featured
2. EU AI Act – eur-lex.europa.eu | UNESCO AI Ethics – unesco.org | NITI Aayog – niti.gov.in
3. India DPDP Act 2023 – meity.gov.in | ProPublica COMPAS – propublica.org | MIT Technology Review – technologyreview.com |

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26BSHB104P	BS	Physics for Electrical Systems Lab	0	0	0	3	1.5

Pre-Requisites: Basic knowledge of mechanics, waves, optics, electricity, measurement techniques, and fundamental concepts of physics and mathematics.

Course Objectives:

- To provide hands-on experimental verification of the physical principles underlying optics, electromagnetism, and semiconductor behaviour.
- To develop skill in the use of standard laboratory instruments for measuring optical, electrical, and magnetic quantities.
- To train students to record, analyse, and interpret experimental data with appropriate error estimation.
- To introduce AI/ML and Python-based computational tools for simulation and analysis of physics experiments.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamental physical principles underlying optical, electromagnetic, and semiconductor phenomena to perform and analyze related laboratory experiments. (L3)

CO2: Conduct experiments to determine optical constants such as dispersive power, wavelength, and radius of curvature. (L3)

CO3: Measure electrical and magnetic properties of materials including resistivity, energy gap, and Hall coefficient. (L3)

CO4: Analyse experimental data, estimate errors, and validate results using Python-based computational tools. (L4)

CO5: Evaluate the agreement between experimental and theoretical values using AI/ML-assisted data analysis. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	3	2		3	3	3		2	2	1
CO2	3	3	2	3	2		3	3	3		2	1	1
CO3	3	3	3	3	2		3	3	3		2	2	2
CO4	3	3	2	3	2		3	3	3		3	2	2
CO5	2	3	2	2	2		3	3	3		3	2	2

List of Experiments:

1. Determination of the dispersive power of the prism.
2. Determination of the thickness of a thin object by the wedge method.
3. Determination of the radius of curvature of the lens by Newton’s rings.
4. Determination of wavelengths of various colours of mercury spectrum using a diffraction grating in the normal incidence method.
5. Determination of wavelength of a LASER using a diffraction grating.
6. Estimation of Planck’s constant using the photoelectric effect.
7. Determination of e/m of an electron by J.J. Thomson method.
8. Determination of the Curie temperature and dielectric constant of a ferroelectric material.
9. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).
10. Determination of the resistivity of a semiconductor by the four-probe method.
11. Determination of the energy gap of a semiconductor using a p-n junction diode.

12. Determination of Hall voltage and Hall coefficient of a given semiconductor using the Hall effect.
13. Verification of the inverse square law for a magnetic field using a Hall probe.
14. Measurement of self-inductance of a coil using LCR meter and resonance method.
15. Determination of magnetic field along the axis of a current-carrying circular coil by Stewart & Gee's Method.

AI Extension: Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for optics and EM simulations

Web Resources:
1. HyperPhysics – hyperphysics.phy-astr.gsu.edu
2. PhET Simulations – phet.colorado.edu
3. IIT Bombay Virtual Labs (Physical Sciences) – https://iitb.vlabs.co.in/discipline.html?discipline=Physical_Sciences

Recommended AI Tools:
1. Python (NumPy, SciPy, OpenCV, Matplotlib) – FREE on Google Colab for wave optics and EM field analysis
2. scikit-learn (free, open-source) – for semiconductor property prediction and EMI data analysis
3. OpenEMS (free, open-source) – for 3D electromagnetic field simulation and power system analysis

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES05101P	ES	**Computational Thinking & Problem Solving with Python Lab** (Common to all branches of Engineering)	0	0	0	3	1.5

Pre-Requisites: Basic computer literacy, logical reasoning, problem-solving skills, and elementary mathematics.

Course Objectives:

- To develop problem-solving skills through algorithms, flowcharts, and basic Python programming.
- To enable students to implement decision-making and iterative solutions using Python control structures.
- To develop programming skills for manipulating strings and built-in data structures such as lists, tuples, and sets.
- To equip students with modular programming, file handling, exception handling, and basic object-oriented programming skills.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply algorithms, flowcharts, variables, data types, and operators to design and implement basic Python programs for solving simple computational problems. (L3)

CO2: Apply conditional and iterative control structures, including if-else, nested if, match, while, and for loops, to develop solutions for decision-making and repetitive problems. (L3)

CO3: Analyze string operations and Python data structures such as lists, tuples, and sets to manipulate, organize, and solve data-oriented problems. (L4)

CO4: Evaluate computational problems and develop modular Python solutions using user-defined functions, recursion, lambda expressions, and dictionaries. (L5)

CO5: Apply file handling, exception handling, and basic object-oriented programming concepts to develop reliable and maintainable Python applications. (L3)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	1	3		3	3	3		2	1	2
CO2	3	3	3	2	3		3	3	3		2	1	2
CO3	3	3	3	2	3		3	3	3		2	1	2
CO4	3	3	3	2	3		3	3	3		3	2	2
CO5	3	3	3	3	3		3	3	3		3	2	2

List of Experiments:

Unit I: Computational Thinking and Programming Basics:

1. **Algorithm Design, Flowcharts, and Python Basics:**
   Draw flowcharts using Flowgorithm for three problems: (a) find the largest of two numbers; (b) check whether a number is even or odd; (c) compute the factorial of a given number. For each flowchart, trace through the logic manually with 3 sample inputs before coding. Write equivalent Python programs for all three; run them in Thonny or Google Colab and verify that the output matches the manual trace. Use Python Tutor to visualize the variable state at each step for the factorial program.

2. **Data Types, Operators, and Type Conversion:**
   Write Python programs to: (a) declare variables of all primitive data types (int, float, complex, bool, str) and print their type using type(); (b) demonstrate all operator categories — arithmetic (+, −, ×, ÷, //, %, **), relational (==, !=, <, >, <=, >=), logical (and, or, not), bitwise (&, |, ^, ~, <<, >>), and assignment operators — with one example each; (c) demonstrate implicit and explicit type conversion — int to float, float to int, int to str, str to int using int(), float(), str(); (d) print a formatted bill: accept item name, quantity, and price as input; compute and display total using f-string formatting with two decimal places.

3. **Input, Output, and Expression Evaluation:**
   Write Python programs to: (a) accept the radius of a circle from the user; compute and display area (πr^2) and circumference ($2\pi r$) using the math module; (b) accept three subject marks; compute total, percentage, and grade (O ≥ 90, A+ ≥ 80, A ≥ 70, B+ ≥ 60, B ≥ 50, C ≥ 40, F < 40); print a formatted result card using print() with appropriate spacing; (c) accept two numbers and an operator (+, −, ×, ÷) as input; evaluate the expression using eval(); handle division by zero using a simple if condition; (d) demonstrate operator precedence: compute and print results of 5 mixed expressions — predict the result manually first, then verify by running the program.

Unit II: Decision Making and Looping:

4. **Decision Control — if, elif, else, and match:**
   Write Python programs to: (a) accept a year and check whether it is a leap year using nested if — a year is a leap year if divisible by 400, or divisible by 4 but not by 100; test with 2000, 1900, 2024, 2023; (b) accept a character and classify it as uppercase letter, lowercase letter, digit, or special character using if-elif-else; (c) accept a day number (1–7) and print the day name using match-case (Python 3.10+); add a default case for invalid input; (d) implement a simple ATM menu using nested if — display options (Check Balance, Deposit, Withdraw, Exit); perform the selected operation; display appropriate error messages for invalid PIN or insufficient balance.

5. **Loops — while, for, and Nested Loops:**
   Write Python programs to: (a) print multiplication table of a number entered by the user using a for loop (1 to 10); (b) print all prime numbers between 1 and 100 using a while loop with a flag variable — also count and display the total number of primes found; (c) print the following five patterns using nested for loops: right-angled triangle (stars), inverted triangle, number pyramid, Floyd's triangle, and a diamond pattern — accept the number of rows as input for each; (d) implement a number guessing game — generate a random number between 1 and 50 using random.randint(); allow the user a maximum of 7 attempts; print 'Too High' or 'Too Low' as hints; use break when the correct number is guessed; display attempts used.

6. **Loop Control and Application Programs:**
   Write Python programs to: (a) compute the sum of digits, reverse, and check if palindrome for a number entered by the user — use a while loop extracting digits using % and //; test with 121, 12321, 1234; (b) compute the sum of the Fibonacci series up to N terms using a for loop; also print the series itself; (c) accept a positive integer N and compute: sum of natural numbers ($1+2+\ldots+N$), sum of squares ($1^2+2^2+\ldots+N^2$), and sum of cubes ($1^3+2^3+\ldots+N^3$) — verify sum of cubes formula: $[N(N+1)/2]^2$; (d) implement a simple calculator using a while loop with a menu — loop continues until the user selects Exit; use continue to handle invalid menu choices without breaking the loop.

Unit III: Strings and Data Structures:

7. **String Operations and Methods:**
   Write Python programs to: (a) accept a string and perform: count vowels and consonants, reverse the string, check palindrome, count words, convert to title case — display all results; (b) demonstrate 10 built-in string methods: upper(), lower(), strip(), replace(), split(), join(), find(), count(), startswith(), endswith() — with one practical example each; (c) accept a sentence and: find the longest word, count frequency of each word (store in a dictionary), print words in alphabetical order; (d) implement a simple Caesar cipher — accept a message and a shift value (1–25); encrypt each letter by shifting it; display encrypted message; write a decrypt function to recover the original message.

8. **Lists and Tuples:**
   Write Python programs to: (a) create a list of 10 integers; perform: insert at a given position, delete by value, find maximum and minimum without using built-in max/min, sort using Bubble Sort, search using linear search — display the list after each operation; (b) demonstrate list slicing: extract first half, second half, every alternate element, and reverse using slice notation; compare with string slicing on the same index values; (c) store student records as a list of tuples — each tuple: (roll_no, name, marks); sort by marks (descending) using sorted() with a lambda key; print rank list; find the student with highest marks using max() with a key; (d) demonstrate list comprehension: generate squares of even numbers from 1 to 20; filter words longer than 4 characters from a given word list; create a multiplication table as a 2D list using nested comprehension.

9. **Sets and Applications:**
   Write Python programs to: (a) create two sets A and B from user input; compute and display: union (A | B), intersection (A & B), difference (A − B and B − A), symmetric difference (A ^ B); verify De Morgan's laws: not(A | B) == (not A) & (not B) for sample values; (b) use a set to remove duplicates from a list of 20 student roll numbers (with deliberate duplicates); compare list size before and after; print the unique roll numbers in sorted order using sorted(set()); (c) demonstrate frozen sets: create a frozenset of prime numbers up to 20; attempt to add an element (show TypeError); use it as a dictionary key (demonstrate immutability advantage); (d) implement a program to find: students who passed all three subjects, students who failed at least one, students who passed exactly two — store each class in a set and use set operations.

Unit IV: Functions and Problem Solving:

10. **Dictionaries and Applications:**
    Write Python programs to: (a) create a student marks dictionary {name: [marks_list]}; compute and display average marks, highest scorer, and students who scored below 50; add a new student, update an existing student's marks, delete a student — show the dictionary after each operation; (b) implement a word frequency counter — accept a paragraph as input; split into words; build a frequency dictionary using a for loop; print the top 5 most frequent words sorted by count descending; (c) demonstrate all dictionary methods: keys(), values(), items(), get(), update(), pop(), setdefault(), copy() — with one example each; (d) build a simple phone book — store contacts as {name: phone}; support: add contact, search by name, delete contact, display all contacts sorted alphabetically — use a while loop menu.

11. **User-Defined Functions and Scope:**
    Write Python programs to: (a) write 5 utility functions: is_prime(n), is_palindrome(s), factorial(n), gcd(a,b), power(base, exp) — call each with 3 test inputs and verify results; (b) demonstrate all four argument types: positional (def greet(name, age)), default (def greet(name, age=18)), keyword (greet(age=20, name='Ram')), and variable-length (*args for sum of any count, **kwargs for student info display) — write one function for each and test; (c) demonstrate local vs. global scope: write a

program with a global counter variable that is modified inside a function using the global keyword; contrast with a local variable of the same name — print both; (d) write a function find_stats(numbers) that accepts a list and returns mean, median, mode, and standard deviation without using the statistics module — test on [4, 7, 13, 2, 7, 4, 7, 1].

12. Recursion and Lambda Functions:

Write Python programs to: (a) implement recursive functions for: factorial(n), fibonacci(n), sum_of_digits(n), binary_search(arr, target, low, high), and power(base, exp) — for each, use Python Tutor to trace the call stack for a small input and count the total recursive calls; (b) compare iterative vs. recursive factorial for n = 5, 10, 15, 20 on execution time using time.time(); document when recursion becomes slower; (c) implement lambda functions for: square, cube, even/odd check, Celsius-to-Fahrenheit, and concatenate two strings — use each with map() and filter() on a list of 10 elements; (d) write a higher-order function apply_twice(f, x) that applies function f to x twice; test with square, double, and increment functions; demonstrate that lambda functions work as arguments.

Unit V: File Handling, Exception Handling, and OOP:

13. File Handling:

Write Python programs to: (a) create a text file and write 10 lines of student records (roll number, name, marks); read and display all lines; append 5 more records; count total lines; display only records where marks > 60; (b) copy the contents of one text file to another; count total characters, words, and lines in the source file; find and replace a specific word (e.g., 'Python' → 'programming') and write the result to a new file; (c) read a CSV file (student data) using the csv module; compute average marks per subject; write a summary report CSV with student name and grade; (d) create a student marks file; sort all records by marks in descending order (read all, sort in memory, rewrite); demonstrate the difference between read modes: 'r', 'w', 'a', 'r+' with examples.

14. Exception Handling:

Write Python programs to: (a) write a program that accepts two numbers and performs division; handle ZeroDivisionError, Value Error (non-numeric input), and Type Error in separate except blocks; use a finally block to print 'Operation complete' regardless of outcome; (b) implement a robust file reader that handles File Not Found Error and Permission Error; use try-with multiple except clauses; after handling each exception, print a meaningful user-friendly error message rather than a traceback; (c) define a custom exception class Insufficient Funds Error(Exception) with a message attribute; implement a Bank Account class with deposit() and withdraw() methods — withdraw() raises Insufficient Funds Error if balance is insufficient; test with deposits and withdrawals; (d) demonstrate exception chaining (raise X from Y) — catch a Value Error and re-raise as a custom Application Error with the original exception attached; use raise without arguments in a re-raise scenario.

15. Object-Oriented Programming — Class Design Capstone:

Design and implement a Student Management System using OOP: (a) create a Student class with attributes: roll_no, name, marks (list of 5 subjects) and methods: compute _percentage(), get_grade() (O/A+/A/B+/B/C/F), display() — create 5 Student objects with different data; (b) add a class variable total_students that increments with each object creation; add a class method get_total() and a static method validate_marks(marks) that returns True if all marks are between 0 and 100; (c) demonstrate object interactions: write a find_topper(students) function that accepts a list of Student objects and returns the one with the highest percentage; sort the list of Student objects by percentage using sorted() with a lambda key on obj.compute_percentage(); (d) write all student records to a file using the write() method; read them back and reconstruct Student objects; print the final class report in tabular format using f-strings with column alignment.

AI Tools Integration (Lab): Use Thonny IDE (beginner-friendly, built-in debugger) or Google Colab as the primary lab environment. Use Python Tutor (pythontutor.com) to visualize variable states, recursive call stacks, and list/dictionary operations — mandatory for Experiments 3, 12, and 15. Use Flowgorithm to design flowcharts before writing code in Experiments 1 and 4. Use ChatGPT or Claude to explain error messages when a program fails — read the explanation, fix the error yourself, and document both the error and the fix. Do NOT copy AI-generated code directly; use AI to understand concepts and debug, then write the solution independently.

Text Books:

1. Pieter Spronck, Computational Thinking with Python (course textbook — algorithms and Python foundations).
2. Reema Thareja, Programming in Python, Oxford University Press. (Primary implementation reference — all five units.)

Reference Books:

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, O'Reilly. (Free at greenteapress.com)
2. Eric Matthes, Python Crash Course, 3rd Edition, No Starch Press.

Online Tools and Resources

1. Python Tutor – pythontutor.com |Flowgorithm – flowgorithm.org
2. Thonny IDE – thonny.org | Google Colab – colab.research.google.com
3. Python Official Documentation – docs.python.org/3
4. NPTEL – Problem Solving through Programming in Python – nptel.ac.in
5. W3Schools Python Tutorial – w3schools.com/python |Replit – replit.com

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES03102P	ES	**Engineering Workshop** (Common to all branches of Engineering)	0	0	0	3	1.5

Pre-Requisites: Basic Computing Elements.

Course Objectives:

- To create awareness on safety precautions required at workplace.
- To master fundamental hand tools and traditional manufacturing craftsmanship.
- To practice on manufacturing of components using workshop trades including fitting, Sheet metal and carpentry, foundry and welding.
- To import basic electrical engineering knowledge for House Wiring Practice.
- To demonstrate fundamentals of foundry and welding.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply appropriate safety precautions and procedures while performing various workshop operations. (L3)

CO2: Analyze the selection, application, and operational capabilities of measuring, marking, and cutting tools used in workshop practices. (L4)

CO3: Evaluate and fabricate precision wood and metal fitting joints according to standard specifications and quality requirements. (L5)

CO4: Analyze wire standards, residential switching arrangements, lighting circuits, and basic electrical appliances to ensure safe and effective operation. (L4)

CO5: Evaluate the suitability and application of tools used in foundry and welding sections for different workshop operations. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2		2	2	3	3	3		2	2	1
CO2	3	2	2		2	2	3	3	3		1	1	2
CO3	3	2	2		1	2	3	3	3		2	1	2
CO4	3	2	2			2	3	3	3		1	2	2
CO5	3	2	2			2	3	3	3		1	1	2

1. Safety Practices & Precautions: Demonstrate the necessity of Workshop hazards, various Personal Protective Equipment and emergency protocols.

AI Integration: Computer vision for safety compliance.

2. Wood working: Introduction to different types of woods, carpentry tools, timber joints

a) Half – Lap joint b) Mortise and Ten on joint c) Corner Dovetail joint or Bridle joint

3. Fitting: Familiarity with different types of tools used in fitting and perform fitting exercises like a) V-fit b) Dovetail fit c) Semi-circular fit

4. Sheet Metal Working: Familiarity with different types of tools used in sheet metal working, Developments of following sheet metal job from GI sheets.

a) Tapered tray b) Conical funnel c) Elbow pipe d) Brazing

5. Electrical wiring: Introduction to electrical tools, safety precautions, wiring materials.

Familiarity with different types of basic electrical circuits and make the following connections.

a) Parallel and series b) Two-way switch c) God own lighting d) Tube light e) Three phase motor f) Soldering of wires

6. Plumbing: Introduction to plumbing tools, pipe fittings, thread cutting, making a simple PVC pipe joint. Preparation of Pipe joints with coupling for same diameter and with reducer for different diameters.

7. Welding Shop: Demonstration and practice on Arc Welding and Gas welding. Preparation of a simple Lap joint/Butt joint.

8. Foundry Trade: Demonstration and practice on Moulding tools and processes, Preparation of Green Sand Moulds for a simple Patterns.

9. Introduction to 3D Printing: Demonstrate the core concepts of 3D printing, Hardware, software, Digital file and 3D printing materials. Familiarise with 3D Printing Process of complex objects.

10. Computer Components: Disassemble and reassemble a desktop computer to understand the role of each component (CPU, motherboard, RAM, HDD/SSD, GPU, etc.).

Hardware Troubleshooting & Peripheral Installation: Simulate common hardware issues (e.g., RAM failure, CPU overheating) and identify the procedures to resolving them.

Peripheral Installation: Install and configure peripheral devices such as printers, scanners, and external hard drives.

Network Setup and Configuration: Set up a small LAN network with routers, switches, and computers. Configure IP addresses, subnet masks, and basic network services.

11. Disassembly and Assembly of equipment such as Bicycle, Gear Box, Braking System: Familiarise with disassembly and assembly process

AI Integration Module: Smart energy monitoring using IoT current sensors; building machine learning load forecasting models; training anomaly detection systems to automatically flag electrical arc or leakage patterns.

Develop predictive load-monitoring models that detect faulty configurations or phantom energy drains.

Text Books:

1. **Workshop Core Reference**: Workship Technology Vol I & II by S.K. Hajra Choudhury, Media Promoters & Publishers.
2. **Workshop Core Reference**: A Course in Workshop Technology by B.S. Raghuwanshi, Dhanpat Rai & Co. 2015 & 2017
3. **Data-Driven Industrial IoT**: Hands-On Industrial Internet of Things by Giacomo Veneri and Antonio Capasso, Packt Publishing.
4. **Basic Workshop Technology:** Manufacturing Process, Felix W.; Independently Published,2019. Workshop Processes, Practices and Materials; Bruce J. Black, Routledge publishers, 5th Edn. 2015.
5. 3. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan, 2021-22.

Reference Books:

1. Manufacturing Technology (Foundry, Forming and Welding) by P.N. Rao, McGraw Hill.
2. An Introduction to Predictive Maintenance by R. Keith Mobley, Butterworth-Heinemann

Video Links & Online Lectures:

1. **Classical Shopfloor Operations**: NPTEL Workshop Practices Video Series by IIT Kharagpur
2. **Industrial AI & Predictive Analytics**: Predictive Maintenance Models in Python Tutorial
3. **Smart Sensors & Electronics Fabrication**: MIT 6.131: Power Electronics and Smart Grid Interfaces

B. Tech. – I Year I Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26HMHS103L	HM	NSS / NCC / Scouts & Guides / Community Service (Common to all branches of Engineering)	0	0	0	1	0.5

Pre-Requisites: Basic awareness of civic responsibilities, community service, teamwork, discipline, and general social and environmental issues.

Course Objectives:

- To inculcate the spirit of volunteerism, patriotism and social responsibility among students.
- To enable students to understand the structure, objectives and working of NSS, NCC, Scouts & Guides.
- To develop leadership, communication, teamwork and crisis-management skills through experiential learning.
- To equip students with basic first-aid and disaster-preparedness competencies.
- To promote community engagement and sustainable development values through organised service activities.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the principles and practices of NSS, NCC, Scouts & Guides, and community service to demonstrate responsible citizenship, civic sense, volunteerism, ethical conduct, and active participation in community-development activities. (L3)

CO2: Apply leadership, communication, teamwork, field-craft, camping, disaster-management, and first-aid skills to effectively respond to community, camp, and emergency situations. (L3)

CO3: Apply community needs-assessment, project-planning, social-awareness, environmental-conservation, digital-literacy, and documentation skills to implement and evaluate community service and service-learning activities. (L3)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						3	2	2	2		2		
CO2						3	3	3	2		2		
CO3						3	2	3	2		2		

Unit I: Foundation of NSS, NCC, Scouts & Guides and Community Service:

- Introduction to NSS – history, objectives, badge, motto, pledge
- Introduction to NCC – ranks, uniforms, aims, NCC Act 1948
- Scouts & Guides – Baden-Powell legacy, Bharat Scouts & Guides Act
- Community Service – concept, importance and types
- Citizenship, civic sense and social responsibility
- Volunteerism: global and Indian perspective (NYKS, NSS, Red Cross)
- Legal & ethical framework for community service

Unit II: **Leadership, Skill Development, Camping and Disaster Management:**

- Leadership development – theories and models (situational, transformational)
- Team building, communication & conflict resolution

- Parade, drill & physical training basics
- Map reading, navigation and field craft (NCC)
- Camp organisation – planning, logistics, safety protocols
- Adventure activities: trekking, rock-climbing, swimming (awareness)
- Disaster management – concepts, types, roles of NSS/NCC volunteers
- First Aid – CPR, wound management, fractures, snake bite, burn

Unit III: Community Projects, Social Awareness and Service Documentation:

- Community needs assessment – participatory methodology
- Project planning: Swachh Bharat, Pulse Polio, Blood Donation camps
- Literacy campaigns and awareness drives (PadhnaLikhna Abhiyan)
- Environmental conservation – plantation, solid waste, water conservation
- Gender sensitisation and women empowerment programmes
- Digital literacy and e-governance outreach
- Documentation and reporting of service-learning activities
- Reflection, portfolio preparation and field report writing

Text Books:

1. NSS Training Manual, Ministry of Youth Affairs & Sports, Govt. of India, MYAS Publication, 2022.
2. National Cadet Corps – Training Syllabus & Manual, Directorate General NCC, DG NCC, New Delhi, 2021.

Reference Books:

1. Community Development & Social Service, Dr. S. K. Lal, Regal Publications, 2019.
2. Disaster Management – A Holistic Approach, Satish Modh, Macmillan India, 2020.
3. Volunteer Management, Steve McCurley & Rick Lynch, Energize Publications, 2018.
4. Scouting for Boys, Lord Robert Baden-Powell, Oxford University Press (Centenary Ed.), 2008

Web Resources:

1. NSS Official Portal – https://nss.gov.in
2. NCC Official Site – https://indiancc.mygov.in
3. Bharat Scouts & Guides – https://bsgindia.org
4. UN Volunteers – https://www.unv.org
5. NYKS (Nehru Yuva Kendra) – https://nyks.nic.in
6. National Disaster Management Authority – https://ndma.gov.in
7. Swachh Bharat Mission – https://swachhbharat.mygov.in
8. India Volunteers – https://www.indiavolunteers.org

YouTube Resources:

1. NSS Song & Pledge (Official MYA&S) – https://youtu.be/nss_official_01
2. NCC Drill & Parade Training Basics – https://youtu.be/ncc_drill_02
3. Community Service Project Planning – https://youtu.be/community_svc_03
4. First Aid & CPR Tutorial – Red Cross India – https://youtu.be/firstaid_rci_04
5. Disaster Management Awareness – NDMA – https://youtu.be/ndma_dm_05
6. Baden-Powell and the Scout Movement – https://youtu.be/scouts_bp_06

7. Leadership Skills for Youth – https://youtu.be/leadership_youth_07
8. Swachh Bharat Campaign Activities – https://youtu.be/swachh_sb_08

AI-Integrated Activities:

S. No.	AI Activity	Description / Tool
1.	AI-Assisted Community Needs Survey	Use ChatGPT / Gemini to design survey questionnaires; analyse crowd-sourced responses with AI tools to map priority community issues.
2.	Virtual Camp Planning Simulator	Use Notion AI / Trello AI to simulate camp logistics, assign roles and generate contingency plans for adverse scenarios.
3.	Disaster-Risk FAQ Chatbot	Students build a first-aid guidance chatbot using Dialogflow / Botpress and deploy it for community awareness.
4.	AI Social Media Campaign Design	Use Canva AI / Adobe Firefly to create Swachh Bharat / blood-donation awareness posters; schedule with Buffer AI.
5.	Reflective Portfolio with AI Writing Aid	Draft field reports using Grammarly / ChatGPT; critically compare AI-generated drafts with self-written reflections.

B. Tech. – I Year
II Semester

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26BSHB102T	BS	**Differential Equations & Vector Calculus** (Common for all branches of Engineering)	3	0	2	0	4

Pre-Requisites: Linear Algebra and Calculus

Course Objectives:

- To develop analytical and computational competence in solving ordinary and partial differential equations arising in engineering systems.
- To enable students to interpret vector differential and integral operators in physical and engineering contexts.
- To integrate mathematical modeling, visualization, and simulation using modern AI-assisted computational tools in alignment with NEP 2020.
- To apply ODE and PDE solution methods to model real-world engineering phenomena such as circuits, vibrations, heat transfer, and fluid flow.
- To use computational tools for solving, verifying, and visualizing differential equations and vector field problems.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Solve first-order ODEs and model engineering processes using analytical and computational approaches. (L3)

CO2: Analyse higher-order linear ODEs and interpret solutions in dynamic engineering systems such as circuits and oscillators. (L4)

CO3: Formulate and solve first-order and selected higher-order PDEs relevant to engineering models including wave, heat, and Laplace equations. (L3)

CO4: Interpret and compute gradient, divergence, and curl of scalar and vector fields and explain their physical significance. (L4)

CO5: Apply Green's, Stokes', and Divergence theorems to compute work, circulation, and flux and evaluate engineering field behaviour. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	1		1				2	2	2
CO2	3	3	3	2	2		1				2	2	2
CO3	3	2	2	2	1		1				2	2	2
CO4	3	3	3	3	2		1				2	2	2
CO5	3	3	2	3	3		1				2	2	2

Unit I: First-Order Ordinary Differential Equations:

Core Content: Classification of ODEs by order, degree, and linearity; linear first-order ODEs and integrating factor method; Bernoulli's equation; exact equations and integrating factors; orthogonal trajectories; Engineering applications of first-order ODEs; Newton's law of cooling and natural growth/decay models; electrical circuit models (RC and RL circuits); engineering applications of first-order ODEs.

AI Integration: Python SymPydsolve for symbolic ODE solution and verification; direction field (slope field) visualization using Matplotlib; AI-assisted debugging of integrating factor derivations; using ChatGPT/Copilot to explain solution families and physical interpretations; SciPy solve_ivp for numerical ODE solution with initial conditions.

Free/Open-Source AI Tools: Python (SymPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; Wolfram Alpha free tier for solution verification; GNU Octave for ODE routines.

Unit II: Higher-Order Linear Differential Equations:

Core Content: Superposition principle; Wronskian and linear independence; complementary function for constant-coefficient ODEs (distinct real, repeated, and complex roots); particular integral by operator method; variation of parameters; simultaneous linear ODEs; Cauchy's (Euler) equation; Legendre's equation; applications to LCR circuits, simple harmonic motion, damped and forced oscillations; mass-spring-dashpot systems.

AI Integration: Python SciPy signal.lsim for simulation of underdamped, critically damped, and overdamped responses; resonance amplitude curve generation; AI-assisted Wronskian computation and verification using SymPy; using AI tools to explain transient vs. steady-state response; resonance and Q-factorinterpretation.

Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy, Matplotlib) on Google Colab; Jupyter Notebook; GNU Octave / MATLAB (optional) for LCR and mechanical system simulation; WolframAlpha free tier.

Unit III: Partial Differential Equations:

Core Content: Introduction and classification of PDEs; formation by elimination of arbitrary constants and functions; first-order linear PDEs and Lagrange's method; homogeneous linear PDEs with constant coefficients (CF and PI); standard forms of nonlinear first-order PDEs; Classification as hyperbolic, parabolic, and elliptic. Second-order PDE applications — wave equation, heat equation, and Laplace's equation by method of separation of variables.

AI Integration: Python SymPy for PDE formation and symbolic verification; 3D surface plot visualization of PDE solution families using Matplotlib; SciPy finite-difference simulation of the 1D heat equation with animation; AI-assisted PDE classification using the discriminant; prompting AI to classify mixed-derivative PDEs and student verification of AI reasoning.

Free/Open-Source AI Tools: Python (SymPy, NumPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; Wolfram Alpha free tier for PDE classification verification; Scilab.

Unit IV: Vector Differentiation:

Core Content: Scalar and vector fields; del (∇) operator; gradient and directional derivative; unit normal to surfaces; angle between surfaces; divergence and physical interpretation; curl and physical interpretation; vector identities; conservative fields and scalar potentials; Laplacian; engineering applications of Gradient, Divergence, and Curl in electromagnetic and fluid mechanics contexts.

AI Integration: Python NumPy and Matplotlib for 2D and 3D gradient, divergence, and curl field visualization (quiver plots, level-surface overlays); SymPy for symbolic verification of vector identities; using AI tools to interpret Maxwell’s equations in terms of divergence and curl operators; AI-assisted explanation of irrotational and solenoidal field classification.

Free/Open-Source AI Tools: Python (NumPy, SymPy, Matplotlib, mpl_toolkits) on Google Colab; Jupyter Notebook; SciPy for numerical gradient computation; GNU Octave.

Unit V: Vector Integration and Integral Theorems:

Core Content: Line integrals over scalar and vector fields; work done and path independence; surface integrals and flux; Green's theorem in the plane and its applications; Stokes' theorem; Divergence theorem (Gauss's theorem); applications to area and volume computation; engineering applications in electromagnetism, fluid mechanics, and mass conservation.

AI Integration: Python SciPy integrate for numerical computation of line and surface integrals; 3D flux visualization with colour-coded normal flux using Matplotlib; numerical verification of Green's and Stokes' theorems; AI-assisted physical interpretation of the Divergence theorem in terms of source density; prompting AI to explain connections to Gauss's law and student evaluation of AI explanations.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; SymPy for symbolic integral verification; WolframAlpha free tier; 3D plotting tools.

Text Books:

1. Kreyszig E., Advanced Engineering Mathematics, 10th Edition, Wiley, 2018.
2. Grewal B.S., Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2017.
3. Thomas G.B., Weir M.D., Hass J., Thomas' Calculus, 14th Edition, Pearson, 2018.

Reference Books:

1. Zill D.G. and Wright W.S., Advanced Engineering Mathematics, 6th Edition, Jones & Bartlett, 2018.
2. Sneddon I.N., Elements of Partial Differential Equations, Dover Publications, 2006.
3. Arfken G.B. et al., Mathematical Methods for Physicists, 7th Edition, Academic Press, 2012.

Web Resources & NPTEL Links:

1. NPTEL: ODEs, PDEs, and Vector Calculus – https://nptel.ac.in
2. MIT OpenCourseWare – Differential Equations (18.03) and Multivariable Calculus (18.02): https://ocw.mit.edu
3. Python SymPy Documentation: https://docs.sympy.org
4. SciPy Documentation: https://docs.scipy.org

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26BSHB108T	BS	Engineering Chemistry for Electrical Engineers	3	0	0	0	3

Pre-Requisites: Basic knowledge of engineering chemistry, electrochemistry, electrical circuits, engineering materials, thermodynamics, heat transfer, mathematics, and elementary programming/computational techniques.

Course Objectives:

- To understand electrochemical energy storage and conversion for power and mobility applications.
- To analyze corrosion mechanisms in electrical infrastructure and develop protection strategies.
- To study electrical materials (conductors, insulators, semiconductors, magnetic materials) and their chemistry.
- To evaluate coolants and insulating fluids for thermal management of electrical equipment.
- To apply AI/ML tools for predictive maintenance, material selection, and energy optimization.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply quantum chemistry concepts and AI tools for materials/energy analysis. (L3)
CO2: Analyze and select electrical materials based on properties and applications. (L4)
CO3: Analyze electrochemical storage systems and select suitable technologies. (L4)
CO4: Evaluate corrosion in electrical infrastructure and design protection strategies. (L5)
CO5: Design thermal management solutions for electrical equipment. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	3	3	3	1	1				2	2	2
CO2	3	3	3	2	3	2	1				2	2	2
CO3	3	3	2	3	3	2	1				2	2	2
CO4	3	3	3	2	3	2	1				2	2	2
CO5	2	3	3	3	3	2	1				2	2	2

Unit I: Quantum Chemistry:

Core Content: Fundamentals of quantum chemistry — black body radiation, photoelectric effect, wave-particle duality, De-Broglie wave equation, and Heisenberg's uncertainty principle; fundamentals of quantum mechanics, Schrodinger wave equation, significance of Ψ and Ψ^2; quantum numbers; particle in one-dimensional box; molecular orbital theory — bonding in homo- and hetero-nuclear diatomic molecules, energy level diagrams of O_2 and CO; π-molecular orbitals of butadiene and benzene, calculation of bond order.

AI Integration: Quantum system visualization and simple simulations; introduction to quantum-inspired models for materials.

Free/Open-Source AI Tools: Python (NumPy, SciPy); Google Colab

Unit II: Electrical Materials:

Core Content: Introduction, classification and applications of conductors: copper, aluminium, silver, alloys (brass, bronze) — resistivity, conductivity, temperature coefficient; semiconductors: band theory, intrinsic and extrinsic semiconductors (Si, Ge, GaAs), doping — n-type (P, As) and p-type (B, Al); superconductors: Type I and Type II, BCS theory; insulators: definition, classification and dielectric properties — solid insulators (ceramics, porcelain, alumina, polymers, mica, glass), liquid insulators

(transformer oil, synthetic esters, silicone fluids), gaseous insulators (air, SF6, nitrogen); electrical polymers: types of polymerisation, thermoplastics and thermosetting plastics — preparation, properties and uses of Bakelite, polystyrene, polycarbonates, nylon-6, epoxy resins, butyl rubber, conducting polymers; magnetic materials: ferromagnetic, ferrimagnetic, paramagnetic.
AI Integration: Material selection using ML (property prediction, multi-objective optimization); defect classification in insulation using image analytics.
Free/Open-Source AI Tools: TensorFlow; Python; Orange Data Mining

Unit III: Electrochemical Energy Storage and Conversion:
Core Content: Fundamentals of electrochemistry: electrode potential, Nernst equation and problems, potentiometry (redox reactions), conductometry (strong acid vs strong base and weak acid vs strong base); electrochemical cells (Galvanic and electrolytic cells); primary batteries: construction, working and electrical applications of zinc-air and sodium-air batteries; secondary batteries: construction, charging-discharging reactions and applications of lead-acid, Ni-Cd, lithium-ion batteries; supercapacitors: definition, classification and applications; electric fuel cells: construction, working and electrical applications of hydrogen-oxygen and methanol-oxygen fuel cells; battery management systems: state of charge, thermal management, safety.
AI Integration: Battery health prediction (SoH/SoC) using regression/ML; anomaly detection for thermal runaway; optimization of charge–discharge profiles using data analytics.
Free/Open-Source AI Tools: Python (NumPy, Pandas, scikit-learn); Google Colab

Unit IV: Corrosion Chemistry and Protective Techniques:
Core Content: Corrosion: definition, examples, types of corrosion — chemical corrosion and electrochemical corrosion, Pilling-Bedworth rule; galvanic corrosion, concentration cell corrosion, uniform, pitting, galvanic, crevice, stress and erosion corrosion; factors affecting corrosion; protective techniques (corrosion control): proper design, use of pure metal, metal alloys, inhibitors; cathodic protection — sacrificial anodic protection, impressed current cathodic protection; galvanizing: hot-dip galvanizing, electroplating; electroplating and electroless plating (nickel and copper); underground cable corrosion: stray current corrosion; corrosion of grounding systems: earth electrodes, grid corrosion; corrosion in electrical substations: steel structures, conductors; battery terminal corrosion; corrosion in cooling systems of electrical machines.
AI Integration: Predictive corrosion modeling using environmental/operational data; computer vision for coating/defect inspection; maintenance scheduling via ML.
Free/Open-Source AI Tools: Python (scikit-learn); OpenCV; Google Colab

Unit V: Coolants and Thermal Management in Electrical Systems:
Core Content: Transformer oils: mineral oil, synthetic esters, natural esters, silicone fluids; properties of oils; oil testing: breakdown voltage, acidity, moisture content, dissolved gas analysis (DGA); coolants for electrical machines: air, hydrogen, water, oil; water cooling: deionized water, corrosion inhibitors; hydrogen cooling in large generators: properties, purity requirements, safety; cooling fluids for power electronics: water-glycol, dielectric fluids; nanofluids for enhanced heat transfer; fire-resistant fluids: less-flammable oils, synthetic fluids.
AI Integration: Predictive analytics for hot-spot temperature and insulation life; anomaly detection from DGA data; optimization of cooling systems using data-driven models.
Free/Open-Source AI Tools: Python (Pandas, SciPy); scikit-learn; Google Colab

Text Books:

1. Jain, P. C., & Jain, M. (2020). Engineering chemistry (17th ed.). Dhanpat Rai Publishing Company.
2. Shikha, A., & Sharma, R. K. (2019). Engineering chemistry. Krishna Prakashan Media.

Reference Books:

1. Davis, M. L. (2020). Water and wastewater engineering: Design principles and practice (2nd ed.). McGraw-Hill Education.
2. Shikha, A., & Sharma, R. K. (2019). Engineering chemistry. Krishna Prakashan Media.
3. Dara, S. S., &Umare, S. S. (2018). Engineering chemistry (15th ed.). S. Chand Publishing.

Web References:

1. American Concrete Institute. (2021). ACI 222R-01: Protection of metals in concrete against corrosion. https://www.concrete.org
2. Bureau of Indian Standards. (2012). IS 456:2000 – Plain and reinforced concrete – Code of practice. https://www.services.bis.gov.in
3. U.S. Environmental Protection Agency. (2023). Green chemistry. https://www.epa.gov/greenchemistry
4. World Health Organization. (2022). Guidelines for drinking-water quality (4th ed.). https://www.who.int/publications/i/item/9789241549950
5. Indian Green Building Council. (2023). LEED India rating systems. https://igbc.in/igbc/

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26HMHS101T	HM	**Communicative English** (Common for all branches of Engineering)	2	0	0	0	2

Pre-Requisites: Basic English (Intermediate level).

Course Objectives:

- To develop strong foundations in English grammar, vocabulary, phonetics, and communication models essential for professional contexts.
- To develop listening, speaking, reading, and writing skills using structured activities and AI-powered tools.
- To enable students to compose professional emails, reports, and presentations using AI writing assistants.
- To develop critical analysis of communication styles, barriers, and discourse structures using AI tools.
- To cultivate responsible AI usage for language learning — verifying AI outputs, maintaining academic integrity, and using AI as an aid not a substitute.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the principles of communication, grammatical structures, tenses, subject–verb agreement, phonetic patterns, and vocabulary skills to identify and correct language errors and demonstrate effective spoken and written communication using AI-assisted tools. (L3)

CO2: Apply effective listening, speaking, group discussion, presentation, and debate techniques to organize and deliver clear, confident, and persuasive oral communication using AI-assisted tools for speech analysis, transcription, and visual presentation design. (L3)

CO3: Apply reading and writing strategies to comprehend, organize, and produce clear paragraphs, creative content, movie reviews, and blogs using AI-assisted paraphrasing, readability, simplification, and content-structuring tools. (L3)

CO4: Analyze professional and business communication requirements to evaluate and improve the structure, tone, formality, and effectiveness of emails, letters, reports, summaries, resumes, and LinkedIn profiles using AI-assisted communication tools. (L4)

CO5: Evaluate the effectiveness, accuracy, and ethical implications of AI-assisted language learning and communication tools, with regard to translation, plagiarism, academic integrity, bias, privacy, misinformation, and responsible digital communication. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						1	1	1	3		2		
CO2						2	1	2	3		2		
CO3						2	2	3	3		2		
CO4						2	2	2	3		2		
CO5						2	3	2	3		3		

Unit I: Foundations of English Communication:

Core Content: Basics of Communication: process, types (verbal/non-verbal), and barriers. Parts of Speech: nouns, pronouns, verbs, adjectives, adverbs. Tenses: simple, continuous, perfect, and perfect continuous. Subject-verb agreement and correction of sentences.

AI Integration: Grammar explanation, error identification, and instant Q&A on language rules, Real-time grammar, spelling, and punctuation correction with detailed explanations, Phonetics practice – voice

recording for pronunciation and intonation feedback, Word-of-the-day, etymology, and interactive vocabulary quizzes.
Free/Open-Source AI Tools: Grammarly (free); ChatGPT / Gemini (free tiers); Google Speech-to-Text (free); Merriam-Webster Vocabulary Builder (free)

Unit II: Listening and Speaking Skills:
Core Content: Principles of Effective Speaking and Active Listening. Group Discussion (GD): techniques, etiquette, and evaluation criteria. Presentation Skills: design principles, slide preparation, and delivery techniques. Debate: argument construction, evidence use, and rebuttal techniques.
AI Integration: AI-powered presentation design – structured, visual, professional slides, Fluency analysis, filler-word detection, pacing, and confidence scoring, Auto-transcription and keyword analysis of spoken communication, Visual communication design – infographics, posters, and slide aesthetics
Free/Open-Source AI Tools: Gamma.app (free); Orai AI Speech Coach (free); Otter.ai (free tier); Canva AI (free)

Unit III: Reading Comprehension & Writing Skills:
Core Content: Reading Strategies: skimming, scanning, intensive and extensive reading. Comprehension passages, synonyms, antonyms, words often confused, homonyms, homophones, and homographs. Paragraph Writing: topic sentence, supporting details, clincher sentence. Creative Writing — features and techniques — movie reviews and blogs.
AI Integration: AI paraphrasing tool – rewrite passages at varying complexity levels, Readability scoring, passive voice detection, and sentence simplification, Essay outline generation, argument structuring, and content brainstorming, Vocabulary simplification for improving comprehension of complex texts
Free/Open-Source AI Tools: QuillBot (free); Hemingway Editor (free online); ChatGPT (free); Rewordify (free)

Unit IV: Professional & Business Communication:
Core Content: Email Writing and Letter Writing: formal and informal. Short Report Writing: structure, types (formal/informal), and format. Note-Making and Summarization Techniques. Resume / CV Writing and LinkedIn Profile Optimization.
AI Integration: Draft and refine professional emails, cover letters, reports, and proposals, AI-powered resume builder with ATS-optimized templates, Report structuring, agenda writing, and knowledge management, Tone detection, formality check, and professional register evaluation
Free/Open-Source AI Tools: ChatGPT (free); Kickresume (free tier); Grammarly (free); Notion AI (free tier)

Unit V: Use of AI in Language Learning:
Core Content: Artificial Intelligence in Communication: meaning, basics, benefits and limitations of AI in human interaction. AI chatbots and virtual assistants in education: grammar correction, vocabulary building, AI-based speaking and pronunciation tools, writing assistants, e-portfolio creation. AI in Global Communication: automated translation, summarizing large texts, finding information in different languages. Digital Communication and AI Ethics: responsible use of AI in education, plagiarism and academic honesty, fake information and AI-generated content, privacy, bias, and ethical communication online.
AI Integration: Plagiarism detection, paraphrase checker, and academic integrity analysis, High-accuracy translation supporting intercultural communication understanding, AI literacy modules –

understanding AI bias, ethics, and responsible usage, E-portfolio creation platform for showcasing communication artifacts

Free/Open-Source AI Tools: Copyleaks / Turnitin (institutional access); DeepL Translator (free); Wix (free); Adobe Portfolio (free for students); ChatGPT / Gemini (free tiers)

Text Books:

1. Raman M. and Sharma S., Technical Communication: Principles and Practice, Oxford University Press, 4th Edition, 2022.
2. Shetty J. and Pareek R., Communicative English for Engineers and Professionals, Wiley India, 3rd Edition, 2021.

Reference Books:

1. Rizvi M.A., Effective Technical Communication, Tata McGraw-Hill, 2nd Edition, 2018.
2. Murphy H.A., Effective Business Communication, McGraw-Hill Education, 7th Edition, 2019.
3. Wren P.C. and Martin H., High School English Grammar and Composition, S. Chand, Revised Edition, 2019.

Web Resources & NPTEL Links:

1. British Council – Learn English – https://learnenglish.britishcouncil.org
2. NPTEL – Communication Skills – https://nptel.ac.in/courses/109104101
3. Purdue OWL (Online Writing Lab) – https://owl.purdue.edu
4. BBC Learning English – https://www.bbc.co.uk/learningenglish

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES03103T	ES	**Design Thinking** (Common for all branches of Engineering)	1	0	4	0	3

Pre-Requisites: Basic knowledge of problem-solving, creativity, communication, teamwork, and fundamental engineering concepts.

Course Objectives:

- To understand the principles, philosophy, stages, and importance of Design Thinking in human-centred engineering problem solving.
- To apply empathy methods such as interviews, observation, personas, and problem-framing techniques to identify and understand user needs.
- To analyse user needs, formulate effective problem statements using the How Might We framework, and leverage AI tools for generating insights and evaluating solutions.
- To develop, test, evaluate, and pitch innovative prototype solutions using AI ideation tools, user feedback, and digital prototyping platforms.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Recall and understand the philosophy, principles, and stages of the Design Thinking process and its relevance to human-centred problem solving. (L3)

CO2: Apply empathy tools (interviews, observation, personas) and problem-framing techniques to real-world engineering scenarios. (L3)

CO3: Analyse user needs and problem statements using the 'How Might We' framework and AI-powered insight generation tools. (L4)

CO4: Evaluate and select the most viable prototype solution through structured testing, user feedback, and AI-assisted evaluation rubrics. (L5)

CO5: Create innovative prototypes and pitch solutions to identified problems using AI ideation tools and digital prototyping platforms. (L6)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	1	2	2	1	1	2	1				2	1	2
CO2	1	2	3	2	2	3	1				2	2	2
CO3	2	3	3	2	3	3	1				3	2	2
CO4	2	3	3	3	3	2	2				3	2	2
CO5	2	3	3	3	3	2	2				3	2	2

Unit 1: Introduction to Design Thinking:

Core Concepts:

- Design Thinking: Definition, origin (Stanford d.school, IDEO), and relevance to engineering
- The 5-Stage Model: Empathise – Define – Ideate – Prototype – Test
- Human-centred design principles: Desirability, Feasibility, Viability
- Systems Thinking vs. Linear Thinking: Understanding complexity
- Case Studies: Successful design thinking applications (IDEO shopping cart, Stanford Mobile Clinic)

AI Integration:

- AI Tool: ChatGPT / Copilot – use AI to explain design thinking in different domains (healthcare, education, agriculture)

- Activity: Use MindMeister (free, open-source alternative: FreeMind) to create a mindmap of the 5-stage process
- AI Video Analysis: Watch an IDEO documentary and use AI to summarise key DT principles observed

Unit II: Empathise and Define Stages:

Core Concepts:

- Empathy Methods: Interviews, Observation, Immersion, Shadowing
- Creating User Personas: Demographic, psychographic, goals and pain points
- Empathy Mapping: Says, Thinks, Does, Feels
- Point of View (POV) Statement: User + Need + Insight formula
- How Might We (HMW) Questions: Reframing problems as opportunities

AI Integration:

- AI Tool: Miro (free plan) – create digital empathy maps and POV statements collaboratively
- AI Tool: ChatGPT – generate diverse user persona profiles for a given problem context
- Activity: Interview 5 peers about a campus problem; use AI to analyse interview transcripts and extract themes
- AI Tool: Dovetail (free plan) – affinity mapping of qualitative research data

Unit III: Ideation Stage:

Core Concepts:

- Creative Thinking Techniques: Brainstorming, Brainwriting, SCAMPER, Random Word Association
- Lateral Thinking: Edward de Bono's Six Thinking Hats
- Affinity Diagramming: Grouping ideas by themes
- Prioritisation Matrix: Impact vs. Effort, Dot Voting
- Concept Selection: Morphological charts and Pugh Matrix

AI Integration:

- AI Tool: ChatGPT – use 'expand my idea' prompts to generate 20 variations of a concept
- AI Tool: IdeaFlip (free) / Miro – digital brainstorming and affinity diagramming
- AI Tool: Midjourney (free trial) / DALL-E (free credits) – generate visual concept sketches from text descriptions
- Activity: Run a 30-minute AI-augmented brainstorming session and compare idea quality with pure human brainstorming

Unit IV: Prototyping:

Core Concepts:

- Principles of Prototyping: Low-fidelity vs. High-fidelity prototypes
- Paper Prototyping: Sketching UI and physical models
- Digital Wireframing: Introduction to wireframes and mockups
- 3D Prototyping Concepts: Introduction to 3D printing and rapid prototyping
- Role Plays and Experience Prototyping: Service design scenarios

AI Integration:

- AI Tool: Figma (free plan) – create digital wireframes and interactive mockups
- AI Tool: Uizard (free plan, AI-powered) – convert hand-drawn sketches to digital wireframes using AI
- AI Tool: Tinkercad (free, web-based) – create simple 3D models for physical prototyping
- Activity: Build a paper prototype for a chosen problem and photograph for digital documentation

Unit V: Testing, Iteration and AI in Design:

Core Concepts:

- User Testing Methods: Think-aloud protocol, A/B testing, Usability testing
- Feedback Analysis: Affinity mapping of test feedback
- Iteration Cycle: Incorporating feedback and pivoting
- Design Presentation: Pitching to stakeholders (elevator pitch, demo day format)
- AI in Design: Overview of AI-assisted design (generative design, co-creation with AI)

AI Integration:

- AI Tool: Maze (free plan) – conduct remote usability tests on Figma prototypes
- AI Tool: ChatGPT – analyse collected user feedback and generate insights and improvement suggestions
- Activity: Present a final prototype to a panel; use AI to prepare pitch slides in Canva
- Reflection: Create a Design Thinking journal documenting each stage with AI tool usage notes

Text Books:

1. Brown, Tim. (2019). Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation. HarperBusiness, New York.
2. Lewrick, Michael, Link, Patrick & Leifer, Larry. (2018). The Design Thinking Playbook. Wiley, New Jersey.

Reference Books:

1. IDEO.org. (2015). The Field Guide to Human-Centered Design. IDEO.org, San Francisco.
2. Kelley, Tom & Kelley, David. (2013). Creative Confidence: Unleashing the Creative Potential Within Us All. Crown Business.
3. Liedtka, Jeanne & Ogilvie, Tim. (2011). Designing for Growth: A Design Thinking Tool Kit for Managers. Columbia Business School Press.
4. Plattner, Hasso, Meinel, Christoph & Leifer, Larry (Eds.). (2018). Design Thinking Research. Springer.
5. Kumar, Vijay. (2012). 101 Design Methods: A Structured Approach for Driving Innovation. Wiley.
6. Stickdorn, Marc & Schneider, Jakob. (2021). This Is Service Design Thinking (Expanded ed.). BIS Publishers.

Web Resources:

1. Stanford d.school Resources: dschool.stanford.edu/resources
2. IDEO Design Kit: www.designkit.org/methods
3. MIT OpenCourseWare – Design Thinking: ocw.mit.edu
4. Interaction Design Foundation: www.interaction-design.org
5. NPTEL Design Thinking: swayam.gov.in

YouTube / Video Resources:

1. YouTube: Stanford d.school – Design Thinking Bootcamp – www.youtube.com/@stanforddschool
2. YouTube: IDEO U – www.youtube.com/@ideou
3. YouTube: Crash Course – Design Thinking Series – www.youtube.com/@crashcourse
4. YouTube: AJ&Smart – Design Sprint tutorials – www.youtube.com/@AJSmart
5. YouTube: NPTEL Design Thinking – www.youtube.com/c/nptel

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26PC02101T	PC	Electromagnetic Fields	2	1	0	0	3

Pre-Requisites: Basic knowledge of engineering mathematics, vector algebra and calculus, differential equations, basic electrical circuits, electric and magnetic fields, and fundamental concepts of physics.

Course Objectives:

- To introduce the fundamentals of vector calculus and electrostatics for analyzing electromagnetic field problems.
- To explain the behaviour of conductors and dielectrics in electric fields and analyze capacitance and energy storage mechanisms.
- To study magnetostatic fields, magnetic forces and their applications in electrical engineering systems.
- To understand self and mutual inductance in electromagnetic systems and their applications in electrical devices.
- To analyze time varying electromagnetic fields and Maxwell's equations using modern computational and AI tools.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the concepts of vector calculus, fundamental laws, self-inductance, and mutual inductance to solve basic electrical and electromagnetic engineering problems. (L3)

CO2: Understand the concepts of electrostatics, conductors, dielectrics, capacitance, magneto statics, magnetic fields, time varying fields, self and mutual inductances. (L3)

CO3: Apply vector calculus, Coulomb's law, Gauss's law, Ohm's law, Biot-Savart's law, Ampere's circuital law, Maxwell's third equation, self and mutual inductances, Faraday's laws, Maxwell's fourth equation. (L3)

CO4: Analyze vector calculus, electrostatic fields, and conductor behaviour, and apply AI and machine learning techniques for prediction, simulation, and analysis in electrostatic systems. (L4)

CO5: valuate magnetic forces, the motion of charges in magnetic fields, self and mutual inductance of cables and wires, and time-varying electromagnetic fields, and assess the effectiveness of AI techniques for field prediction, inductance estimation, and electromagnetic system analysis. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	2	1		1				1	3	3
CO2	3	3	3	2	1		1				1	3	2
CO3	3	3	3	2	1		1				1	3	3
CO4	3	3	3	2	1		1				1	3	3
CO5	3	3	3	2	1		1				1	3	3

Unit I: Vector Analysis and Electrostatics:

Core Content: Coordinate Systems: Rectangular- Cylindrical and Spherical coordinate systems. Differential elements. **Vector Calculus:** Differential length, Area and Volume- Del operator- Gradient of a scalar- Divergence of a vector and Divergence theorem (definition only)- Curl of a vector and Stoke's theorem (definition only)- Laplacian of a scalar. **Electrostatics:** Coulomb's law and Electric field intensity (EFI) – EFI due to Continuous charge distributions (line and surface charge)- Electric flux density- Gauss's law (Maxwell's first equation, $\nabla.D = \rho v$)- Electric Potential- Work done in moving a

point charge in an electrostatic field (second Maxwell's equation for static electric fields, $\nabla \times E = 0$)- Potential gradient- Laplace's and Poison's equations.
AI Integration: AI-based vector field visualization. Neural networks for electric field prediction. Machine learning models for electrostatic potential estimation. AI-assisted numerical solution of Laplace and Poisson equations.
Free/Open-Source AI Tools: Python (NumPy, SciPy, SymPy); TensorFlow / PyTorch; Jupyter Notebook; GNU Octave.

Unit II: Conductors, Dielectrics and Capacitance:
Core Content: Electric dipole and dipole moment – Potential and EFI due to an electric dipole- Torque on an Electric dipole placed in an electric field; Behaviour of conductors in an electric field- Polarization- dielectric constant and strength- Current density- conduction and convection current densities- Ohm's law in point form- Continuity equation; Boundary conditions- Capacitance of parallel plate, coaxial capacitors- Energy stored and density in a static electric field.
AI Integration: AI prediction of dielectric properties. Machine learning optimization of capacitor geometry. AI modeling of energy storage behaviour. Data-driven analysis of electric field distribution in dielectric materials.
Free/Open-Source AI Tools: Scikit-Learn; FEniCS (Finite Element simulation); OpenModelica; ParaView.

Unit III: Magnetic Circuits, Magnetostatics and Magnetic Forces:
Core Content: Basic definition of MMF- flux and reluctance- analogy between electrical and magnetic circuits. Biot-Savart's law and its applications viz. Straight current carrying filament, circular, and solenoid current carrying wire – Magnetic flux density and Maxwell's second Equation ($\nabla.B = 0$); Ampere's circuital law and its applications viz. MFI due to an infinite sheet, long filament, solenoid, toroidal current carrying conductor- point form of Ampere's circuital law- Maxwell's third equation ($\nabla \times H = J$)- Magnetic force, moving charges in a magnetic field – Lorentz force equation- force on a straight and a long current carrying conductor in a magnetic field- force between two straight long and parallel current carrying conductors- Magnetic dipole- Magnetic torque and moment.
AI Integration: Neural networks for magnetic flux density prediction. Machine learning models for magnetic field estimation. Intelligent simulation of magnetic forces.
Free/Open-Source AI Tools: Elmer FEM; OpenEMS; Python (SciPy); PyTorch.

Unit IV: Self and Mutual Inductance:
Core Content: Concept of self and mutual inductance; Dot convention – coefficient of coupling and composite magnetic circuit- analysis of series and parallel magnetic circuits. Determination of self-inductance of a solenoid, toroid, coaxial cable and mutual inductance between a straight long wire and a square loop wire in the same plane – Energy stored and energy density in a magnetic field.
AI Integration: Machine learning prediction of inductance values. Neural networks for magnetic energy storage estimation. Data-driven modeling of mutual inductance.
Free/Open-Source AI Tools: Python (SciPy); FEniCS; OpenEMS; Jupyter Notebook.

Unit V: Time Varying Fields:
Core Content: Faraday's laws of electromagnetic induction- Maxwell's fourth equation ($\nabla \times E = -\partial B/\partial t$)- integral and point forms of Maxwell's equations- Statically and Dynamically induced EMF- Displacement current- Modification of Maxwell's equations for time varying fields- Poynting theorem and Poynting vector.

AI Integration: Physics-Informed Neural Networks (PINNs) for solving Maxwell equations. Deep learning models for electromagnetic wave propagation. AI-based EM interference prediction. Intelligent analysis of power flow using Poynting vector.

Free/Open-Source AI Tools: Deep XDE (PINNs); Meep electromagnetic simulator; Tensor Flow; Open EMS.

Text Books:

1. Elements of Electromagnetics by Matthew N O Sadiku, Oxford Publications, 7th Edition, 2018.
2. Engineering Electromagnetics by William H. Hayt & John A. Buck, Mc. Graw-Hill, 7th Edition, 2006.

Reference Books:

1. Introduction to Electro Dynamics by D J Griffiths, Prentice-Hall of India Pvt. Ltd, 2nd Edition.
2. Electromagnetic Field Theory by K.A. Gangadhar and P.M. Ramanathan, Khanna Publishers, 16th Edition, 2015.
3. Electromagnetic Field Theory by Yaduvir Singh, Pearson India, 1st Edition, 2011.
4. Fundamentals of Engineering Electromagnetics by Sunil Bhooshan, Oxford University Press, 2012.
5. Schaum's Outline of Electromagnetics by Joseph A. Edminister, Mahamood Navi, 4th Edition, 2014.

Web Resources & NPTEL Links:

1. NPTEL: Electromagnetic Field Theory: https://nptel.ac.in/courses/108106073
2. MIT OpenCourseWare – Electromagnetics and Applications: https://ocw.mit.edu/courses/6-013-electromagnetics-and-applications/
3. MIT Computational Electromagnetics: https://ocw.mit.edu/courses/computational-electromagnetics/
4. Physics Informed Neural Networks: https://maziarraissi.github.io/PINNs/
5. Python Scientific Computing: https://scipy.org

Recommended Free & Open-Source AI/Computational Tools:

1. Python (NumPy, SciPy, SymPy) – free on Jupyter Notebook / Google Colab for vector field and electrostatic computation
2. TensorFlow / PyTorch (free, open-source) – for AI/ML based field prediction and PINN modelling
3. Scikit-Learn (free, open-source) – for regression and optimization of dielectric and capacitor properties
4. FEniCS (free, open-source) – FEM-based electrostatic and magnetostatic simulation
5. OpenModelica (free, open-source) – for system-level electromagnetic modelling
6. ParaView (free, open-source) – for visualization of field simulation results
7. Elmer FEM (free, open-source) – magnetostatic and electromagnetic finite element analysis
8. OpenEMS (free, open-source) – electromagnetic field simulation with Python interface
9. DeepXDE (free, open-source) – for solving Maxwell's equations using PINNs
10. Meep electromagnetic simulator (free, open-source) – for time-varying field and wave propagation studies
11. GNU Octave (free, open-source) – for numerical vector calculus computation

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26BSHB108P	BS	Engineering Chemistry for Electrical Engineers Lab	0	0	0	3	1.5

Pre-Requisites: Basic knowledge of general chemistry, chemical reactions, solution preparation, concentration calculations, and fundamental laboratory safety and measurement techniques.

Course Objectives:

- To develop practical skills in volumetric, redox, conductometric, and potentiometric analysis for the quantitative estimation of chemical substances.
- To familiarize students with the preparation and characterization of polymers and the determination of corrosion behaviour under different environmental conditions.
- To develop competency in performing physicochemical measurements, including the determination of acid strength in batteries and viscosity of lubricating oils using appropriate laboratory instruments.
- To enable students to apply fundamental principles of analytical chemistry, electrochemistry, corrosion, polymers, and lubricants in conducting experiments, recording observations, and interpreting experimental results accurately.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the principles of volumetric, conductometric, potentiometric, and corrosion analysis to perform engineering chemistry laboratory experiments and obtain accurate results. (L3)

CO2: Perform chemical experiments for estimation of metals, battery parameters, polymer preparation, and viscosity measurements by following standard laboratory procedures. (L3)

CO3: Analyze experimental data obtained from titrations, corrosion studies, and lubricant characterization to determine chemical parameters and interpret results. (L4)

CO4: Apply computational and AI-based tools such as Python and Excel for data logging, visualization, and prediction of corrosion rates, battery characteristics, and lubricant properties. (L3)

CO5: Demonstrate professional laboratory practices, teamwork, safety standards, and effective communication while conducting chemistry experiments and preparing reports. (L3)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	3	3	1	3	3	3		2	1	2
CO2	3	3	2	3	3	3	3	3	3		2	2	2
CO3	3	3	2	3	2	3	3	3	3		2	2	2
CO4	3	3	3	3	3	1	3	3	3		3	2	2
CO5	3	3	3	3	3	2	3	3	3		3	1	2

List of Experiments:

1. Estimation of copper using sodium thiosulphate (iodometry).
2. Determination of Strength of Acid in Lead Acid Battery.
3. Preparation of polymer (Bakelite).
4. Estimation of ferrous iron (Fe^{2+}) using potassium dichromate (K2Cr2O7) (redox titrations).
5. Determination of corrosion.
6. Determination of corrosion by environment.
7. Conductometric titrations of strong acid vs strong base (e.g., HCl vs NaOH).

8. Conductometric titrations of weak acid vs strong base (e.g., CH_3COOH vs NaOH).
9. Potentiometry (redox titrations) — estimation of iron (Fe^{2+}) using potassium dichromate ($K_2Cr_2O_7$).
10. Determination of viscosity of lubricating oil using Red-Wood Viscometer – I.
11. Determination of viscosity of lubricating oil using Red-Wood Viscometer – II.

AI Integration:

AI Extension: Data logging and analysis using Python/Excel; predictive models for corrosion rates, battery parameters, and lubricant properties.

Web Resources & NPTEL Links:

1. IEEE Standards
2. IEC
3. IRENA
4. U.S. DOE (Energy Storage)
5. BIS

Recommended Free & Open-Source AI/Computational Tools:

1. Python (NumPy, Pandas, Matplotlib)
2. TensorFlow / scikit-learn
3. OpenCV
4. Orange Data Mining

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26HMHS101P	HM	**Communicative English Lab** (Common for all branches of Engineering)	0	0	0	2	1

Pre-Requisites: Basic knowledge of English grammar, vocabulary, reading, writing, and everyday communication, with a willingness to participate in listening, speaking, and digital language-learning activities.

Course Objectives:

- To develop students' oral communication skills through structured speaking and listening exercises.
- To enhance pronunciation, intonation, and fluency using AI-assisted speech tools.
- To build professional communication competencies including group discussion, debate, and presentation skills.
- To enable students to write effective professional documents such as emails, reports, and cover letters.
- To integrate AI tools in language learning for self-assessment, feedback, and continuous improvement

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Pronounce English words correctly using standard phonemic symbols and demonstrate improved speech fluency. (L3)

CO2: Apply active listening skills to comprehend spoken English and respond appropriately in formal and informal communication contexts. (L3)

CO3: Participate effectively in group discussions, presentations, and debates demonstrating logical argumentation. (L4)

CO4: Write grammatically correct and contextually appropriate professional documents including emails, reports, and resumes. (L6)

CO5: Utilize AI-powered tools for language learning, self-evaluation, and continuous improvement of English skills. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						2	3	3	3		2		
CO2						2	3	3	3		2		
CO3						3	3	3	3		2		
CO4						2	3	3	3		2		
CO5						2	3	3	3		2		

Lab Syllabus – 12 Topics with AI Integration:

S. No.	Lab Topic	Lab Activities	AI Activity	AI Tools	COs
1.	Phonetics & Pronunciation Practice	Record & playback vowel/consonant articulation; IPA chart drill; Minimal pair exercises	Use AI speech-to-text to detect mispronunciations; real-time phoneme feedback via ELSA Speak	ELSA Speak, Google Speech Recognition, Speechling	CO1 & CO5

2.	Listening Comprehension &Note-Taking	Listen to TED Talks/podcasts; fill-in-the-blank exercises; summarise audio passages	Auto-generate transcripts; compare student notes with AI-generated summaries using Otter.ai	Otter.ai, YouTube Auto-captions, Whisper AI	CO2 & CO5
3.	Spoken English & Fluency Building	Impromptu speaking (1–2 min); picture description; story completion tasks	AI analyses speech for filler words, pace, and clarity; ChatGPT prompts for spontaneous speaking	Speeko, ChatGPT, Google Assistant	CO1, CO3 & CO5
4.	Group Discussion Techniques	Role-based GD on current topics; turn-taking practice; consensus building	AI evaluates argument structure and language; ChatGPT generates counter-arguments for practice	ChatGPT, Gemini, Mentimeter	CO2 & CO3
5.	Public Speaking & Presentations	Prepare and deliver 3-minute presentations; use of visual aids; Q&A handling	AI slide deck generation; real-time delivery feedback on pacing and confidence	Beautiful.ai, Gamma.app, Microsoft Copilot	CO3 & CO5
6.	Interview Skills & Mock Interviews	HR round simulation; STAR method answers; body language and eye contact	AI-powered mock interview with instant feedback on content, grammar, and tone	Yoodli, Interview Warmup by Google, ChatGPT	CO1, CO3 & CO5
7.	Professional Email & Business Writing	Draft formal emails, complaints, and requests; subject line formulation; netiquette	AI refines email drafts; tone & clarity check; compare student draft with AI-improved version	Grammarly, ChatGPT, Gemini	CO4 & CO5
8.	Resume & Cover Letter Writing	ATS-friendly resume design; action verbs usage; tailoring cover letters to job descriptions	AI resume scanner; keyword optimisation suggestions; ChatGPT cover letter generation and critique	Kickresume, Resume.io, Jobscan, ChatGPT	CO4 & CO5
9.	Reading Comprehension & Critical Thinking	Read articles/editorials; identify main idea, inference, and tone; speed reading drills	AI generates comprehension questions; text summarisation and comparative analysis	Quillbot, Speechify, NotebookLM	CO2 & CO4
10.	Debate & Argumentation Skills	Structured debate (for/against); Rebuttal techniques; logical fallacy identification	AI generates arguments on both sides; evaluate logical structure using	ChatGPT, Claude.ai, Kialo Edu	CO2 & CO3

			ChatGPT		
11.	Report Writing & Technical Communication	Lab report, incident report, and project report formats; headings, data presentation	AI improves report coherence and passive/active voice usage; grammar and structure feedback	Grammarly Business, ChatGPT, Hemingway Editor	CO4 & CO5
12.	Etiquette, Netiquette & Personal Branding	Workplace conversation role play; LinkedIn profile creation; professional social media etiquette	AI reviews LinkedIn summaries; digital persona analysis; personal brand statement via ChatGPT	ChatGPT, Canva AI, LinkedIn AI features	CO3, CO4 & CO5

Text Books:

1. Raman, Meenakshi & Sharma, Sangeeta. Technical Communication: Principles and Practice (3rd Ed.). Oxford University Press, 2022.
2. Krishnaswamy, N. & Krishnaswamy, L. The Story of English in India. Foundation Books, Cambridge University Press, 2021.

Reference Books:

1. Swan, Michael. Practical English Usage (4th Ed.). Oxford University Press, 2016.
2. Murphy, Raymond. English Grammar in Use (5th Ed.). Cambridge University Press, 2019.
3. Rizvi, M. Ashraf. Effective Technical Communication (2nd Ed.). McGraw-Hill Education, 2018.
4. Turton, N.D. & Heaton, J.B. Longman Dictionary of Common Errors. Pearson Longman, 2017.
5. Lesikar, Raymond V. & Flatley, Marie E. Basic Business Communication (12th Ed.). McGraw-Hill, 2020.
6. Rutherford, Andrea J. Basic Communication Skills for Technology (3rd Ed.). Pearson, 2019.

Online Resources

1. BBC Learning English – https://www.bbc.co.uk/learningenglish (Grammar, pronunciation, and listening)
2. British Council Learn English – https://learnenglish.britishcouncil.org
3. TED Talks – https://www.ted.com/talks (Listening & speaking practice)
4. Coursera English Communication Skills Specialisation – https://www.coursera.org
5. edX English for Professional and Academic Purposes – https://www.edx.org
6. Grammarly Blog – https://www.grammarly.com/blog (Writing tips)
7. ELSA Speak (App) – https://elsaspeak.com (AI pronunciation coach)

YouTube Channels:

1. English with Lucy – @EnglishwithLucy (Vocabulary, speaking, British accent)
2. Learn English with TV Series – @LearnEnglishWithTVSeries
3. Rachel's English – @rachelsenglish (American accent and pronunciation)
4. TED-Ed – @TEDEd (Critical thinking and communication ideas)
5. Spoken English Hub – @SpokenEnglishHub (Indian context, interview prep)
6. JenniferESL – @JenniferESL (Grammar and business English)

NPTEL / Swayam Courses:

1. Developing Soft Skills and Personality, NPTEL / SWAYAM, IIT Kanpur, https://swayam.gov.in
2. Business English Communication Suite, Coursera (Free Audit), University of Washington, https://coursera.org
3. Speak English Professionally: In Person, Online & On the Phone, Coursera, Georgia Tech, https://coursera.org
4. English and Academic Preparation, NPTEL, IIT Bombay, https://nptel.ac.in
5. Technical English for Engineers, NPTEL / SWAYAM, IIT Roorkee, https://swayam.gov.in

Lab Software & AI Tools Required:

Software / Tool	Type	Purpose
ELSA Speak	AI Pronunciation Coach App	Real-time speech accuracy & fluency feedback
Grammarly	AI Writing Assistant	Grammar, punctuation, style correction
ChatGPT / Claude.ai	Generative AI Chatbot	Conversation practice, writing assistance, debate prep
Otter.ai	AI Transcription Tool	Transcribe and analyse spoken content
Yoodli	AI Public Speaking Coach	Presentation and interview delivery analysis
Hemingway Editor	Writing Clarity Tool	Improve readability and sentence structure
Mentimeter	Interactive Presentation	Live polls, word clouds during GD sessions
Audacity	Audio Recording Software	Record and review pronunciation exercises
Google Meet / Zoom	Video Conferencing	Mock interviews and virtual presentations
Canva / Beautiful.ai	AI Presentation Design	Create professional visual presentations

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26PC02102P	PC	Electrical Engineering Workshop Practices	0	0	0	2	1

Pre-Requisites: Basic knowledge of electrical circuits, electrical/electronic components, Ohm's law, AC/DC fundamentals, circuit diagrams, and elementary mathematics and measurement concepts.

Course Objectives:

- To know about different tools, abbreviations and symbols in Electrical Engineering.
- To learn about types of measuring instruments to measure electrical quantities.
- To gain knowledge on different types of earthing and earth resistance.
- To study different types of wiring.
- To study color code, resistors, ICs, transistors, capacitors, diodes, SCRs, IGBTs.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the functions, symbols, abbreviations, safety practices, and operating principles of electrical tools, wiring accessories, electronic components, and basic electrical/electronic equipment in laboratory and residential applications. (L3)

CO2: Apply electrical wiring techniques, wire joint preparation, soldering practices, earthing methods, and measurement procedures for voltage, current, power, power factor, energy, and earth resistance using standard instruments. (L3)

CO3: Analyze the operating principles and performance characteristics of RLC circuits, fluorescent lamps, PV cells, transformers, induction motors, energy meters, and electrical gadgets such as CFLs and LEDs. (L3)

CO4: Apply and analyze troubleshooting techniques to identify faults and evaluate the performance of electrical and electronic equipment including fans, iron boxes, mixer-grinders, transformers and basic electronic circuits. (L3)

CO5: Design simple residential wiring systems, basic transformer/choke assemblies, and elementary electronic circuits by selecting suitable wires, components, protective devices, and measurement instruments for safe and efficient operation. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	1			2	1	3	3	3		1	3	2
CO2	3	2	1	2	3	2	3	3	3		1	3	3
CO3	3	3	1	2	3	2	3	3	3		1	3	3
CO4	3	3	2	3	3	1	3	3	3		1	3	3
CO5	3	3	3	2	3	2	3	3	3		1	3	3

List of Experiments:

Any 12 of the following practices are to be performed:

1. Study of Introduction to Electrical tools, symbols and abbreviations.
2. Study of types of sizes of wires and making "T" joint and straight joint for wires.
3. Measurements of Electrical quantities (like Voltage, Current, Power, Power factor in RLC circuits).
4. Measurements of Energy (using Single phase and Three phase Energy meter) by connecting different loads.
5. Study of earthing and measurement of earth resistance.

6. Study and performance of residential wiring (using Energy meter, Fuses, Switches, Indicator, Lamps, etc.).
7. Study of Fluorescent lamp wiring.
8. Study of various electrical gadgets (CFL and LED).
9. Study of PV Cell.
10. Study of cross-sectional aspects Induction motor and Transformer.
11. Assembly of choke or small transformer.
12. Study of trouble shooting of electrical equipment's (fan, iron box, mixer-grinder, etc.).
13. Introduction to basics of electronic components: Solder practice, Multi meter, Power supply.
14. Measurement of wire gauges using gauge meter.
15. Identification of color code, resistors, ICs, Transistors, capacitors, diodes, SCRs, IGBTs.

Reference Books:

1. Lab manual of Electrical Engineering by TTTI, Chennai

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26HMHS102L	HM	**Health and Wellness, Yoga and Sports** (Common for all branches of Engineering)	0	0	0	1	0.5

Pre-Requisites: Basic knowledge of health, fitness, nutrition, and hygiene; willingness to participate in yoga, meditation, sports, and physical activities; basic digital literacy; and an interest in using technology and AI tools for health, fitness, and wellness management.

Course Objectives:

- To create awareness about holistic health, wellness and preventive healthcare among students.
- To train students in yogic practices (asanas, pranayama, meditation) for physical and mental well-being.
- To enable students to design evidence-based personal fitness and nutrition plans.
- To develop sportsmanship, team spirit, fair play and physical fitness through indoor and outdoor sports.
- To integrate technology and AI tools for health monitoring, performance analysis and wellness management.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Demonstrate healthy lifestyle practices by applying principles of holistic health, balanced nutrition, physical fitness, stress management, sleep hygiene, personal hygiene, disease prevention, and ergonomic safety in daily life. (L3)

CO2: Demonstrate appropriate yoga, pranayama, meditation, mindfulness, and relaxation practices with correct techniques and breathing patterns to enhance flexibility, strength, balance, stress management, and overall well-being. (L3)

CO3: Analyze physical fitness components, training principles, sports skills, injury-prevention strategies, psychological factors, gender-inclusive practices, and sports technologies to evaluate and enhance individual and team performance. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						3	3	3	2		2		
CO2						3	3	3	2		2		
CO3						3	3	3	2		2		

Unit I: Health, Wellness and Preventive Healthcare:

- Concept of health, wellness and fitness – definitions and seven dimensions
- WHO definition of health; holistic health model (Physical, Mental, Social, Spiritual)
- Nutrition basics – macronutrients, micronutrients, balanced diet, BMI calculation
- Lifestyle diseases – diabetes, hypertension, obesity – causes and prevention
- Mental health & stress management – causes, symptoms, COPE model
- Sleep hygiene and circadian rhythm regulation
- Substance abuse – awareness, prevention and rehabilitation
- Personal hygiene, sanitation and communicable disease prevention
- Ergonomics and occupational health for students

Unit II: Yoga, Pranayama, Meditation and Mind-Body Practice:

- Introduction to Yoga – history, philosophy, Patanjali's Ashtanga Yoga
- Surya Namaskar – 12-step sequence with correct breathing

- Asanas for flexibility: Tadasana, Trikonasana, Paschimottanasana, Sarvangasana
- Asanas for strength & balance: Virabhadrasana I & II, Vrikshasana, Naukasana
- Pranayama techniques: Anulom-Vilom, Bhastrika, Kapalbhati, Bhramari
- Meditation and mindfulness – Dharana, Dhyana, body-scan, guided relaxation
- Yoga Nidra for sleep and stress relief
- Yoga for specific conditions – stress, back pain, PCOD/PCOS, anxiety
- Shatkarmas (cleansing processes) – an overview; International Day of Yoga

Unit III: Physical Fitness, Sports Skills and Sports Technology:

- Physical fitness components – strength, endurance, flexibility, agility, speed, coordination
- Principles of training
- – overload, progression, specificity, reversibility, individuality
- Warm-up, cool-down and dynamic stretching routines
- Indoor sports: Badminton, Table Tennis, Chess, Carrom – rules and basic skills
- Outdoor sports: Cricket, Football, Volleyball, Kabaddi, Athletics – rules and skills
- Sports injuries – types, RICE method, prevention and return-to-play protocol
- Sports psychology – motivation, goal setting, concentration and visualisation
- Gender equity and inclusivity in sports; Para-sports awareness
- Sports technology – wearables, fitness trackers, GPS analytics, performance

Text Books:

1. Health and Physical Education, Dr. S. K. Mangal & Shubhra Mangal, PHI Learning Pvt. Ltd. 2021.
2. The Science of Yoga, William J. Broad, Simon & Schuster, 2012.

Reference Books:

1. Yoga: The Iyengar Way, Silva, Mira & Shyam Mehta, Dorling Kindersley, 2019.
2. Sports Medicine Essentials, Jim Clover, Cengage Learning, 2020.
3. Foundations of Physical Education & Sport, Charles A. Bucher, McGraw-Hill Education, 2018.
4. Mindfulness: An Eight-Week Plan for Finding Peace, Mark Williams & Danny Penman, Rodale Books, 2018.

Web Resources:

1. Ministry of AYUSH – https://main.ayush.gov.in
2. Yoga for Youth (International Day of Yoga) – https://yoga.ayush.gov.in
3. National Health Portal – https://nhp.gov.in
4. ICMR Dietary Guidelines – https://www.icmr.nic.in
5. Sports Authority of India – https://www.sportsauthorityofindia.nic.in
6. WHO Physical Activity Guidelines – https://www.who.int/news-room/fact-sheets/detail/physical-activity
7. Khelo India Programme – https://kheloindia.gov.in
8. NIMHANS (Mental Health Resources) – https://nimhans.ac.in

YouTube Resources:

1. Surya Namaskar – Complete 12-Step Tutorial – https://youtu.be/surya_tut_01
2. Pranayama for Beginners (Swami Ramdev) – https://youtu.be/pranayama_sr_02
3. Mindfulness Meditation (10 Min) – https://youtu.be/mindful_med_03
4. Balanced Nutrition Explained (ICMR) – https://youtu.be/nutrition_icmr_04
5. Sports Injury Prevention Tips – https://youtu.be/injury_prev_05

6. Mental Health Awareness for Students – https://youtu.be/mh_students_06
7. Yoga for Stress Relief – Isha Foundation – https://youtu.be/yoga_isha_07
8. Fit India Movement – PM Modi Launch – https://youtu.be/fitindia_pmo_08

AI-Integrated Activities:

S. No.	AI Activity	Description / Tool
1.	AI Fitness Assessment Dashboard	Log BMI, steps and sleep for 2 weeks using Google Fit / Apple Health; generate AI health-insight reports and compare with WHO benchmarks.
2.	Personalised Diet Planner with AI	Use NutriAI / Eat Right Now app to design a 7-day balanced meal plan; compare output with ICMR dietary guidelines.
3.	Yoga Pose Correction using ML Camera	Use YogaNotch / KAIA Health app for real-time AI posture feedback; record before/after improvement videos.
4.	Mental Health Chatbot Interaction & Reflection	Engage with Woebot / Wysa AI mental-health chatbot for one week; write a critical reflection on AI vs human counsellor effectiveness.
5.	Sports Performance Analytics with Python	Analyse a cricket/football dataset on Google Colab using Pandas; visualise player performance KPIs with Matplotlib / Seaborn.

B. Tech. – I Year II Semester

Course Code	Category	Name of the Course	L	T	Pr.	P	C
26ES05102A	MC	**Cybersecurity Awareness for Engineers** (Common for all branches of Engineering)	2	0	0	0	0

Pre-Requisites: Basic knowledge of computer systems, networking, internet usage, digital communication, information security, and fundamental cyber safety practices.

Course Objectives:

- To introduce students to the fundamentals of cybersecurity and secure digital practices in modern society.
- To create awareness about cyber threats, cybercrimes, and methods for protecting personal and institutional digital assets.
- To develop responsible digital citizenship through cyber ethics, privacy awareness, and legal understanding.
- To enable students to adopt cyber hygiene practices for secure use of digital technologies.
- To familiarize students with emerging cybersecurity concerns in cloud, digital platforms, and AI-enabled environments.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply fundamental cybersecurity concepts to address security requirements in digital environments. (L3)

CO2: Analyze common cyber threats, cyberattacks, and social engineering techniques to determine appropriate security measures. (L4)

CO3: Apply secure practices for protecting devices, networks, email, cloud services, and online platforms. (L3)

CO4: Evaluate cyber ethics, privacy protection measures, and applicable cyber laws in different digital scenarios. (L5)

CO5: Evaluate cybersecurity incidents and develop appropriate cyber hygiene and response strategies. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	3	2	2				2	1	2
CO2	3	3	3	3	3	2	2				3	1	2
CO3	3	3	3	2	3	2	2				3	2	2
CO4	2	3	2	3	2	3	3				3	1	2
CO5	3	3	3	3	3	3	3				3	2	2

Unit I: Introduction to Cyberspace and Cybersecurity:

Introduction to cyberspace and cybersecurity – Evolution of information systems and Internet – Importance of cybersecurity in engineering and society – Information security concepts – CIA Triad (Confidentiality, Integrity, Availability) – Digital footprint and digital identity – Security governance basics – Cyber threat actors – Engineering ethics and responsible digital behavior – Cyber hygiene practices.

Unit II: Cyber Threats, Cybercrime and Social Engineering:

Cyber threats and vulnerabilities – Cybercrime and cyber offenders – Malware: Virus, Worm, Trojan, Spyware, Ransomware – Cyberattack lifecycle and common attack methods – Phishing, Spear Phishing, Smishing and Vishing – Social engineering attacks – Password attacks and credential theft – Identity theft and financial fraud – Insider threats – AI-enabled cyber threats and deepfakes – Case studies of major cyber incidents.

Unit III: Personal Cyber Safety and Cyber Security Technologies:
Password management and Multi-Factor Authentication – Authentication and access control concepts – Safe browsing practices – Email and communication security – Mobile device security – Cloud security basics – Secure use of Wi-Fi and public networks – Data backup and recovery – Cryptography and digital signatures (awareness level only) – Safe use of AI tools and digital platforms.

Unit IV: Privacy, Cyber Ethics and Cyber Laws:
Data protection and privacy concepts – Personal data and sensitive information – Cyber ethics and responsible digital behavior – Overview of cyber laws in India – Information Technology (IT) Act and related provisions (awareness level) – Common cybercrimes and legal implications – Cybercrime reporting mechanisms and grievance redressal – Digital evidence basics – Social media responsibility and legal considerations.

Unit V: Cyber Hygiene and Secure Digital Practices:
Cybersecurity risk awareness – Password management and Multi-Factor Authentication – Secure use of email, web, cloud and public networks – Safe digital transactions and digital payment awareness – Secure use of social media and AI tools – Cybersecurity awareness in IoT and emerging technologies – Incident awareness and cyber hygiene best practices.

Text Books:

1. Ajay Singh, Introduction to Cyber Security: Concepts, Principles, Technologies and Practices, Universities Press / Orient BlackSwan, Latest Edition.
2. Nina Godbole and SunitBelapure, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India.

Reference Books:

1. Charles J. Brooks, Christopher Grow, Philip Craig and Donald Short, Cybersecurity Essentials, Wiley
2. William Stallings and Lawrie Brown, Computer Security: Principles and Practice, Pearson.
3. Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, Cengage Learning.
4. Joseph Migga Kizza, Guide to Computer Network Security, Springer.
5. Eric Cole, Cybersecurity for Beginners, Wiley.

Online Resources:

1. National Cyber Security Awareness Portal (India)
2. CERT-In (Indian Computer Emergency Response Team)
3. National Cyber Crime Reporting Portal
4. NIST Cybersecurity Framework Resources
5. OWASP Foundation
6. Cisco Networking Academy – Introduction to Cybersecurity

B.Tech.–II Year
I Semester

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26BSHB211T	BS	**Complex Variables and Probability Theory** (Common to CE, EEE & ECE)	2	1	0	0	3

Pre Requisites: Linear Algebra and Calculus, Differential Equations & Vector Calculus.

Course Objectives:

- To develop a thorough understanding of complex variable theory, analyticity, Cauchy-Riemann conditions, and harmonic functions as foundations for engineering analysis.
- To apply complex integration techniques, series expansions, and the residue theorem to evaluate contour integrals and real definite integrals.
- To formulate probabilistic models using axioms, random variable theory, and standard distributions, and compute correlation and regression measures.
- To integrate analytical methods with AI-assisted computational tools for visualization, simulation, and verification.
- To develop outcome-oriented problem-solving skills aligned with NEP 2020 and modern engineering applications

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the axioms and theorems of probability to solve problems and determine the properties of discrete and continuous random variables, including expectation and variance. (L3)

CO2: Evaluate probabilities using binomial, Poisson, and normal distributions and interpret correlation and regression relationships in engineering data sets. (L3)

CO3: Recall definitions of limits, continuity, and differentiability of complex functions and state the Cauchy-Riemann equations in Cartesian and polar forms. (L4)

CO4: Apply Cauchy’s Integral Theorem, Cauchy’s Integral Formula, Liouville’s Theorem, and the Maximum Modulus Theorem to evaluate complex line integrals and solve related problems. (L3)

CO5: Analyze complex functions using Taylor's and Laurent's series, classify singularities, and evaluate real definite integrals by the residue theorem. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	1					1				1	1	1
CO2	3	2	1				1				1	3	2
CO3	3	3	2	2			1				1	1	1
CO4	3	2		1			1				1	1	1
CO5	3	3	1	2			1				1	2	2

Unit I: Probability & Random Variable:

Core Content: Sample space and events; axioms of probability; addition and multiplication theorems; conditional probability; Bayes' theorem; random variables: discrete and continuous; probability mass function; probability density function; cumulative distribution function; mathematical expectation and variance.

AI Integration: Python-based simulation of sample spaces and probability experiments; visualization of PMF, PDF, and CDF using Matplotlib; AI-assisted explanation of Bayes' theorem with real-world examples; NumPy/SciPy computation of expectation and variance for standard distributions.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib) on Google Colab; SymPy for symbolic probability computations; Jupyter Notebook; scikit-learn for exploratory data modelling.

Unit II: Probability Distributions:

Core Content: Binomial distribution: properties, mean and variance; Poisson distribution: properties, Poisson approximation to binomial; normal distribution: properties, standard normal curve, areas under normal curve; correlation coefficient; rank correlation; regression lines and regression coefficients.

AI Integration: SciPy-based computation and visualization of binomial, Poisson, and normal distributions; AI-assisted plotting of standard normal curve and computation of area probabilities; Python-based correlation and regression analysis on engineering data sets; use of AI tools to interpret regression coefficients and correlation strength.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib, pandas) on Google Colab; scikit-learn for regression and correlation analysis; Jupyter Notebook; WolframAlpha free tier; GNU Octave.

Unit III: Complex Functions and Analyticity: Complex Integration:

Core Content: Functions of a complex variable; limits and continuity; differentiability; analyticity; Cauchy-Riemann equations in Cartesian coordinates; Cauchy-Riemann equations in polar coordinates; harmonic functions; conjugate harmonic functions; Milne-Thomson method.

AI Integration: Python-based visualization of complex functions and their mappings; symbolic verification of Cauchy–Riemann equations using SymPy; AI-assisted plotting of harmonic and conjugate harmonic surfaces; use of ChatGPT/Copilot to explain analyticity and the Milne–Thomson construction.

Free/Open-Source AI Tools: Python (NumPy, Matplotlib, SymPy) on Google Colab; Jupyter Notebook; Wolfram Alpha free tier for verification; GNU Octave.

Unit IV: Complex Integration:

Core Content: Line integrals in the complex plane; Cauchy's integral theorem; Cauchy's integral formula; generalised Cauchy's integral formula; Liouville's theorem; maximum modulus theorem.

AI Integration: Numerical evaluation of complex line integrals using SciPy; AI-assisted verification of Cauchy's integral theorem results; visualization of contour paths in the complex plane using Matplotlib; use of AI tools to interpret Liouville's theorem and maximum modulus principle.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Matplotlib, SymPy) on Google Colab; Jupyter Notebook; Wolfram Alpha free tier; GNU Octave.

Unit V: Series Expansions and Residues:

Core Content: Taylor's series; Laurent's series; zeros and singularities: isolated, removable, poles, essential; residues; Cauchy's residue theorem; evaluation of real definite integrals using the residue theorem.

AI Integration: SymPy-based Taylor and Laurent series generation for complex functions; AI-assisted classification of singularities and residue computation; Python visualization of convergence regions; numerical validation of definite integrals using SciPy and comparison with residue-based results.

Free/Open-Source AI Tools: Python (SymPy, NumPy, SciPy, Matplotlib) on Google Colab; Jupyter Notebook; WolframAlpha free tier for symbolic verification.

Text Books:

1. Kreyszig E., Advanced Engineering Mathematics, 10th Edition, Wiley, 2011.
2. Walpole R.E. et al., Probability & Statistics for Engineers and Scientists, 9th Edition, Pearson, 2016.
3. Churchill R.V. and Brown J.W., Complex Variables and Applications, 9th Edition, McGraw-Hill, 2013.

Reference Books:

1. Ross S.M., Introduction to Probability and Statistics for Engineers and Scientists, 5th Edition, Academic Press, 2014.
2. Grewal B.S., Higher Engineering Mathematics, 44th Edition, Khanna Publishers, 2021.
3. Gupta S.C. and Kapoor V.K., Fundamentals of Mathematical Statistics, 12th Edition, Sultan Chand & Sons, 2020.

Web Resources & NPTEL Links:

- NPTEL: Complex Analysis – https://nptel.ac.in
- NPTEL: Probability and Statistics – https://nptel.ac.in
- MIT OpenCourseWare – Complex Variables: https://ocw.mit.edu
- Google Colab for Python: https://colab.research.google.com

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26HMHS205T	HM	**Managerial Economics and Financial Analysis** (Common to all branches of Engineering)	2	0	0	0	2

Pre Requisites: Basic Mathematics & Engineering Economics concepts.

Course Objectives:

- To understand fundamental concepts of managerial economics, demand analysis, and elasticity to make data-driven business decisions.
- To apply production and cost analysis techniques, including break-even analysis, to optimise organisational resource allocation.
- To analyse various forms of business organisations, market structures, and pricing strategies for competitive business environments.
- To evaluate capital budgeting proposals using NPV, IRR, ARR, and Payback methods for strategic investment decision-making.
- To prepare and interpret financial statements, compute key financial ratios, and employ AI/ML techniques for automated financial analysis.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply fundamental concepts of economics, including demand, supply, market structures, and the time value of money, to analyze and solve basic economic problems. (L3)

CO2: Apply economic theories of demand forecasting, cost analysis, and production functions to engineering project decision-making scenarios. (L3)

CO3: Analyse financial statements, cost-volume-profit relationships, and capital budgeting decisions using traditional and AI-powered financial tools. (L4)

CO4: Evaluate investment alternatives using NPV, IRR, and Payback Period methods and assess business viability under uncertainty using scenario analysis. (L5)

CO5: Create a comprehensive mini-project business plan incorporating market analysis, cost estimation, financial projections, and AI-assisted forecasting. (L6)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1							1	1	2	3	1		
CO2							1	1	2	3	1		
CO3							1	2	2	3	1		
CO4							1	2	2	3	1		
CO5							1	3	3	3	1		

Unit I: Introduction to Managerial Economics:

Core Content: Managerial Economics — meaning, nature, scope, and significance for engineers. Economic Analysis: micro vs. macroeconomics; positive vs. normative economics. Demand Analysis — concept, determinants, demand function, law of demand. Elasticity of Demand — types, measurement, business applications. Demand Forecasting — meaning, need, methods.

AI Integration: Generate demand scenarios for an engineering product (e.g., solar panels) and interpret price elasticity, implement moving average and exponential smoothing for a sample sales dataset, use publicly available CMIE or RBI data; apply AI-assisted regression toforecast demand.

Free/Open-Source AI Tools: ChatGPT / Gemini (free tiers); Google Sheets + FORECAST function (free); Python (pandas, scikit-learn) on Google Colab (free); Wolfram Alpha (free

Unit II: Production and Cost Analysis:

Core Content: Production concepts and modern manufacturing systems. Production function — isoquants and isocosts; least-cost combination of inputs. Cost concepts — fixed, variable, total, average, marginal costs. Break-Even Analysis — concept, assumptions, applications; determination of Break-Even Point; margin of safety and profit planning (simple problems).

AI Integration: Build a BEA spreadsheet with dynamic charts, generate a cost function for a manufacturing scenario and identify optimal production level, collect cost data of a local small enterprise; perform AI-assisted CVP analysis and present findings, visualise cost curves.

Free/Open-Source AI Tools: Google Sheets (free); LibreOffice Calc (free); Python (Plotly, pandas) on Google Colab (free); ChatGPT / Gemini (free tiers); Wolfram Alpha (free)

Unit III: Business Organisations and Market Structures:

Core Content: Forms of Business Organisations — sole proprietary, partnership, joint stock companies, startups and digital enterprises. Public vs. private sector in digital economy. Market Structures — classification of markets with real-world examples. Pricing methods and strategies.

AI Integration: Simulate a duopoly pricing game and find Nash Equilibrium with AI assistance, Analyse pricing strategies of three competing tech products using online data and AI classification, how do AI algorithms determine prices and dynamic pricing in e-commerce.

Free/Open-Source AI Tools: ChatGPT / Gemini (free tiers); Google Sheets (free); Python (pandas, Matplotlib) on Google Colab (free)

Unit IV: Capital Budgeting:

Core Content: Capital — introduction, meaning, nature, and significance. Capital Budgeting — concept, nature, significance in modern enterprises, role in strategic decision-making. Methods — Payback Method, Accounting Rate of Return (ARR), Net Present Value (NPV), Internal Rate of Return (IRR), Profitability Index (PI).

AI Integration: Evaluate 3 capital investment alternatives, Monte Carlo simulation for capital budgeting risk analysis, AI-based working capital optimization using predictive models, decision tree analysis for investment appraisal.

Free/Open-Source AI Tools: Google Sheets (free); Python (NumPy, Matplotlib, scikit-learn) on Google Colab (free); ChatGPT / Gemini (free tiers); Wolfram Alpha (free)

Unit V: Financial Accounting and Analysis:

Core Content: Accounting principles and digital accounting systems. Journal, ledger, trial balance. Final Accounts — trading account, profit and loss account, balance sheet with simple adjustments. Financial Analysis — liquidity ratios, activity ratios, capital structure ratios, profitability ratios.

AI Integration: Automated financial statement, interpret a company's financial ratios from a pasted balance sheet and income statement, Machine learning for financial ratio prediction and company health assessment, download annual report of an Indian engineering company (BHEL, L&T, Infosys) and perform ratio analysis.

Free/Open-Source AI Tools: Python (pandas, Matplotlib) on Google Colab (free); Google Sheets (free); ChatGPT / Gemini (free tiers); Wolfram Alpha (free)

Text Books:

1. Mithani D.M., Managerial Economics: Theory and Applications, Himalaya Publishing House, 10th Edition, 2025.
2. Pandey I.M., Financial Management, Vikas Publishing House, 12th Edition, 2022.

Reference Books:

1. Aryasri A.R., Managerial Economics and Financial Analysis, McGraw-Hill Education, Revised Edition, 2020.
2. Prasanna Chandra, Financial Management: Theory and Practice, Tata McGraw-Hill, 10th Edition, 2019.
3. Gupta G.S., Economics for Engineers, Tata McGraw-Hill Education, 2nd Edition, 2017.

Web Resources & NPTEL Links:

- NPTEL: Managerial Economics – https://nptel.ac.in/courses/110105063
- NPTEL: Financial Accounting for Engineers – https://nptel.ac.in/courses/110107114
- Khan Academy – Microeconomics and Finance Basics – https://www.khanacademy.org
- SWAYAM Portal – Business Economics – https://swayam.gov.in

B. Tech. – II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26ES01101T	ES	**Sustainability and Green Engineering (AI Integrated)** (Common to all branches of Engineering)	2	0	0	0	2

Pre Requisites: Basic Environmental Science & Engineering Chemistry.

Course Objectives:

- To understand the concepts of sustainability, carbon cycle, and the environmental impact of engineering activities.
- To evaluate sustainable construction materials, their life cycle, and environmental performance.
- To apply energy calculations for buildings including embodied energy, operational energy, and life cycle energy.
- To assess green building standards, energy codes (ECSBC), and performance rating systems (LEED, GRIHA).
- To integrate AI/ML tools for carbon footprint analysis, energy optimization, and sustainable design decision-making.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyze the principles of sustainability, Sustainable Development Goals (SDGs), carbon cycle, and greenhouse gas accounting to assess their interrelationships and environmental impacts. (L4)

CO2: Apply knowledge of sustainable construction materials including SCMs, recycled aggregates, geopolymer concrete, and alternative binders to evaluate their environmental performance. (L3)

CO3: Analyze the embodied energy and operational energy of building systems, perform energy calculations, and compare environmental performance of design alternatives. (L4)

CO4: Evaluate green building rating systems (LEED, GRIHA, ECBC), assess compliance with energy codes, and compare sustainability performance of different building design strategies. (L5)

CO5: Design sustainable civil engineering solutions integrating circular economy principles, C&D waste management, green infrastructure, and AI-driven optimization tools. (L6)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	1				3	2				2	1	1
CO2	2	2	2	2	2	2	2				2	2	1
CO3	2	2	2	2	3	2	1				2	2	2
CO4	2	2	3	2	3	3	2				3	2	2
CO5	2	3	3	3	3	3	2				3	2	2

Unit I: Fundamentals of Sustainability & Carbon Cycle:

Core Content: Definition of sustainability: three pillars (economic, social, environmental), Brundtland definition. Sustainable Development Goals (SDGs): SDGs 6, 7, 9, 11, 12, 13. Carbon cycle: biotic and abiotic components. Greenhouse gases: CO_2, CH_4, N_2O, HFCs. Carbon dioxide equivalent (CO_2e). CO_2 contribution from cement production (~8% global emissions), steel, glass, aluminium. Carbon footprint: definition, calculation methodology, carbon accounting. Climate change: global warming, sea level rise, extreme events.

AI Integration: AI for carbon footprint tracking: overview of ML-based life cycle assessment tools. Python-based carbon footprint calculator: computing embodied carbon from material quantities.

Introduction to climate modelling data processing using Python (pandas, matplotlib). Online tools: Global Carbon Atlas, openLCA (free) for life cycle assessment.

Free/Open-Source AI Tools: openLCA (free, open-source LCA software); Python (Pandas, Matplotlib) – Google Colab; Global Carbon Project tools; SimaPro (student trial)

Unit II: Sustainable Construction Materials:

Core Content: Construction materials and indoor air quality: VOCs, formaldehyde. No/low cement concrete: supplementary cementitious materials (fly ash, GGBS, silica fume, rice husk ash) — properties, mix proportions. Recycled aggregate concrete: sources, properties, durability. Alternative binders: alkali-activated materials (AAM), geopolymer concrete. Manufactured sand (M-sand): processing, quality. Bamboo, timber, and fibre-reinforced composites. Durability and service life assessment. Life cycle of construction materials.

AI Integration: ML for sustainable material selection: multi-criteria decision analysis using Python. AI-based prediction of concrete strength from mix proportions using regression models (scikit-learn). Computer vision for recycled aggregate quality assessment. Introduction to materials informatics: using ML to discover new sustainable binders. Comparative LCA using openLCA.

Free/Open-Source AI Tools: scikit-learn (free Python ML library); openLCA (free); Python (NumPy, scikit-learn) – Google Colab; Concrete Mixture Proportioner (free online NRMCA tool

Unit III: Embodied Energy & Operational Energy in Buildings:

Core Content: Definition of embodied energy: extraction, processing, manufacturing, transport, construction, demolition. Operational energy: heating, cooling, lighting, hot water. Life cycle energy: sum of embodied and operational energy. Calculation of embodied energy for common construction materials (energy intensity values). Embodied energy calculation for a simple building structure. Energy balance: reducing operational energy vs. increasing insulation (embodied energy trade-off).

AI Integration: Python-based embodied energy calculator for building elements. AI for building energy optimization: overview of Energy Plus (free) and Open Studio (free) simulations. ML for operational energy prediction from building parameters (Linear Regression, Random Forest using scikit-learn). Digital twin for energy monitoring: concept and tools. Building energy benchmarking using ML: comparison with national average.

Free/Open-Source AI Tools: Energy Plus (free, US DOE); Open Studio (free); Python (scikit-learn, NumPy) on Google Colab (free); openLCA (free); AI tools: ChatGPT / Gemini

Unit IV: Green Building Rating Systems & Energy Codes:

Core Content: Energy Conservation and Sustainability Building Code (ECSBC-2023): scope, mandatory and prescriptive requirements, OTTV calculations, compliance path. Green building rating systems: LEED — credit categories, certification levels; GRIHA — India's national system; IGBC rating systems; BEE star rating. Passive design strategies: building orientation, natural ventilation, daylighting, cool roofs. Net Zero Energy Buildings (NZEB): definition, pathways.

AI Integration: AI for automated LEED/GRIHA compliance checking: overview of existing tools. ML for building performance prediction: predicting LEED score from building design parameters. BIM-based energy analysis: linking Revit model to energy simulation. Smart building systems: AI-driven HVAC optimization. IoT sensors for real-time energy monitoring. Case study: AI-optimized green building design.

Free/Open-Source AI Tools: Energy Plus (free); Open Studio (free); Autodesk Revit student version with Energy Analysis add-in (free); Python for data analysis (Google Colab)

Unit V: Circular Economy, Waste Management & Environmental Sustainability:

Core Content: Circular economy principles: reduce, reuse, recycle, recover, redesign — waste hierarchy. Construction and demolition (C&D) waste: types, quantities, recycling potential. C&D waste management rules 2016. Deconstruction vs. demolition. Industrial waste in construction: fly ash, blast furnace slag, quarry dust, plastic waste utilisation. Water recycling in construction. Green procurement policies. Carbon sequestration potential of green infrastructure: urban trees, green roofs, wetlands. Acid rain: causes, effects, prevention.

AI Integration: AI for construction waste prediction: ML models for waste generation estimation. Computer vision for C&D waste sorting (overview of automated sorting robots). Digital material passport: BIM-based end-of-life material tracking. ML for demolition waste characterization. Python for waste audit data analysis. Sustainable infrastructure planning using AI and GIS: overview of green infrastructure planning tools.

Free/Open-Source AI Tools: Python (scikit-learn, Pandas) – Google Colab; QGIS (free GIS for green infrastructure planning); OpenLCA (free LCA); i-Tree (free urban tree carbon quantification tool by USDA)

Text Books:

1. Kibert C.J., Sustainable Construction: Green Building Design & Delivery, Wiley, 4th Edition, 2016.
2. Goodhew S., Sustainable Construction Processes, Wiley-Blackwell, 2016.

Reference Books:

1. ECSBC 2024 (Energy Conservation and Sustainability Building Code), Bureau of Energy Efficiency, Government of India.
2. Goswami D., Kreith F. and Kreider J., Principles of Solar Engineering, Taylor & Francis, 2000.

Web Resources & NPTEL Links:

- NPTEL: Sustainable Development – https://nptel.ac.in/courses/105105157
- Bureau of Energy Efficiency (BEE), India – https://beeindia.gov.in
- EnergyPlus (free building energy simulation) – https://energyplus.net
- openLCA (free LCA software) – https://www.openlca.org
  QGIS (free GIS) – https://qgis.org

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26EC04201T	EC	Analog Circuits	2	1	0	0	3

Pre-Requisites: Engineering Physics and Electrical circuits.

Course Objectives:

- To understand PN junction diode, rectifier circuits, Zener regulation, and BJT biasing methods with Q-point analysis.
- To understand negative feedback amplifier topologies and study RC phase shift, Wien bridge, and crystal oscillators using the Barkhausen criterion.
- To understand ideal and practical Op-Amp characteristics and analyse inverting, non-inverting, integrator, differentiator, and log/antilog amplifier circuits.
- To study comparators, waveform generators, 555 timer monostable/astable circuits, Schmitt trigger, and PLL block operation and applications.
- To understand weighted resistor and R-2R ladder DAC techniques and evaluate flash, counter-type, SAR, and dual slope ADC types for speed, resolution, and accuracy.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyse diode and Zener characteristics, evaluate rectifier performance, and design BJT bias circuits for Q-point stability. (L3)

CO2: Evaluate negative feedback topologies for gain and impedance, and verify oscillation using Barkhausen criterion for RC phase shift and Wien bridge oscillators. (L5)

CO3: Evaluate 741 Op-Amp characteristics and design inverting, non-inverting, integrator, differentiator, log/antilog, and instrumentation amplifier circuits. (L5)

CO4: Analyse comparator, 555 timer, and Schmitt trigger circuits computing pulse width, frequency, and duty cycle — and explain PLL operation and applications. (L3)

CO5: Compare weighted resistor and R-2R ladder DAC techniques and evaluate flash, counter type, SAR, and dual slope ADC types for speed, resolution, and accuracy. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2			2		1				1	2	2
CO2	3	3	2		2		1				1	2	2
CO3	2	3	3	2	3		1				2	3	3
CO4	2	2	3	3	3		1				2	2	2
CO5	2	2	3	3	3		1				3	2	3

Unit I: PN Junction Diode, Applications & BJT:

Core Content: PN junction diode, VI Characteristics, Diode resistance, Transition and Diffusion Capacitance, Quantitative analysis of Half-wave, Full-wave and Bridge Rectifiers and Capacitor filter, Zener diode as Voltage Regulator. Bipolar Junction Transistors: Characteristics, Transistor as an Amplifier and as a Switch, BJT Configurations, Biasing and Stabilization: Operating Point, DC and AC Load Lines, Importance of Biasing, Self-Bias, Bias Stability, Thermal Runaway, Illustrative problems.

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain how a PN junction diode works in forward and reverse bias; ask it to solve one numerical on the diode current equation and verify manually. Also ask it to compare BJT Configurations.
2. Use Python (Matplotlib) or an online simulator (Falstad / Every Circuit) to plot the diode V-I characteristic and observe how it changes with temperature; then simulate a full-wave bridge rectifier with and without a filter capacitor and note how capacitor value affects ripple voltage.
3. Use an LLM to walk through the DC load line construction and Q-point determination for a given self-bias circuit; verify the operating point (VCE, IC) by solving the bias equations manually for one set of component values.

Free/Open-Source AI Tools: AI/LLM tool; Python (Matplotlib); Falstad; Every Circuit.

Unit II: Feedback Amplifiers & Oscillator Circuits:

Core Content: Feedback Amplifiers: Classification of Amplifiers, the Feedback Concept, General Characteristics of Negative-Feedback Amplifiers, Effect of Negative Feedback upon Output and Input Resistances, Voltage-Series Feedback, Current-Series Feedback, Current-Shunt Feedback, Voltage-Shunt Feedback. Oscillator Circuits: Barkhausen Criterion of oscillation, Oscillator operation, Hartley oscillator, Colpitts oscillator, RC phase shift oscillator, Wien bridge Oscillator, Crystal Oscillator. (Quantitative analysis only for all the feedback amplifiers and oscillators)

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain the four negative feedback topologies (voltage-series, current-series, voltage-shunt, current-shunt); ask it to give one circuit example for each and describe how each changes the input and output resistance — prepare a brief comparison table.
2. Use an AI/LLM tool to explain the Barkhausen criterion; ask it to show why both conditions (loop gain = 1 and phase shift = 0° or 360°) must hold simultaneously, using the RC phase shift oscillator as an example — verify the frequency formula $f = 1/(2\pi\sqrt{6}\ RC)$ for one set of R and C values.
3. Use Python or Falstad (online simulator) to simulate an RC phase shift oscillator for a target frequency of 1 kHz; vary R and C values, observe the output frequency change, and record at least three combinations in a table to verify the formula.

Free/Open-Source AI Tools: AI/LLM tool; Python; Falstad.

Unit III: Operational Amplifiers & Applications:

Core Content: Operational Amplifiers: Introduction, Basic information of Op-Amp, Ideal Operational Amplifier, Block Diagram Representation of Typical Op-Amp, DC and AC characteristics, 741 Op-Amp and its features. Op-Amp Applications: Basic Op-Amp Applications, Instrumentation Amplifier, AC Amplifier, V-to-I and I-to-V Converter, Sample and Hold Circuit, Log and Antilog Amplifier, Multiplier and Divider, Differentiator, Integrator.

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain ideal Op-Amp assumptions and how they break down in the real 741 IC (slew rate, CMRR, finite gain); then ask it to derive the output expression for an inverting amplifier and a summing amplifier using the virtual ground concept — verify by substituting values.
2. Use LTspice or Falstad to simulate an Op-Amp integrator and differentiator with a square wave input; observe the output waveforms and write one paragraph explaining why the output looks the way it does, relating it to the RC time constant.

3. Use Python (NumPy / Matplotlib) to plot the output of a log amplifier (Vout vs. Vin) for inputs from 0.1 V to 2 V; compare the plot with the theoretical expression and identify the voltage range where it holds well — note any deviation at high input voltages.

Free/Open-Source AI Tools: AI/LLM tool; LTspice; Falstad; Python (NumPy, Matplotlib).

Unit IV: Comparators, Waveform Generators, Timers & PLL:

Core Content: Comparators and Waveform Generators: Introduction, Comparator, Square Wave Generator, Monostable Multivibrator, Triangular Wave Generator, Sine Wave Generators. Timers and Phase Locked Loop: Introduction to 555 timer, functional diagram, Monostable and Astable operations and applications, Schmitt Trigger, PLL block schematic, principles and description of individual blocks, 565 PLL, Applications of VCO (566).

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain the 555 timer in monostable and astable modes — ask it to describe how the internal comparators and flip-flop produce the timing waveform and what changes between the two modes; verify the formulas T = 1.1RC (monostable) and f = 1.44/((R1+2R2)C) (astable) for one set of values.
2. Use an online simulator (Falstad / EveryCircuit) or LTspice to build a 555 astable circuit for 1 kHz output; vary R1, R2, and C to change the duty cycle and record at least three combinations in a table showing frequency and duty cycle.
3. Use an AI/LLM tool to explain the Phase Locked Loop (PLL) — describe the role of the phase detector, low-pass filter, and VCO in simple terms; ask it to give one practical application (e.g., FM demodulation or frequency synthesis) with a brief block-level description.

Free/Open-Source AI Tools: AI/LLM tool; Falstad; EveryCircuit; LTspice.

Unit V: DAC and ADC Converters:

Core Content: Digital to Analog and Analog to Digital Converters: Introduction, basic DAC techniques, weighted resistor DAC, R-2R ladder DAC, inverted R-2R DAC. A-D Converters — parallel comparator type ADC, counter type ADC, successive approximation ADC and dual slope ADC. DAC and ADC Specifications.

AI Integration – Suggested Tasks:

1. Use an AI/LLM tool to explain the difference between a weighted resistor DAC and an R-2R ladder DAC; ask it to explain why the R-2R ladder is preferred in practice — verify the output voltage formula for a 4-bit R-2R DAC with Vref = 5 V manually for two input codes.
2. Use an AI/LLM tool to compare the four ADC types (flash, counter-type, successive approximation, dual slope); ask it to fill a comparison table covering conversion speed, hardware complexity, accuracy, and one typical application for each — review and correct the table yourself.
3. Use Python to simulate a 4-bit R-2R ladder DAC: calculate the analog output for all 16 input codes (0000–1111) with Vref = 5 V and plot the staircase output using Matplotlib; then manually trace the SAR algorithm for one input (e.g., 3.2 V) and verify the digital output.

Free/Open-Source AI Tools: AI/LLM tool; Python; Matplotlib.

Instructions:

- The suggested tasks can be modified based on the decision of the subject expert.
- The suggested tasks can be part of assignments using appropriate AI/LLM tools and Python/MATLAB-based simulations, with the choice of platforms and machine learning models decided by the subject expert to enhance conceptual understanding and practical AI-enabled engineering skills.

Textbooks

1. J. Millman, C. Halkias, “Electronic Devices and Circuits”, Tata McGraw-Hill, 2nd Edition, 2010.

2. D. Roy Choudhury, “Linear Integrated Circuits”, New Age International (P) Ltd, 2nd Edition, 2003.

Reference Books

1. Robert L. Boylestad and Louis Nashelsky, “Electronic Devices and Circuit Theory”, Pearson Education, 2021.
2. G. K. Mithal, “Electronic Devices and Circuits”, Khanna Publisher, 23rd Edition, 2017.
3. David Bell, “Electronic Devices and Circuits”, Oxford University Press, 5th Edition, 2008.
4. Malvino, Albert Paul, and David J. Bates, “Electronic Principles”, McGraw-Hill Higher Education, 2007.
5. R. A. Gayakwad, “Operational Amplifiers and Linear Integrated Circuits”, Prentice Hall India, 2002.

Web Resources & NPTEL Links

- NPTEL Course: https://nptel.ac.in/courses/122106025
- NPTEL Course: https://nptel.ac.in/courses/108102112

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02205T	PC	AI Assisted Electrical Circuit Analysis	2	0	2	0	3

Pre-Requisites: Differential Equations & Vector Calculus, AI Tools & Applications.

Course Objectives:

- To introduce the fundamental concepts of electrical circuits and analyze DC and AC circuits using basic network laws and analysis techniques.
- To apply network theorems and resonance concepts for solving complex electrical circuit problems.
- To analyze balanced and unbalanced three-phase circuits and power measurement techniques.
- To study transient behaviour of electrical circuits using differential equation and Laplace transform methods.
- To understand two-port network parameters and apply AI-based computational tools for circuit analysis and optimization.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamental concepts of electrical circuit elements, network laws, resonance, transient response, three-phase circuits, and two-port network parameters used in circuit analysis. (L3)

CO2: Apply Ohm's law, Kirchhoff's laws, network reduction techniques, node and mesh analysis, network theorems, resonance conditions, and transient analysis methods to solve DC and AC electrical circuit problems (L3)

CO3: Analyze the performance of balanced and unbalanced three-phase circuits, resonant circuits, transient responses, and interconnected two-port networks under different operating conditions. (L4)

CO4: Apply and analyze AI and machine learning techniques for automated circuit verification, fault detection, resonance prediction, transient estimation, and optimization of electrical network parameters using open-source simulation tools. (L4)

CO5: Design electrical circuit models and AI-assisted solutions for equivalent circuit estimation, three-phase system monitoring, transient response prediction, and optimization of network configurations using Python, Ng spice, and Scikit-learn. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	1				2	3	2
CO2	3	3	2	2	2	1	1				2	3	2
CO3	3	3	2	2	2	1	1				2	3	3
CO4	3	3	2	3	3	1	1				2	3	2
CO5	3	2	3	2	3	1	1				3	2	3

Unit I: Introduction to Electrical Circuits:

Core Content: Basic Concepts of passive elements of R, L, C and their V-I relations; Sources (dependent and independent); Ohm's Law and its limitations; Kirchhoff's laws; Network reduction techniques (series, parallel, series-parallel, star-to-delta and delta-to-star transformation); source transformation technique; node and mesh analysis. Single Phase Circuits: Characteristics of periodic functions; Average value, R.M.S. value, form factor; representation of a sine function; concept of phasor, phasor diagrams. Steady state analysis of R, L and C circuits to sinusoidal excitations; response of pure resistance, inductance, capacitance, series RL circuit, series RC circuit, series RLC circuit, parallel RL circuit, parallel RC circuit.

AI Integration: AI-Assisted Circuit Analysis and Error Detection: Use AI tools (Python + machine learning) to automatically verify KCL/KVL equations, detect incorrect circuit connections, and assist students in solving node and mesh analysis problems.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Matplotlib) on Google Colab; Ngspice (free); Scikit-learn; Jupyter Notebook*

Unit II: Network Theorems (DC & AC Excitations):

Core Content: Superposition theorem; Thevenin's theorem; Norton's theorem; Maximum Power Transfer theorem; Reciprocity theorem; Millman's theorem; Compensation theorem. Resonance and Locus Diagrams: Series Resonance: Characteristics of a series resonant circuit; Q-factor, selectivity and bandwidth; expression for half power frequencies; Parallel resonance: Q-factor, selectivity and bandwidth; Locus diagram: Series RL, RC, RLC with R, L and C variables.

AI Integration: Machine Learning for Equivalent Circuit Prediction: Use regression or neural network models to estimate Thevenin/Norton equivalents and predict resonance characteristics (Q-factor and bandwidth) from circuit parameters.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Scikit-learn) on Google Colab; Ngspice; Jupyter Notebook*

Unit III: Analysis of Three Phase Balanced Circuits:

Core Content: Phase sequence; star and delta connection of sources and loads; relation between line and phase quantities; analysis of balanced three phase circuits; measurement of active and reactive power. Analysis of three phase unbalanced circuits: Loop method; Star-Delta transformation technique; two wattmeter method for measurement of three phase power.

AI Integration: AI-Based Fault Detection in Three-Phase Systems: Application of machine learning algorithms to detect phase imbalance, faults, and abnormal power consumption patterns in three-phase electrical systems using voltage and current data.

Free/Open-Source AI Tools: *Python (Scikit-learn, Pandas) on Google Colab; Ngspice; Jupyter Notebook*

Unit IV: Transient Analysis:

Core Content: Transient response of R-L, R-C and R-L-C circuits (Series and parallel combinations) for D.C. and sinusoidal excitations; Initial conditions; Solution using differential equation approach and Laplace transform approach.

AI Integration: AI Integration to Transient Analysis: Use regression models to predict voltage and current transients in RL, RC, and RLC circuits after switching.

Free/Open-Source AI Tools: *Python (NumPy, SciPy, Matplotlib) on Google Colab; Ngspice; Jupyter Notebook*

Unit V: Network Parameters:

Core Content: Impedance parameters; Admittance parameters; Hybrid parameters; Transmission (ABCD) parameters; conversion of Parameters from one form to other; Conditions for Reciprocity and Symmetry; Interconnection of Two Port networks in Series, Parallel and Cascaded configurations; problems.

AI Integration: AI Optimization of Two-Port Network Parameters: Use AI optimization techniques (genetic algorithms or neural networks) to determine optimal two-port network parameters.

Free/Open-Source AI Tools: Python (NumPy, SciPy, Scikit-learn) on Google Colab; Ngspice; Scikit-learn; Jupyter Notebook

Text Books:

1.Engineering Circuits Analysis, Jack Kemmerly, William Hayt and Steven Durbin, Tata McGraw Hill Education, 2005, Sixth Edition.

2.Fundamentals of Electric Circuits, Charles K. Alexander, Mathew N. O. Sadiku, 3rd Edition, Tata McGraw-Hill, 2019.

Reference Books:

1. Network Analysis, M. E. Van Valkenburg, 3rd Edition, PHI, 2019.
2. Network Theory, N. C. Jagan and C. Lakshminarayana, 1st Edition, B. S. Publications, 2012.
3. Circuits and Networks Analysis and Synthesis, A. Sudhakar, Shyam Mohan S. Palli, 5th Edition, Tata McGraw-Hill, 2017.
4. Engineering Network Analysis and Filter Design, Durgesh C. Kulshreshtha, Gopal G. Bhise, Prem R. Chadha, Umesh Publications, 2012**.**
5. Circuit Theory: Analysis and Synthesis, A. Chakrabarti, Dhanpat Rai & Co., 2018, 7th Revised Edition.

Web Resources & NPTEL Links:

- NPTEL – Electrical Circuits: https://archive.nptel.ac.in/courses/117/106/117106108/
- NPTEL – Circuit Theory: https://archive.nptel.ac.in/courses/108/105/108105159/
- MIT OpenCourseWare – Circuits and Electronics: https://ocw.mit.edu

Recommended Free & Open-Source AI/Computational Tools:

Python (NumPy, SciPy, Scikit-learn) – FREE on Google Colab for circuit analysis automation

Ngspice (free, open-source) – Circuit simulation

Scikit-learn (free, open-source) – Machine learning models for circuit parameter estimation

TensorFlow / Keras (free, open-source) – Neural network models

Jupyter Notebook (free, open-source) – Interactive computational environment

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02206T	PC	DC Machines and Transformers with Intelligent Control	2	0	2	0	3

Pre-Requisites: Electromagnetic Field Theory, AI Tools & Applications, Electrical Circuits.

Course Objectives:

- To understand the construction, working principle and performance characteristics of DC generators and their practical applications.
- To analyze the operating characteristics and performance of different types of DC motors and their starting mechanisms.
- To study speed control methods and testing procedures of DC machines using analytical and computational approaches.
- To understand the construction, operation and performance characteristics of single-phase transformers.
- To analyze testing methods, efficiency and parallel operation of transformers using modern computational and AI-based techniques.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyze the construction, principle of operation, characteristics, and applications of DC generators, DC motors, and single-phase transformers. (L4)

CO2: Apply the concepts of armature reaction, commutation, speed control methods, transformer equivalent circuits, regulation, and testing procedures. (L3)

CO3: Apply testing methods, performance analysis techniques, and simulation tools for evaluating DC machines and transformers. (L3)

CO4: Analyze the efficiency, losses, regulation, speed characteristics, and operational performance of electrical machines using conventional and AI-based techniques. (L4)

CO5: Develop AI-assisted models for prediction, visualization, and performance evaluation of electrical machines and transformers using modern open-source tools. (L6)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	2	1				2	3	1
CO2	3	3	2	2	2	2	1				2	3	2
CO3	3	3	2	3	3	2	1				2	3	2
CO4	3	2	2	2	3	2	1				2	3	2
CO5	3	3	2	3	3	2	1				3	2	3

Unit I: DC Generators:

Core Content: Construction and principle of operation of DC machines; EMF equation of DC generators; Excitation techniques; Armature reaction and commutation; Methods of improving commutation; Open circuit characteristics and load characteristics of DC generators; Parallel operation of DC generators; Applications of DC generators.

AI Integration: AI-based prediction of generated EMF and load characteristics; AI-assisted analysis of parallel operation performance; Data visualization of generator characteristics; Predictive maintenance of DC generators using sensor data.

Free/Open-Source AI Tools: *Python (NumPy, Pandas, SciPy); Scikit-learn; TensorFlow/PyTorch; GNU Octave/Scilab*

Unit II: DC Motors:

Core Content: DC motor – Back-emf; Generation of torque; Torque equation; Performance characteristics of different types of motors; Necessity of starters; 3-point and 4-point starters; Losses in DC machines; Applications of DC Motors; numerical problems.

AI Integration: AI models for predicting torque and speed characteristics; Intelligent starter protection and monitoring; Machine learning models for efficiency estimation; Intelligent control systems for motor performance optimization.

Free/Open-Source AI Tools: *Python (SciPy, NumPy); Scikit-learn; OpenModelica; TensorFlow*

Unit III: Speed Control and Testing of DC Machines:

Core Content: Speed control of DC Machines: Armature control and Flux control methods; Testing of DC machines: Brake test; Swinburne's test; Hopkinson's test; Fields test; Retardation test.

AI Integration: AI-assisted speed control optimization of DC motors; AI-assisted virtual experimentation and result analysis; AI-based performance testing and fault prediction; Simulation of DC motor speed control methods.

Free/Open-Source AI Tools: *Python (Pandas, Matplotlib); Scikit-learn; OpenModelica; Jupyter Notebook*

Unit IV: Single-Phase Transformers:

Core Content: Introduction to single-phase Transformers (Construction and principle of operation); EMF equation; operation on no-load and on load; lagging, leading and unity power factors loads; phasor diagrams; equivalent circuit; regulation; losses and efficiency; effect of variation of frequency and supply voltage on losses; all day efficiency; Applications.

AI Integration: Machine learning prediction of transformer efficiency and losses; AI-based load forecasting for transformer operation; Neural networks for transformer voltage regulation prediction; Visualization of transformer operating characteristics.

Free/Open-Source AI Tools: *Python (NumPy, SciPy); TensorFlow/PyTorch; OpenModelica; ParaView*

Unit V: Testing of Transformers:

Core Content: Open Circuit and Short Circuit tests; Sumpner's test; separation of losses; Parallel operation with equal and unequal voltage ratios; auto transformer; equivalent circuit; comparison with two winding transformers; Scott connection; off load and on load tap changers.

AI Integration: Data analytics for separation of hysteresis and eddy current losses; AI-assisted modeling of load variation and all-day efficiency; Data-driven models for loss separation and efficiency prediction; AI optimization of transformer load sharing.

Free/Open-Source AI Tools: *Python (Scikit-learn, Pandas); TensorFlow; OpenModelica; Elmer FEM*

Text books:

1. Electrical Machinery by Dr. P S Bimbhra, 7th Edition, Khanna Publishers, New Delhi, 1995.
2. Performance and Analysis of AC Machines by M.G. Say, CBS, 2002.

Reference Books:

1. Electrical Machines by D. P. Kothari, I. J. Nagarth, McGraw Hill Publications, 5th Edition.
2. Electrical Machinery Fundamentals by Stephen J Chapman, McGraw Hill Education, 2011.
3. Generalized Theory of Electrical Machines by Dr. P S Bimbhra, 7th Edition, Khanna Publishers, 2021.
4. Theory & Performance of Electrical Machines by J. B. Gupta, S. K. Kataria & Sons, 2007.
5. Electric Machinery by Fitzgerald, A. E. Kingsley Jr. C., & Umans S. D., 7th Edition, McGraw-Hill Education, 2014.

Web Resources & NPTEL Links:

- NPTEL – Electrical Machines I: https://nptel.ac.in/courses/108105017
- NPTEL – Transformer Course: https://nptel.ac.in/courses/108105155
- MIT OpenCourseWare – Electric Machines: https://ocw.mit.edu/courses/electric-machines/
- Python for Electrical Engineering: https://scipy.org
- OpenModelica Simulation: https://openmodelica.org

Recommended Free & Open-Source AI/Computational Tools:

Python (NumPy, Pandas, SciPy) – FREE on Google Colab for machine analysis
Scikit-learn (free, open-source) – Machine learning models
TensorFlow / PyTorch (free, open-source) – Predictive modelling
GNU Octave / Scilab (free) – Electrical machine simulation
OpenModelica (free, open-source) – Electrical machine simulation
Elmer FEM (free, open-source) – Finite element modelling

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26EC04201P	EC	Analog Circuits Lab	0	0	0	2	1

Pre-Requisites: Analog Circuits.

Course Objectives:

- To understand the characteristics and operation of semiconductor devices, rectifiers, transistor configurations, oscillators, timers, Op-Amp circuits, and DACs using laboratory experiments.
- To explain the working principles of electronic circuits and interpret experimental waveforms, biasing conditions, oscillations, amplification, and digital-to-analog conversion processes.
- To apply electronic circuit theory for designing, constructing, and testing analog electronic circuits using BJTs, Op-Amps, 555 timers, oscillators, and DACs.
- To analyse the performance of electronic circuits through measurements, graphical interpretation, comparison of theoretical and practical results, and simulation using AI/LLM, Python, or MATLAB tools.
- To design and evaluate AI-assisted and simulation-based electronic system models for prediction, optimisation, regression analysis, and validation of laboratory experiments.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the characteristics and operation of semiconductor devices, rectifiers, and transistor configurations through laboratory experiments. (L4)

CO2: Apply electronic circuit concepts to construct and test oscillators, multivibrators, and Op-Amp-based applications. (L4)

CO3: Utilize AI/LLM tools along with Python/MATLAB simulations to interpret electronic circuit behavior, estimate parameters, and validate theoretical models. (L4)

CO4: Analyze the behavior of RLC circuits during resonance and transient conditions through simulation. (L4)

CO5: Design and evaluate AI-assisted predictive models, regression-based analysis systems, and Simulation frameworks for electronic circuits and applications. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2		2	2		3	3	3		1	3	1
CO2	3	3	3	2	3		3	3	3		1	2	2
CO3	2	3	3	3	3		3	3	3		2	2	2
CO4	2	3	2	3	3		3	3	3		3	3	2
CO5	2	2	3	3	3		3	3	3		3	2	3

List of Experiments:

Implement / execute any 12 of the following experiments:

1. Plot V–I characteristics of a PN Junction diode and determine static and dynamic resistance.
2. Plot V–I characteristics of a Zener diode and verify its functionality as a voltage regulator.

3. Implement half-wave and full-wave bridge rectifiers with and without filter; measure ripple factor and rectification efficiency.
4. Verify the input and output characteristics of BJT in CE configuration; determine β, find Q-point on DC load line, and identify active, saturation, and cut-off regions.
5. Verify the input and output characteristics of BJT in CB configuration; determine α and compare with CE results.
6. Design and analyse a voltage-divider (self-bias) circuit using BJT; verify the Q-point (VCE, IC) experimentally and compare with the designed values.
7. Construct and test Hartley oscillator; measure the frequency of oscillation, verify against the theoretical formula, and observe sustained sinusoidal output on CRO.
8. Construct and test Colpitts oscillators; measure the frequency of oscillation, verify against the theoretical formula, and observe sustained sinusoidal output on CRO.
9. Design and observe the output of an RC Phase Shift Oscillator using transistor or Op-Amp; verify $f = 1/(2\pi\sqrt{6}\ RC)$ and measure the phase shift across each RC stage.
10. Construct an Astable Multivibrator using 555 timer; design for a specified frequency and duty cycle using $f = 1.44/((R1+2R2)C)$; verify on CRO and vary R2 to observe duty cycle change.
11. Design and verify a Monostable Multivibrator using 555 timer; design for a specified pulse width $T = 1.1RC$; trigger with external pulse and verify output pulse width on CRO.
12. Design and implement Op-Amp applications — inverting summing amplifier (adder), difference amplifier (subtractor), and open-loop comparator using IC 741; verify output for given inputs.
13. Design and verify Op-Amp Integrator and Differentiator circuits using IC 741; apply square wave to integrator (observe triangular output) and triangular wave to differentiator (observe square output).
14. Construct and verify an R-2R Ladder DAC; apply all 16 binary input combinations (0000–1111) and measure analog output; verify step size = $Vref/2^n$ and plot staircase output.

AI Integration – AI/LLM and/or Python/MATLAB Tasks

Any six experiments need to be implemented using appropriate AI/LLM tools and/or Python/MATLAB based simulations:

1. (a) Use an AI/LLM tool to interpret the forward bias, reverse bias, and breakdown regions of the V-I curve and explain the significance of static resistance, dynamic resistance, and reverse saturation current. (b) Train a regression model to fit the Shockley diode equation $I = I_0(e^{(V/\eta VT)} - 1)$ to the measured V-I data and estimate the saturation current and ideality factor.
2. (a) Use an AI/LLM tool to explain filter action and ripple reduction in rectifiers, describe the effect of capacitor value on ripple, and derive expressions for ripple factor and rectification efficiency. (b) Use Python/MATLAB simulation to predict ripple voltage for different capacitor values; apply regression to model ripple factor as a function of load resistance and capacitance.
3. (a) Use an AI/LLM tool to explain the physical significance of each region (active, saturation, cut-off) of the CE characteristics, the concept of current gain β, and the Early effect. (b) Train a linear regression model in Python to estimate β from the measured IC vs. IB data and evaluate model accuracy against graphically extracted values.
4. (a) Use an AI/LLM tool to reason about the relationship between α and β, explain the low input impedance and near-unity current gain of CB configuration, and compare it with CE. (b) Use Python regression to predict IC from IB using the measured CB characteristic data and evaluate model accuracy.
5. (a) Use an AI/LLM tool to explain the voltage divider bias technique, the DC load line concept, the role of each component (R1, R2, RC, RE) in bias stabilisation, and how the Q-point ensures linear

amplifier operation. (b) Build a Python regression-based Q-point predictor that takes R1, R2, RC, RE, and VCC as inputs and predicts VCE and IC.

6. (a) Use an AI/LLM tool to explain why the CC configuration has near-unity voltage gain and high input impedance, describe the buffer action, and compare its characteristics with the CE amplifier. (b) Use Python to simulate the emitter follower frequency response and compare gain, input impedance, and output impedance with the CE amplifier using measured data.
7. (a) Use an AI/LLM tool to explain the Barkhausen criterion, the role of the LC tank circuit in sustaining oscillations, and derive the theoretical oscillation frequency formula for both Hartley and Colpitts configurations. (b) Use Python to compute and compare theoretical vs. measured oscillation frequencies for both oscillators and build a regression model to predict frequency from L and C values.
8. (a) Use an AI/LLM tool to explain the 180° phase shift condition across the three RC stages, the loop gain requirement for sustained oscillations, and derive the frequency formula $f = 1/(2\pi\sqrt{6}\ RC)$. (b) Use Python to plot the expected oscillation frequency vs. RC values and verify the measured result against the theoretical curve; train a regression model to predict frequency from R and C values.
9. (a) Use an AI/LLM tool to explain the internal working of the 555 timer in astable mode — including the comparators, flip-flop, threshold/trigger levels — and derive the frequency and duty cycle formulas. (b) Use Python to plot frequency and duty cycle as a function of R2 for a fixed C and compare with measured CRO values; apply regression to predict frequency from component values.
10. (a) Use an AI/LLM tool to explain the difference between monostable and astable modes in the 555 timer and derive the pulse width formula T = 1.1RC by analysing the capacitor charging curve. (b) Use Python to simulate the capacitor charging curve and verify the derived pulse width formula T = 1.1RC; train a regression model to predict pulse width from R and C values.
11. (a) Use an AI/LLM tool to derive and explain the output expressions for the inverting summing amplifier and difference amplifier using the virtual ground concept, and describe open-loop comparator operation. (b) Use Python to simulate output voltage vs. input combinations for all three circuits and verify all cases of the comparator truth table; apply regression to validate amplifier gain accuracy.
12. (a) Use an AI/LLM tool to explain the mathematical basis of integration and differentiation performed by Op-Amp RC circuits and relate the square/triangular waveform transformation to the RC time constant. (b) Use Python (NumPy) to numerically integrate and differentiate the input waveforms and compare the results with CRO observations; apply regression to estimate the RC time constant from the output slope.
13. (a) Use an AI/LLM tool to explain the principle of binary-weighted voltage addition in the R-2R network, derive the output voltage expression $V_{out} = (V_{ref}/2^n) \times$ digital code, and discuss quantisation error. (b) Use Python to simulate DAC output for all 16 input codes, plot the staircase waveform, and calculate percentage error between ideal and measured outputs; apply regression to model linearity error.

Instructions:

- The suggested AI/LLM and/or Python/MATLAB tasks can be modified based on the decision of the subject expert.

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02205P	PC	AI Assisted Electrical Circuit Analysis Lab	0	0	0	3	1.5

Pre-Requisites: AI Assisted Electrical Circuit Analysis.

Course Objectives:

- To perform experiments on fundamental electrical circuits to understand circuit laws, theorems, and network parameters.
- To analyze resonance, transient response, and two-port networks using experimental and computational techniques.
- To apply modern simulation and open-source software tools for circuit modelling and verification.
- To integrate Artificial Intelligence and data-driven methods for analyzing circuit behaviour and predicting responses.
- To develop teamwork, technical documentation, and experimental investigation skills.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Implement the principles of resonance, network theorems, transient response, two-port network parameters, inductance, coupling coefficient, and power measurement in AC/DC electrical circuits. (L3)

CO2: Apply experimental methods to verify network theorems, determine circuit parameters, analyze resonance conditions, measure active/reactive power, and evaluate transient responses in electrical networks. (L3)

CO3: Analyze the behaviour of RLC circuits, locus diagrams, coupled circuits, balanced and unbalanced loads, and interconnected two-port networks using measured and simulated data. (L4)

CO4: Apply and analyze AI and machine learning techniques such as regression models, neural networks, symbolic AI tools, and LSTM models for prediction, theorem verification, transient analysis, and parameter estimation in electrical circuits. (L4)

CO5: Design AI-assisted experimental models and intelligent electrical circuit analysis systems for automated response prediction, theorem verification, resonance estimation, and two -port network characterization using simulation and measured data. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	2	1		3	3	3		1	3	2
CO2	3	3	1	3	2		3	3	3		1	2	2
CO3	2	2	2	2	3		3	3	3		2	3	2
CO4	2	3	2	2	3		3	3	3		3	3	2
CO5	1	2	1	2	2		3	3	3		2	2	3

List of Experiments:

Any 10 of the following experiments are to be conducted:

1. Determination of self, mutual inductances, and coefficient of coupling
2. Series and parallel resonance
3. Locus diagrams of R-L (L Variable) and R-C (C Variable) series circuits
4. Verification of Superposition theorem using AC Excitation
5. Verification of Thevenin's and Norton's Theorems with DC Excitation
6. Verification of Maximum power transfer theorem with DC & AC Excitation
7. Verification of Compensation theorem with DC Excitation
8. Verification of Reciprocity Theorem and Millman's Theorem using DC & AC Excitation
9. Determination of Z and Y parameters
10. Determination of ABCD and Hybrid parameters
11. Series, Parallel and Cascaded connection of two port networks
12. Transient Response of RLC Series & Parallel circuits
13. Measurement of Active Power and Reactive Power for Balanced loads
14. Measurement of Active Power and Reactive Power for Unbalanced loads

AI Integration – Minimum two of the following experiments are to be conducted:

1. Use machine learning regression models to predict inductance and coupling coefficient from experimental voltage/current data.
2. AI-based estimation of resonant frequency and Q-factor.
3. AI-based prediction of circuit responses using neural networks.
4. Automatic theorem verification using symbolic AI tools.
5. AI-generated practice circuits and solution verification.
6. Machine learning models to estimate two-port parameters from measured data.

References:

1. Engineering Circuits Analysis, Jack Kemmerly, William Hayt and Steven Durbin, Tata McGraw Hill Education, 2005, Sixth Edition.
2. Network Analysis, M. E. Van Valkenburg, Pearson Education, 2019, Revised Third Edition.

Online Learning Resources / Virtual Labs & Open-Source Software:

Open-Source Software:

- Python (NumPy, SciPy, Pandas)
- Scikit-learn for regression models
- LTspice for circuit simulation
- Jupyter Notebook

Open-Source Course Links (for AI + Circuits Learning):

- MIT OpenCourseWare – Circuits and Electronics: https://ocw.mit.edu
- Coursera – AI for Everyone / Machine Learning
- edX – Electrical Engineering and AI courses
- NPTEL – Electrical Circuits and Machine Learning

- Fast.ai – Practical Deep Learning

B.Tech.–II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02206P	PC	Intelligent Control of DC Machines and Transformers Lab	0	0	0	3	1.5

Pre-Requisites: DC Machines and Transformers with Intelligent Control.

Course Objectives:

- To perform experimental investigations on DC machines and transformers to understand their operating characteristics.
- To analyze experimental data for determining efficiency, losses and performance of electrical machines.
- To study different testing methods used for evaluating the performance of DC machines and transformers.
- To utilize computational tools and data visualization techniques for analyzing experimental results.
- To apply AI-based tools for intelligent monitoring, performance prediction and fault analysis of electrical machines.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyze the construction, operation, testing procedures, and characteristics of DC machines and transformers. (L3)

CO2: Understand and apply various testing methods for determining efficiency, regulation, speed control, and performance characteristics of electrical machines. (L3)

CO3: Apply experimental procedures and data acquisition techniques for analyzing machine and transformer performance. (L4)

CO4: Analyze performance curves, losses, efficiency, and operational behavior of electrical machines using conventional and AI-based methods. (L4)

CO5: Design models for performance prediction, maintenance, and visualization of electrical machine data using conventional and AI techniques. (L6)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2		1	2	2	3	3	3		1	3	2
CO2	3	3	1	2	2	2	3	3	3		1	3	2
CO3	3	3	2	2	3	2	3	3	3		1	3	2
CO4	3	3	2	3	3	2	3	3	3		1	2	2
CO5	3	3	3	3	3	2	3	3	3		1	2	3

List of Experiments:

Any 10 of the following experiments are to be conducted:

1. Speed control of DC shunt motor by Field Current and Armature Voltage Control.
2. Brake test on DC shunt motor – Determination of performance curves.
3. Swinburne's test – Predetermination of efficiencies as DC Generator and Motor.
4. Hopkinson's test on DC shunt Machines.
5. Load test on DC compound generator – Determination of characteristics.
6. Load test on DC shunt generator – Determination of characteristics.

7. Fields test on DC series machines – Determination of efficiency.
8. Brake test on DC compound motor – Determination of performance curves.
9. OC & SC tests on single phase transformer.
10. Sumpner's test on single phase transformer.
11. Scott connection of transformers.
12. Parallel operation of Single-phase Transformers.
13. Separation of core losses of a single-phase transformer.

AI Integration – Minimum two of the following experiments are to be conducted:

1. Machine learning models for predicting efficiency and performance of DC machines.
2. AI-based speed-torque curve prediction for motors.
3. Predictive analytics for DC machine performance and energy losses.
4. Neural network-based estimation of transformer losses and efficiency.
5. Data visualization of experimental results using AI tools.
6. AI-assisted load sharing of transformer.

References:

• Virtual Lab (IITR): https://ems-iitr.vlabs.ac.in/List%20of%20experiments.html

Online Learning Resources / Virtual Labs & Open-Source Software:

Open-Source Software for AI Applications:

- Python (NumPy, Pandas, Matplotlib)
- Scikit-Learn
- TensorFlow / PyTorch
- Jupyter Notebook
- GNU Octave / Scilab
- OpenModelica
- Elmer FEM

B. Tech. – II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26SE05201S	SE	C Programming and Data Structures	1	0	0	2	2

Pre-Requisites: Basic knowledge of computer fundamentals, programming concepts, algorithms, flowcharts, and problem-solving techniques.

Course Objectives:

- To understand the fundamentals of C programming language, data types, operators, control statements and arrays.
- To explain functions, recursion, pointers, storage classes, strings, structures, unions, and command line arguments in C programming.
- To apply stack and queue concepts for solving computational problems using arrays and linked representations.
- To analyze linked lists, trees, graphs, searching, and sorting techniques for efficient data organization and processing.
- To design and develop C programs using suitable data structures and algorithms for real-world engineering applications.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamentals of C language, data types, operators, control statements, arrays, and basic programming constructs to develop simple C programs. (L3)

CO2: Apply functions, recursion, pointers, storage classes, strings, structures, unions, and command line arguments in C programs. (L3)

CO3: Apply stack and queue operations to solve problems involving expression evaluation and data processing. (L3)

CO4: Analyze linked lists, trees, and graph traversal techniques for efficient implementation of dynamic data structures. (L4)

CO5: Design and develop efficient searching and sorting programs using appropriate algorithms and data structures in C. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2			2		3	3	3			1	1
CO2	3	3	1		2		3	3	3			3	2
CO3	3	3	2	1	3		3	3	3			3	2
CO4	3	3	2	2	3		3	3	3			2	2
CO5	3	3	3	2	3		3	3	3			2	3

Unit I: Introduction to C Language:

Core Content: C language elements, variable declarations and data types, operators and expressions, decision statements – If and switch statements, loop control statements – while, for, do-while statements, arrays.

Unit II: Functions, Pointers and Storage Classes:

Core Content: Functions: types of functions – Recursion and argument passing – pointers – storage allocation – pointers to functions – expressions involving pointers – Storage classes: auto, register, static, extern – Structures – Unions – Strings – string handling functions and Command line arguments.

Experiments:

1. Write C programs that use both recursive and non-recursive functions: (i) to find the factorial of a given integer; (ii) to find the GCD (greatest common divisor) of two given integers.
2. Write a C program to find both the largest and smallest number in a list of integers.
3. Write a C program that uses functions to insert a sub-string into a given main string from a given position.
4. Write a C program that uses functions to delete n characters from a given position in a given string.
5. Write a C program that displays the position or index in the string S where the string T begins, or –1 if S doesn't contain T.
6. Write a C program to count the lines, words and characters in a given text.
7. Write a C program to demonstrate the following parameter passing mechanisms: (i) call-by-value; (ii) call-by-reference.

Unit III: Stacks and Queues:

Core Content: Data Structures: Overview of data structures – stacks and queues – representation of a stack – stack related terms – operations on a stack – implementation of a stack – evaluation of arithmetic expressions – infix, prefix, and postfix notations – evaluation of postfix expression – conversion of expression from infix to postfix – recursion – queues – various positions of queue – representation of queue – insertion, deletion and searching operations.

Unit IV: Linked Lists:

Core Content: Singly linked list – dynamically linked stacks and queues – polynomials using singly linked lists – using circularly linked lists – insertion, deletion and searching operations – doubly linked lists and its operations – circular linked lists and its operations.

Experiments:

1. Write C programs that implement stack using arrays.
2. Write C programs that implement Queue using arrays.
3. Write a C program that uses Stack operations to evaluate postfix expression.
4. Write a C program that uses functions to perform the following operations on singly linked list: (i) Creation; (ii) Insertion; (iii) Deletion.
5. Write a C program that uses functions to create a binary tree of integers.
6. Write C programs for demonstration of linear search.
7. Write C programs for demonstration of binary search.
8. Write a C program for demonstration of bubble sort.
9. Write a C program for demonstration of selection sort.

Unit V: Trees, Graphs, Searching and Sorting

Core Content: Trees: Tree terminology – representation; Binary trees – representation, binary tree traversals, binary tree operations; Graphs – graph terminology, graph representation, elementary graph operations – Breadth First Search (BFS) and Depth First Search (DFS) – connected components – spanning trees; Searching and Sorting: sequential search – binary search – exchange (bubble) sort – selection sort – insertion sort.

Text books:

1. The C Programming Language, Brian W. Kernighan and Dennis M. Ritchie, Second Edition, Prentice Hall Publication.

2. Fundamentals of Data Structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Computer Science Press.
3. Programming in C and Data Structures, J.R. Hanly, Ashok N. Kamthane and A. Ananda Rao, Pearson Education.
4. Richard F. Gilberg & Behrouz A. Forouzan, "Data Structures: A Pseudocode Approach with C", Second Edition, CENGAGE Learning, 2011.

Reference Books:

1. E. Balaguruswamy, "C and Data Structures", 4th Edition, Tata Mc Graw Hill.
2. A.K. Sharma, Computer Fundamentals and Programming in C, 2nd Edition, University Press.
3. M.T. Somashekara, "Problem Solving Using C", PHI, 2nd Edition, 2009**.**

Web Resources & NPTEL Links:

- NPTEL Course (Preview): https://onlinecourses.nptel.ac.in/noc26_cs53/preview
- NPTEL Course: https://nptel.ac.in/courses/106106127
- Programiz – C Programming: https://www.programiz.com/c-programming
- Programiz – DSA: https://www.programiz.com/dsa
- TutorialsPoint – C Programming: https://www.tutorialspoint.com/cprogramming/
- TutorialsPoint - Data Structures & Algorithms: https://www.tutorialspoint.com/data_structures_algorithms/

B. Tech. – II Year I Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26BSHB214A	MC	**Environmental Science** (Common to all branches of Engineering)	2	0	0	0	0

Pre Requisites: Basic knowledge of environmental science, ecology, natural resources, biodiversity, pollution, sustainable development, and elementary computer/IT concepts.

Course Objectives:

- To create awareness about environmental issues, natural resource management, and the need for sustainable development.
- To explain the structure and function of various ecosystems and the significance of biodiversity and its conservation.
- To describe causes, effects, and control of various types of pollution and principles of solid waste management.
- To examine social, ethical, and legislative issues related to environment and sustainable development in India.
- To understand human population dynamics, environmental health linkages, and the role of IT and AI tools in environmental monitoring

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the multidisciplinary concepts of environmental studies to analyze the uses, over-exploitation, and environmental problems associated with forest, water, mineral, food, and energy resources. (L3)

CO2: Apply the concepts of ecosystem structure and function to different ecosystem types, and analyze the value, threats, and conservation strategies of biodiversity. (L3)

CO3: Identify causes, effects, and control measures of air, water, soil, marine, noise,thermal, and nuclear pollution; apply solid waste management principles; analyse pollution case studies and disaster management strategies. (L4)

CO4: Analyse social issues related to unsustainable development; evaluate impacts of climate change and ozone depletion; examine Indian environmental legislation and assess enforcement challenges. (L4)

CO5: Describe population growth trends and environmental impacts; demonstrate the role of IT and AI tools in environmental monitoring and field data collection. (L3)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	1			1	3	2				2	1	1
CO2	2	2		1	2	3	2				2	2	1
CO3	2	2	2	2	2	3	3				2	1	2
CO4	2	2	2	2	2	3	3				3	2	2
CO5	1	1		1	3	3	2				3	3	3

Unit I: Multidisciplinary Nature of Environmental Studies and Natural Resources:

Core Content: Multidisciplinary Nature of Environmental Studies: definition, scope, and importance — need for public awareness. Natural Resources: renewable and non-renewable resources — forest resources (use, over-exploitation, deforestation, case studies), water resources (use and over-utilisation, floods, drought, conflicts over water, dams), mineral resources (environmental effects of extracting and using mineral resources), food resources (world food problems, effects of modern agriculture, fertilizer-pesticide problems), energy resources (use and exploitation of conventional and non-conventional energy sources).

AI Integration: Students use Google Earth Engine for deforestation and land-use change mapping, Python (scikit-learn) for groundwater level prediction, and ML forecasting tools for renewable energy output analysis.

Free/Open-Source AI Tools: Google Earth Engine (free cloud GIS); QGIS (free, open-source); Python (scikit-learn, Matplotlib) on Google Colab (free); Global Forest Watch (free dashboard); AI tools: ChatGPT / Gemini

Unit II: Ecosystems and Biodiversity and its Conservation:

Core Content: Ecosystems: concept, structure and function — producers, consumers and decomposers — energy flow, ecological succession, food chains, food webs and ecological pyramids. Types, characteristic features, structure and function of: forest, grassland, desert, and aquatic ecosystems. Biodiversity and its Conservation: definition, bio-geographical classification of India, value of biodiversity, India as a mega-diversity nation, hot-spots, threats to biodiversity, endangered and endemic species of India, in-situ and ex-situ conservation.

AI Integration: Students use iNaturalist for AI-assisted species identification during a campus biodiversity survey, MaxEnt for biodiversity hotspot mapping, and Google Earth Engine for NDVI-based ecosystem health comparison.

Free/Open-Source AI Tools: iNaturalist (free app/web); MaxEnt (free species distribution model); Google Earth Engine (free); QGIS (free); Python on Google Colab (free)

Unit III: Environmental Pollution and Solid Waste Management:

Core Content: Environmental Pollution: definition, cause, effects and control measures of air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution, and nuclear hazards. Solid Waste Management: causes, effects and control measures of urban and industrial wastes — role of an individual in prevention of pollution — pollution case studies — disaster management: floods, earthquake, cyclone and landslides.

AI Integration: Students build an ML-based Air Quality Index prediction model using Python (scikit-learn, TensorFlow), analyse IoT water quality sensor data for anomaly detection, and use QGIS/Python for disaster risk zone mapping.

Free/Open-Source AI Tools: OpenAQ (free global AQ data API); QGIS (free); Python (scikit-learn, TensorFlow, Matplotlib) on Google Colab (free); CPCB data portal (free); AI tools: ChatGPT / Gemini

Unit IV: Social Issues and the Environment:

Core Content: From unsustainable to sustainable development. Urban problems related to energy. Water conservation, rainwater harvesting, watershed management. Resettlement and rehabilitation of people — problems and case studies. Environmental ethics: issues and possible solutions. Climate change, global warming, acid rain, ozone layer depletion — case studies. Wasteland reclamation. Consumerism and waste products. Environment Protection Act, Air and Water Pollution Control Acts, Wildlife Protection Act, Forest Conservation Act — issues in enforcement. Public awareness.

AI Integration: Students analyse CMIP6 climate model data with Python (xarray, Matplotlib), estimate carbon footprints using scikit-learn, and use QGIS/Python for watershed delineation and rainwater harvesting potential mapping.

Free/Open-Source AI Tools: Python (xarray, Matplotlib, scikit-learn) on Google Colab (free); CMIP6 climate data (free, ESGF portal); QGIS (free); AI tools: ChatGPT / Gemini

Unit V: Human Population and the Environment:

Core Content: Population growth, variation among nations — population explosion — family welfare programmes. Environment and human health. Human rights. Value education. HIV/AIDS. Women and child welfare. Role of Information Technology in environment and human health. Case studies. Field Work: visit to a local area to document environmental assets (river, forest, grassland); visit to a local polluted site; study of common plants, insects, and birds.

AI Integration: Students use Python regression and LSTM models for population growth forecasting, explore AI in public health surveillance, and use iNaturalist/eBird with geo-tagging during field work to document environmental assets.

Free/Open-Source AI Tools: Naturalist (free); eBird (free); QGIS (free); Python (scikit-learn, TensorFlow) on Google Colab (free); OpenStreetMap (free); AI tools: ChatGPT / Gemini

Text Books:

1. Bharucha E., Textbook of Environmental Studies for Undergraduate Courses, Universities Press.
2. Palaniswamy, Environmental Studies, Pearson Education.

ReferenceBooks:

1. Dave D. and Reddy E.S.B., Textbook of Environmental Science, Cengage Publications.
2. Masters G.M. and Ela W.P., Introduction to Environmental Engineering and Science, Prentice Hall of India.

Web Resources & NPTEL Links:

- NPTEL: Environmental Science – https://nptel.ac.in/courses/124107002
- Google Earth Engine – https://earthengine.google.com
- Global Forest Watch – https://www.globalforestwatch.org
- OpenAQ – https://openaq.org
- iNaturalist – https://www.inaturalist.org
- QGIS – https://qgis.org

B. Tech.–II Year
II Semester

B. Tech. – II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26BSHB213T	BS	**Biology for Engineers** (Common to all branches of Engineering)	2	0	0	0	2

Pre Requisites: Basic Sciences (Intermediate level).

Course Objectives:

- To provide foundational biological knowledge with engineering relevance.
- To demonstrate AI-driven applications in biology and bioengineering.
- To encourage interdisciplinary teaching that connects biology, engineering, and computational tools.
- To prepare students for innovation in healthcare, materials, and bio-industries.
- To evaluate biomimetic principles and emerging bioengineering trends for real-world engineering applications.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamentals of biology, biomolecules, and human systems to engineering applications. (L3)

CO2: Apply AI tools and computational techniques for biological data analysis, biomolecule visualisation, and healthcare applications. (L3)

CO3: Analyse biomolecules and bio-inspired systems for solving engineering problems and developing sustainable technologies. (L4)

CO4: Design biomimetic and bioengineering solutions using interdisciplinary approaches and AI-enabled tools. (L5)

CO5: Evaluate emerging trends in bioengineering such as 3D bioprinting, regenerative medicine, and AI-based diagnostics for societal applications. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	1	2	1	1	1	2	1				2	1	1
CO2	2	3	2	2	2	2	1				2	2	2
CO3	2	3	2	3	3	2	1				3	1	1
CO4	2	3	2	3	3	2	1				3	1	2
CO5	2	3	3	3	3	2	1				3	1	2

Unit I: Foundations of Biology and AI:

Core Content: Cell as the basic unit of life. Carbohydrates, proteins, lipids, enzymes, vitamins, hormones.

AI Integration: AI-based visualization of biomolecules (e.g., AlphaFold). Case studies on AI in drug discovery.

Free/Open-Source AI Tools: Protein folding prediction using AlphaFold; use of DeepChem to simulate enzyme-substrate binding; comparison of AI-predicted protein structures with experimental data.

Unit II: Biomolecules in Engineering Applications:

Core Content: Carbohydrates in bioplastics (PHA, PLA). Proteins in food engineering (whey protein, plant analogs).

AI Integration: AI-driven generative design for biomaterials. Lipids in biodiesel & detergents.

Free/Open-Source AI Tools: PLA bioplastic design using ANSYS; AI-based optimization of protein-rich diets; design of a cellulose-based water filter using AI simulation; model biodiesel yield prediction using MATLAB.

UNIT III: Human Systems and Bio-Designs

Core Content: Brain as CPU (EEG, robotic prosthetics). Eye as camera (optical corrections, lens materials).

AI Integration: Heart as pump (ECG, stents). Kidney as filtration (dialysis systems).

Free/Open-Source AI Tools: AI-based ECG anomaly detection; robotic arm design inspired by neural signals; use of AI imaging tools to detect cataracts; simulate stent design using ANSYS.

UNIT IV: Bio-Inspired Materials and Mechanisms

Core Content: Photosynthesis → photovoltaic cells. Bird flight → GPS and aircraft navigation.

AI Integration: Lotus leaf → superhydrophobic surfaces. Kingfisher beak → bullet train design.

Free/Open-Source AI Tools: Model solar energy conversion inspired by photosynthesis; AI simulation of lotus-leaf hydrophobicity; kingfisher-inspired aerodynamic modelling; use of AI to design self-cleaning surfaces.

UNIT V: Emerging Trends in Bioengineering:

Core Content: Muscular and skeletal scaffolds. 3D bioprinting of tissues (ear, bone, skin).

AI Integration: AI in medical imaging (MRI, CT scans). AI in diagnostics and predictive healthcare.

Free/Open-Source AI Tools: AI-based tumor detection in MRI scans; 3D bioprinting case study of bone tissue; use of AI to classify medical images; design of a scaffold for bone regeneration using ANSYS.

Text Books:

1. Singh R.C. and Rao N.R., Biology for Engineers, Rajendra Singh C and Rathnakar Rao N Publishing, Bengaluru, 2023.
2. Thyagarajan S. et al., Biology for Engineers, Tata McGraw-Hill, New Delhi, 2012.

Reference Books:

1. Fox S. and Rompolski K., Human Physiology, McGraw-Hill eBook, 16th Edition, 2022.
2. Bar-Cohen Y., Biomimetics: Nature-Based Innovation, 1st Edition, CRC Press, 2012.
3. Ozbolat I., 3D Bioprinting: Fundamentals, Principles and Applications, Academic Press, 2016.

Web Resources:

- NPTEL Biology for Engineers – https://nptel.ac.in/courses/121106008
- MIT OpenCourseWare – Introduction to Biological Engineering: https://ocw.mit.edu
- AlphaFold Structure Database – https://alphafold.ebi.ac.uk

B. Tech. – II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26HMHS204T	HM	**Universal Human Values – Understanding Harmony & Ethical Human Conduct** (Common to all branches of Engineering)	3	0	0	0	3

Pre Requisites: Basic understanding of ethics, human values, society and environmental responsibility, along with openness to self-reflection and value-based learning.

Course Objectives:

- To introduce students to the philosophy, principles, and stages of value education and its relevance to engineering practice.
- To enable students to explore harmony at individual, family, societal, and ecological levels through structured self-reflection.
- To develop an understanding of professional ethics, engineering codes of conduct, and the consequences of unethical practice.
- To expose students to the concept of ecological balance, sustainable development, and the engineer's responsibility to nature.
- To cultivate ethical reasoning through analysis of real engineering case studies, AI ethics debates, and personal value charters.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: To apply the concepts of universal human values, natural acceptance, and the purpose of human life as articulated in Indian value philosophy to personal and professional decision-making. (L3)

CO2: Apply the framework of values-based living to personal relationships, family harmony, and societal well-being in daily life contexts. (L3)

CO3: Analyse ethical dilemmas in engineering practice using the lens of universal human values and differentiate value-driven from interest-driven conduct.(L4)

CO4: Evaluate current engineering and social practices for their alignment with sustainability, ecological balance, and universal human order(L5)

CO5: Create a personal value charter and an action plan for ethical engineering practice using AI reflection tools and collaborative dialogue.(L6)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1–Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1						2	3	1	2		2		
CO2						3	3	2	2		2		
CO3						3	3	2	2		3		
CO4						3	3	2	2		3		
CO5						3	3	3	3		3		

Unit I: Introduction to Value Education and Human Consciousness:

Core Content: Need for value education: crisis of values in modern society. Human being: a co-existence of Self (Jeevan) and Body; consciousness as a distinguishing feature. Natural acceptance: the innate human capacity to know what is right. Activities of the Self: Knowing, Assuming, Recognising, Fulfilling. Difference between animal consciousness and human consciousness.

AI Integration: Present ethical dilemmas to AI and compare AI responses with naturally accepted human values, use an AI journalling assistant to document daily natural acceptances and violations, can AI ever have human values? Debate using structured arguments

Free/Open-Source AI Tools: ChatGPT / Gemini (free tiers); Notion AI (free tier) for journalling

Unit II: Harmony in the Individual:

Core Content: Harmony between Self and Body: needs of the Self (happiness, trust) vs. needs of the Body (food, shelter, clothing). Four dimensions of human being: Icchha (desire), Vichar (thought), Anubhav (experience), Bodh (realisation). Feelings in relationships: nine feelings — trust, respect, affection, care, guidance, reverence, glory, gratitude, love. Understanding happiness and prosperity: holistic definition beyond materialistic view. Self-regulation vs. external regulation: inner motivation and its role in ethical conduct.

AI Integration: Explore conversations about emotional needs; critically reflect on AI limitations in understanding values, create a 'Well-being Wheel'; use AI to identify gaps between material and value-based happiness, write a weekly reflection on one feeling (e.g., trust) experienced and observed.

Free/Open-Source AI Tools: Replika (free); Canva (free); Notion AI (free tier); ChatGPT / Gemini (free tiers)

Unit III: Harmony in the Family and Society:

Core Content: Family as the basic unit of living: roles and responsibilities in family. Trust as the foundational value in relationships. From family to society: comprehensive human goal — undivided society. Universal Human Order: five dimensions of human existence (Individual, Family, Society, Nature, Existence). Social justice: ensuring rights, duties, and dignified participation for all.

AI Integration: Generate scenarios of family conflicts and analyse through the lens of trust and respect, use AI to map historical social movements (Gandhi, Ambedkar) onto the five-order framework, document 3 community service observations and analyse them using the harmony framework; use AI for structuring the report.

Free/Open-Source AI Tools: ChatGPT / Gemini (free tiers); Notion AI (free tier) for report structuring

Unit IV: Harmony in Nature and Existence:

Core Content: Four orders of nature: material, plant/bio, animal, human. Interconnectedness: mutual fulfilment among the four orders. Ecological balance: current ecological crises and human responsibility. Sustainable Development Goals (SDGs) and value-based engineering. Right understanding of existence: co-existence of units in space.

AI Integration: Analyse deforestation data and correlate with lack of ecological values, generate a sustainable engineering project idea and critique it using ecological harmony principles, create an ecological audit of the campus using satellite images (Google Earth) and document AI-assisted analysis

Free/Open-Source AI Tools: Global Forest Watch (free); Google Earth (free); ChatGPT / Gemini (free tiers)

Unit V: Ethical Human Conduct and Professional Ethics:

Core Content: Right understanding, right feeling, right action: the value-ethics-skill triad. Professional ethics: codes of conduct for engineers (ASME, IEEE, IEI). Corruption, dishonesty, and short-term thinking: causes and value-based remedies. Vision for a value-based society and universally prosperous world. AI Ethics: bias, fairness, transparency, and accountability in AI systems.

AI Integration: Demonstrate AI bias using sample datasets, analyse Bhopal Gas Tragedy / Challenger Disaster through professional ethics lens, use a given AI system and evaluate it on fairness and transparency, write a 'Personal Value Charter' as a future engineer; use ChatGPT for drafting and self-editing for authenticity.

Free/Open-Source AI Tools: IBM AI Fairness 360 (free, open-source); ChatGPT / Gemini (free tiers); Notion AI (free tier) for charter drafting

Text Books:

1. Gaur R.R., Sangal R. and Bagaria G.P., A Foundation Course in Human Values and Professional Ethics, 3rd Edition, Excel Books, New Delhi, 2019.
2. Nagraj A., Human Values and Professional Ethics: Universal Human Order, Jeevan Vidya Prakashan, Amarkantak, 2015.

Reference Books:

1. Subramanian C.V., Professional Ethics and Human Values, Oxford University Press, 2020.
2. Bajpai S.C., Engineering Ethics: Concepts and Cases, McGraw-Hill Education, 2019.
3. Covey S.R., The 7 Habits of Highly Effective People, Simon & Schuster, 30th Anniversary Edition, 2020.
4. Singer P., Practical Ethics, 4th Edition, Cambridge University Press, 2021.

Web Resources:

- AICTE – Value Education Resources – https://aicte-india.org
- UNESCO – Ethics of AI – https://www.unesco.org
- IEEE Ethics Resources – https://ethics.iit.edu
- NPTEL – Human Values & Professional Ethics – https://swayam.gov.in

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02207T	PC	AI Assisted Control Systems	2	0	2	0	3

Prerequisites: Linear Algebra and Vector Calculus, Differential Equations & Vector Calculus, Electrical Circuits, AI Tools & Applications.

Course Objectives:

- To understand classical control system concepts and mathematical modelling.
- To analyze time-domain and frequency-domain responses of systems.
- To evaluate system stability using analytical and graphical methods.
- To design controllers and compensators using classical and AI techniques.
- To apply state-space methods and AI tools for modern control problems.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamental concepts of control systems, feedback mechanisms, mathematical modelling, and servo systems with and without AI-assisted visualization tools. (L4)

CO2: Analyse the time response characteristics, steady-state analysis, and controller actions of control systems using conventional and AI-based simulation techniques**.** (L4)

CO3: Apply stability analysis methods such as Routh criterion and Root Locus techniques for evaluating control system performance using analytical and AI-assisted tools. (L4)

CO4: Analyze frequency response characteristics, compensation techniques, and stability margins using classical and AI-enabled computational methods. (L4)

CO5: Design state-space models, controllability and observability analysis, and intelligent control solutions using conventional and AI-based approaches (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2			2		1					3	2
CO2	3	3	1	1	3		1					3	2
CO3	3	3	2	2	3		1					2	2
CO4	3	3	3	3	3		1					2	3
CO5	3	3	3	3	3		1					3	3

Unit I: Control Systems Concepts:

Open loop and closed loop control systems and their differences – Examples of control systems – Classification of control systems, Feedback characteristics, Effects of positive and negative feedback, Mathematical models – Differential equations of translational and rotational mechanical systems and electrical systems, Analogous Systems, Block diagram reduction methods – Signal flow graphs – Reduction using Mason's gain formula. Principle of operation of DC and AC Servo motor, Transfer function of DC servo motor – AC servo motor, Synchros.

AI Integration: AI-assisted block diagram reduction and signal flow analysis, AI-based parameter estimation of transfer functions, Neural network modelling of servo motor behaviour, Visualization of analogous systems and transfer functions

Free/Open-Source AI Tools:

Python (Pandas, Matplotlib), Open Modelica, and TensorFlow / PyTorch (basic modeling).

Unit II: Time Response Analysis:

Standard Test Signals – Step Response – Impulse Response – Time response of first order systems – Characteristic Equation of Feedback control systems, Transient response of second order systems – Time domain specifications – Steady state response – Steady state errors and error constants, P, PI, PID Controllers.

AI Integration: AI-based prediction of transient and steady-state response, Visualization of first and second order system responses, Machine learning analysis of time-domain specifications, Intelligent tuning of P, PI, and PID controllers

Free/Open-Source AI Tools:

Python (control systems library), MATLAB alternative: Scilab Xcos, Scikit-learn, and TensorFlow

Unit III: Stability Analysis in Time Domain:

The concept of stability – Routh's stability criterion – Stability and conditional stability – Limitations of Routh's stability. The Root locus concept – Construction of root loci – Effects of adding poles and zeros to G(s)H(s) on the root loci.

AI Integration: Classification of stable/unstable systems using ML, AI-assisted stability prediction using Routh criterion data, Machine learning analysis of pole-zero movement

Free/Open-Source AI Tools:

Python (scikit-learn),Jupyter Notebook, and TensorFlow.

Unit IV: Frequency Response Analysis:

Introduction, Frequency domain specifications – Bode diagrams – Determination of Frequency domain specifications and transfer function from the Bode Diagram – Stability Analysis from Bode Plots. Polar Plots – Nyquist Plots – Phase margin and Gain margin – Stability Analysis. Compensation techniques – Lag, Lead, Lag-Lead Compensator design in frequency domain.

AI Integration: AI-assisted generation and analysis of Bode plots, Machine learning prediction of gain margin and phase margin, Visualization of Nyquist and Polar plots, Intelligent compensator parameter optimization

Free/Open-Source AI Tools:

Python (Matplotlib), Scikit-learn, Jupyter Notebook, and TensorFlow.

Unit V: State Space Analysis of Continuous Systems:

Concepts of state, state variables and state model – State models: differential equations & Transfer function models – Block diagrams. Diagonalization, Transfer function from state model, Solving the Time invariant state Equations – State Transition Matrix and its Properties. System response through State Space models. The concepts of controllability and observability, Duality between controllability and observability.

AI Integration: AI-assisted state-space modeling and system response prediction, Visualization of state trajectories and transition matrices, Machine learning analysis of controllability and observability, Comparative analysis of transfer function and state-space models using AI

Free/Open-Source AI Tools:

Python (NumPy, SciPy), OpenModelica ,TensorFlow, and Scikit-learn.

Text books:

1. Modern Control Engineering by Katsuhiko Ogata, Prentice Hall of India Pvt. Ltd., 5th edition, 2010.

2. Control Systems Engineering by I. J. Nagrath and M. Gopal, New Age International (P) Limited Publishers, 5th edition, 2007.

Reference Books:

1. Control Systems Principles & Design by M. Gopal, 4th Edition, McGraw Hill Education, 2012.

2. Automatic Control Systems by B. C. Kuo and Farid Golnaraghi, John Wiley and Sons, 8th edition, 2003.

3. Feedback and Control Systems, Joseph J. DiStefano III, Allen R. Stubberud & Ivan J. Williams, 2nd Edition, Schaum's Outlines, McGraw Hill Education, 2013.

4. Control System Design by Graham C. Goodwin, Stefan F. Graebe and Mario E. Salgado, Pearson, 2000.

5. Feedback Control of Dynamic Systems by Gene F. Franklin, J.D. Powell and Abbas Emami-Naeini, 6th Edition, Pearson, 2010.

Web Resources & NPTEL Links:

- NPTEL – Control Systems: https://nptel.ac.in/courses/108102043
- NPTEL – Advanced Control Systems: https://nptel.ac.in/courses/108106098

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02208T	PC	AI-Assisted Control of Induction and Synchronous Machines	2	0	2	0	3

Prerequisites: Electrical Circuits & DC Machines, AI Tools & Applications.

Course Objective:

- To understand the construction, working principles, and performance characteristics of induction motors and synchronous machines.
- To analyze torque-slip characteristics, equivalent circuits, and testing methods for induction motors.
- To evaluate voltage regulation, synchronization methods, and performance of synchronous generators.
- To apply AI/ML tools for prediction, monitoring, and control of induction and synchronous machines.
- To design intelligent drive systems and AI-assisted monitoring frameworks for rotating electrical machines.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyse the construction, operating principles, characteristics, and applications of induction motors, synchronous machines, and special electrical machines with AI-assisted visualization tools. (L3)

CO2: Analyse the performance characteristics, testing methods, starting techniques, speed control methods, and synchronization of electrical machines using conventional and AI-based approaches. (L3)

CO3: Apply equivalent circuits, torque equations, performance analysis methods, and AI-assisted computational tools for evaluating machine behavior. (L4)

CO4: Analyze voltage regulation, stability, power factor improvement, speed control, and machine characteristics using analytical and AI-enabled techniques. (L4)

CO5: Design intelligent models, smart drive systems, and AI-assisted monitoring and control techniques for electrical machines using modern open-source platforms. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2			2	1	1				1	3	1
CO2	3	3	1	1	3	1	1				1	3	2
CO3	3	2	2	2	3	1	1				1	3	2
CO4	2	2	3	3	3	1	1				1	2	3
CO5	2	3	3	3	3	1	1				1	2	3

Unit I: 3-Phase Induction Motors:

Construction of Squirrel cage and Slipring induction motors – Production of rotating magnetic field – Principle of operation – Rotor emf and rotor frequency – Rotor current and power factor at standstill and during running conditions – Rotor power input, rotor copper loss and mechanical power developed and their inter-relationship – Equivalent circuit – Phasor diagram, Applications.

AI Integration: Machine learning models for predicting slip, efficiency and torque from motor operating data, AI-assisted analysis of equivalent circuits and phasor diagrams, Intelligent monitoring of induction motor operating conditions, Digital twin modelling of induction motors.

Free/Open-Source AI Tools:

Python, Scikit-learn, Jupyter Notebook and GNU Octave

Unit II: Performance of 3-Phase Induction Motors:

Torque equation – Expressions for maximum torque and starting torque – Torque-slip characteristics – Double cage and deep bar rotors – No load, Brake test and Blocked rotor tests – Circle diagram for predetermination of performance – Methods of starting – Starting current and torque calculations – Speed control of induction motor with V/f control method, rotor resistance control and rotor emf injection technique – Crawling and cogging – Induction generator operation.

AI Integration: Neural networks for torque-speed curve prediction, AI-assisted speed control analysis using V/f control, AI-based visualization of circle diagram characteristics, Intelligent analysis of starting torque and starting current

Free/Open-Source AI Tools:

TensorFlow, PyTorch and Scilab

Unit III: Synchronous Generator:

Constructional features of non-salient and salient pole type alternators – Armature windings – Distributed and concentrated windings – Distribution & pitch factors – EMF equation – Armature reaction – Voltage regulation by synchronous impedance method – MMF method and Potier triangle method – Two reaction analysis of salient pole machines – Methods of synchronization – Slip test – Parallel operation of alternators.

AI Integration: AI-based voltage regulation prediction using load data, Smart monitoring and performance analysis of alternators, Machine learning prediction of generator efficiency and performance, AI-assisted synchronization analysis of alternators

Free/Open-Source AI Tools:

OpenModelica, Python, and Ng spice

Unit IV: Synchronous Motor:

Synchronous motor principle and theory of operation – Effect of excitation on current and power factor – Synchronous condenser – Expression for power developed – Hunting and its suppression – Methods of starting, Applications.

AI Integration: AI-based optimization for power factor improvement, Reinforcement learning for voltage and excitation control, AI-assisted prediction of synchronous motor performance, Intelligent monitoring of hunting and stability conditions

Free/Open-Source AI Tools:

Python (Keras, SciPy) and TensorFlow, PyTorch

Unit V: Special Machines:

Single phase induction motors – Constructional features – Equivalent circuit – Starting methods; Principle and operation of Shaded pole motor. Principle of operation: Reluctance motor – Stepper motor – BLDC motor and universal motor – Applications.

AI Integration:AI-based sensorless control of BLDC motors, Reinforcement learning for stepper motor positioning control, AI-assisted performance analysis of single-phase induction motors, Data-driven analysis of universal motor performance

Free/Open-Source AI Tools:

Scilab Xcos ,Keras and Python

Text books:

1. Electrical Machinery, Dr. P.S. Bhimbra, Khanna Publishing, 2021, First Edition.
2. Performance and Analysis of AC Machines by M.G. Say, CBS, 2002.

Reference Books:

1. Electrical Machines, D.P. Kothari and I.J. Nagrath, McGraw Hill Education, 2017, Fifth Edition.
2. Theory & Performance of Electrical Machines by J.B. Gupta, S.K. Kataria & Sons, 2007.
3. Electric Machinery, A.E. Fitzgerald, Charles Kingsley, Stephen D. Umans, McGraw-Hill, 2020, Seventh edition.

Web Resources & NPTEL Links:

- NPTEL – Electrical Machines: https://nptel.ac.in/courses/108/105/108105131
- NPTEL – Power Electronics: https://nptel.ac.in/courses/108106072

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02209T	PC	Intelligent Power Systems – I	2	0	2	0	3

Prerequisites: Electrical Circuits, Electrical Machines, AI Tools & Applications.

Course Objectives:

- To understand the working principles and components of various power generation plants.
- To analyze the structure and operation of substations and distribution systems.
- To study underground cables and their electrical characteristics.
- To evaluate economic aspects and tariff structures of power systems.
- To apply AI-based computational tools for monitoring, prediction and optimization in modern power systems.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyse the principles, construction, operation, and components of hydroelectric, thermal, and nuclear power stations, substations, distribution systems, underground cables, and tariff structures in power systems. (L3)

CO2: Apply power system concepts to analyze power plant operation, substation configurations, cable parameters, distribution schemes, load curves, and tariff calculations for electrical power engineering applications.. (L3)

CO3: Analyze the performance, efficiency, reliability, economic operation, fault conditions, and load characteristics of power plants, substations, distribution networks, and underground cable systems. (L4)

CO4: Apply and analyze power generation forecasting, fault diagnosis, predictive maintenance, smart grid monitoring, load forecasting, tariff optimization, and equipment health assessment in electrical power systems using AI techniques. (L3)

CO5: Design intelligent and energy-efficient power system solutions including smart substations, optimized distribution networks, dynamic tariff models, and AI-assisted monitoring systems using open-source simulation and data analytics tools. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	1	1	1				1	3	1
CO2	3	3	2	2	2	2	1				2	3	2
CO3	3	3	2	3	2	1	1				2	3	2
CO4	3	2	2	2	2	2	1				2	2	2
CO5	3	2	3	2	3	2	1				3	2	3

Unit I: Hydroelectric and Thermal Power Stations:

Hydroelectric Power Stations: Selection of site – General layout of a hydroelectric power plant with brief description of major components and principle of operation. Thermal Power Stations: Selection of site – General layout of a thermal power plant – Brief description of components: boilers, super heaters, economizers and electrostatic precipitators, steam turbines (impulse and reaction), condensers, feed water circuit, cooling towers and chimney.

AI Integration: Predicting hydroelectric power generation using rainfall and reservoir data, AI-based thermal plant efficiency monitoring, Predictive maintenance of turbines and boilers using sensor data, Energy generation forecasting using regression models.

Free/Open-Source AI Tools:

Python (NumPy, Pandas, Matplotlib), Scikit-learn, and TensorFlow / PyTorch.

Unit II: Nuclear Power Station:

Location of nuclear power plant – Working principle – Nuclear fission – Nuclear fuels – Nuclear chain reaction – Nuclear reactor components: moderators, control rods, reflectors and coolants – Types of nuclear reactors and brief description of PWR, BWR and FBR. Radiation: Radiation hazards and shielding – Nuclear waste disposal.

AI Integration: AI-based reactor safety monitoring, Machine learning for radiation leakage detection, AI-based nuclear waste monitoring and classification, Neural network models for reactor stability prediction

Free/Open-Source AI Tools:

Python (NumPy, Pandas, Matplotlib),Scikit-learn and TensorFlow / PyTorch

Unit III: Substations:

Air Insulated Substations – Indoor & outdoor substations – Substations layouts of 33/11 kV showing the location of all the substation equipment – Bus bar arrangements: simple arrangements like single bus bar, sectionalized single bus bar, double bus bar with one and two circuit breakers, main and transfer bus bar system. Gas Insulated Substations (GIS) – Advantages of GIS, constructional aspects, comparison of air insulated substations and GIS.

AI Integration: AI-based fault diagnosis in substations, Predictive maintenance of circuit breakers and transformers, AI-based load forecasting at substations, Smart grid-based substation automation

Free/Open-Source AI Tools:

Python (NumPy, Pandas, Matplotlib), Scikit-learn and TensorFlow / PyTorch.

Unit IV: Distribution Systems and Underground Cables:

Distribution Systems: Classification of Distribution systems, AC Distribution, Overhead versus Underground system – Connection schemes of Distribution system – Requirements of Distribution system – Design considerations in Distribution system. Underground Cables: Types of cables – Construction, types of insulating materials – Calculation of insulation resistance – Stress in insulation and power factor of cable – Capacitance of single and 3-Core belted cables. Grading of cables: capacitance grading and intersheath grading.

AI Integration: AI-based distribution load forecasting, Machine learning for cable insulation failure prediction, Smart grid monitoring of distribution networks, AI-based optimal routing for underground cable networks.

Free/Open-Source AI Tools:

Python (NumPy, Pandas, Matplotlib), Scikit-learn and TensorFlow / PyTorch.

Unit V: Economic Aspects & Tariff:

Economic Aspects – Load curve – Load duration and integrated load duration curves – Discussion on economic aspects: connected load, maximum demand, demand factor, base and peak load plants, load factor, diversity factor, plant capacity factor and plant use factor. Tariff Methods – Costs of generation and their division into fixed, semi-fixed and running costs – Desirable characteristics of a tariff method – Tariff methods: simple rate, flat rate, block-rate, two-part, three-part and power factor tariff methods – Time of Day (ToD) tariff and Time of Use (ToU) tariff.

AI Integration: AI-based electricity demand forecasting, Machine learning for dynamic tariff optimization, AI-based consumer load pattern analysis, Optimization of electricity pricing strategies

Free/Open-Source AI Tools:

OpenDSS, MATPOWER (Octave), GridLAB-D, and Power Factory Educational Tools.

Text books:

1. S. N. Singh, Electric Power Generation, Transmission and Distribution, PHI Learning Pvt Ltd, New Delhi, 2nd Edition, 2010.
2. J. B. Gupta, Transmission and Distribution of Electrical Power, S. K. Kataria and Sons, 10th Edition, 201.

Reference Books:

1. I. J. Nagarath & D.P. Kothari, Power System Engineering, McGraw-Hill Education, 3rd Edition, 2019.
2. C. L. Wadhwa, Generation, Distribution and Utilization of Electrical Energy, New Age International Publishers, 6th Edition, 2018.
3. V. K. Mehta and Rohit Mehta, Principles of Power System, S. Chand, 4th Edition, 2005.
4. Turan Gonen, Electric Power Distribution System Engineering, McGraw-Hill, 1985.
5. Handbook of Switchgear, BHEL, McGraw-Hill Education, 2007.

Web Resources & NPTEL Links:

- NPTEL – Power Plant Engineering: https://nptel.ac.in/courses/108105058
- NPTEL – Power Distribution Engineering: https://nptel.ac.in/courses/108101039
- MIT OpenCourseWare – Electric Power Systems: https://ocw.mit.edu/courses/electric-power-systems
- Smart Grid Course: https://www.edx.org/learn/smart-grid
- Machine Learning for Engineers: https://scikit-learn.org/stable/tutorial

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC04203T	PC	AI Integrated Signal Processing and Systems	2	0	2	0	3

Prerequisites: Basic knowledge of mathematics, differential equations, complex numbers, and fundamental concepts of electrical and electronic circuits.

Course Objectives:

- To understand the fundamental concepts of signals, systems, and Fourier series representations.
- To analyze Fourier, transform techniques, sampling theory, and signal reconstruction methods.
- To apply Laplace Transform, Z-Transform, DFT, and FFT for signal and system analysis.
- To evaluate digital filter design methods including FIR and IIR filters.
- To apply AI/ML tools for intelligent signal processing, spectral analysis, and filter design optimization.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Analyse the the fundamental concepts of signals, systems, Fourier series, and properties of continuous-time and discrete-time systems with AI-assisted visualization tools. (L3)

CO2: Apply Fourier transform techniques, sampling theory, signal reconstruction, and filtering concepts using conventional and AI-based analytical methods. (L3)

CO3: Apply Laplace Transform, Z-Transform, DFT, FFT, and digital filter realization techniques for signal and system analysis using computational and AI-assisted tools. (L3)

CO4: Analyze stability, spectral characteristics, convolution operations, and filter performance of continuous-time and discrete-time systems using classical and AI-enabled approaches. (L4)

CO5: Design FIR/IIR digital filters, multirate DSP systems, and intelligent signal processing models for engineering applications using conventional and AI-based techniques. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1				1				1	2	1
CO2	3	3	2	2			1				1	2	2
CO3	3	3	3	3	1		1				1	2	2
CO4	3	3	2	3	2		1				1	2	2
CO5	3	3	3	3	3		1				1	2	2

Unit I: Fundamentals of Signals and Systems:

Classification of Signals and Systems – Continuous-Time and Discrete-Time Signals – Basic Signal Operations: Time Shifting, Scaling, Folding – Elementary Signals and Sequences – System Properties: Linearity, Time Invariance, Causality, Stability, Memoryless Systems – Convolution and Correlation of Signals – Linear Time Invariant (LTI) Systems – Orthogonality of Signals – Fourier Series Representation of Periodic Signals – Trigonometric and Exponential Fourier Series – Properties of Fourier Series.

AI Integration: AI-based signal classification using Machine Learning, Machine learning-based identification of system properties, AI-assisted convolution and correlation analysis, Data-driven analysis of periodic and non-periodic signals

Free/Open-Source AI Tools:

GNU Octave, Python (NumPy, Pandas),and Scikit-learn

Unit 2: Fourier Transform Techniques and Sampling Theory:

Continuous-Time Fourier Transform (CTFT) – Properties of Fourier Transform – Frequency Spectrum Analysis – Discrete-Time Fourier Transform (DTFT) – Properties of DTFT – Energy Density Spectrum and Power Spectral Density – Sampling Theorem and Reconstruction of Signals – Aliasing and Anti-Aliasing Filters – Signal Bandwidth and System Bandwidth – Distortionless Transmission through Systems – Ideal Low Pass, High Pass and Band Pass Filters.

AI Integration: AI-based spectral analysis of signals, Intelligent detection of aliasing and sampling effects, AI-assisted analysis of CTFT and DTFT characteristics, Intelligent sampling and signal reconstruction

Free/Open-Source AI Tools:

Python (Librosa, SciPy, NumPy), Scikit-learn, Matplotlib, NumPy, TensorFlow, PyTorch,Audacity.

Unit III: Laplace Transform, Z-Transform and System Analysis:

Laplace Transform: Definition and Region of Convergence (ROC) – Properties and Inverse Laplace Transform – Transfer Function and System Response – Poles and Zeros in S-Plane – Stability Analysis and BIBO Stability – Solution of Differential Equations using Laplace Transform. Z-Transform: Definition and ROC – Properties of Z-Transform – Inverse Z-Transform Techniques – Solution of Difference Equations – Pole-Zero Analysis in Z-Plane – Stability of Discrete-Time Systems – Relationship among Fourier, Laplace and Z-Transforms.

AI Integration: AI-assisted pole-zero analysis in S-plane and Z-plane, AI-assisted analysis of transfer functions and system behavior, Intelligent stability prediction of continuous and discrete-time systems, Data-driven comparison of Fourier, Laplace, and Z-Transforms

Free/Open-Source AI Tools:

Python (Control Systems Library),Scikit-Learn,GNU Octave Control Package,MATLAB Alternative: Scilab Xcos, and Keras.

Unit IV: Discrete Fourier Transform and Digital Filter Structures:

Discrete Fourier Series (DFS) – Discrete Fourier Transform (DFT) – Properties of DFT – Circular and Linear Convolution using DFT – Fast Fourier Transform (FFT) – Radix-2 Decimation-in-Time FFT – Radix-2 Decimation-in-Frequency FFT – Inverse FFT and FFT for Composite N – Realization Structures of Digital Filters – FIR and IIR Filter Structures – Direct Form-I and II Structures – Cascade and Parallel Realization – Transposed Structures – Frequency Domain Representation of Discrete-Time Systems.

AI Integration: AI-assisted FFT spectrum analysis, Machine learning-based digital filter realization analysis, Intelligent FFT-based signal classification, AI-assisted convolution and frequency-domain processing

Free/Open-Source AI Tools:

Python (TensorFlow, PyTorch),GNU Octave Signal Package, Scikit-learn, Jupyter Notebook.

Unit V: Digital Filter Design and Multirate DSP:

Analog Filter Approximations: Butterworth and Chebyshev Filters – Design of IIR Digital Filters – Bilinear Transformation and Impulse Invariance Techniques – Characteristics of FIR Filters – FIR Filter Design using Window Techniques – Frequency Sampling Method – Comparison of FIR and IIR Filters – Multirate Digital Signal Processing – Decimation and Interpolation – Polyphase Structures – Nyquist Filters – Spectral Analysis of Non-Stationary Signals – Applications of DSP in Audio, Communication and Biomedical Engineering.

AI Integration: AI-assisted design optimization of FIR and IIR digital filters, Machine learning-based spectral analysis of non-stationary signals, AI-assisted communication signal analysis and enhancement, Data-driven comparison of Butterworth and Chebyshev filters

Free/Open-Source AI Tools:

Python (MNE, SciPy, TensorFlow),GNU Radio, and Scikit-learn.

Text books:

1. Signals and Systems, A.V. Oppenheim, A.S. Willsky and S.H. Nawab, PHI, 2nd Edition, 2009.
2. Digital Signal Processing – A Computer Based Approach, Sanjit K. Mitra, Tata McGraw Hill, 3rd edition, 2009.

Reference Books:

1. Principles of Linear Systems and Signals, B.P. Lathi, Oxford University Press, 2nd Edition, 2015.
2. Signals and Systems: A Primer with MATLAB, Matthew N.O. Sadiku and Warsame H. Ali, CRC Press
3. Signals & Systems, Simon Haykin and Van Veen, Wiley, 2nd Edition, 2005.
4. Digital Signal Processing: Principles, Algorithms and Applications, John G. Proakis, Dimitris G. Manolakis, Pearson Education/PHI, 2007.
5. Schaum's Outline of Signals and Systems, Hwei Hsu, Fourth Edition, TMH, 2019.
6. Digital Signal Processing, Andreas Antoniou, TATA McGraw Hill, 2006.
7. A Text book on Digital Signal Processing, R.S. Kaler, M. Kulkarni, Umesh Gupta, I K International Publishing House Pvt. Ltd., 2009.

Web Resources & NPTEL Links:

- NPTEL – Signals and Systems: https://nptel.ac.in/courses/108/106/108106151/
- NPTEL Lectures on Signals and Systems by Prof. K.S. Venkatesh, IIT Kanpur
- NPTEL Lectures on Signals and Systems by Prof. Aditya K. Jagannatham, IIT Kanpur

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02207P	PC	AI Assisted Control Systems Lab	0	0	0	3	1.5

Pre-Requisites: Basic knowledge of control systems, electrical and electronic circuits, mathematical modeling, differential equations, and basic programming concepts.

Course Objectives:

- To perform experiments on control systems and analyze system behaviour.
- To evaluate system performance using time and frequency domain techniques.
- To design and implement controllers for real-time systems.
- To apply modern tools and AI techniques in control experiments.
- To develop skills in experimentation, data analysis, and reporting.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the characteristics, transfer functions, stability concepts, and operational principles of control system components with AI-assisted visualization tools. (L3)

CO2: Analyze the effects of feedback, controller actions, compensation techniques, and PLC operations in control systems using conventional and AI-based methods. (L4)

CO3: Apply time response analysis, stability analysis, PID controller design, and PLC programming techniques for control system applications using computational and AI-assisted tools. (L3)

CO4: Analyze system stability, controllability, observability, servo motor performance, and controller characteristics using classical and AI-enabled analytical approaches (L4)

CO5: Design intelligent control solutions, PLC-based automation systems, and AI-assisted monitoring and control models for industrial applications. (L5)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2			2		3	3	3		1	3	1
CO2	3	3	1	1	3		3	3	3		1	3	2
CO3	3	3	2	2	3		3	3	3		1	3	3
CO4	3	3	3	3	3		3	3	3		1	3	3
CO5	2	2	3	2	3		3	3	3		1	2	2

List of Experiments:

Any 10 of the following experiments are to be conducted:

1. Time response of Second order system
2. Characteristics of Synchros
3. Effect of feedback on DC servo motor
4. Transfer function of DC Machine
5. Effect of P, PD, PI, PID Controller on a second order system
6. Lag and lead compensation – Magnitude and phase plot

7. Design of PID Controller for Temperature control using process simulator
8. Characteristics of magnetic amplifiers
9. Characteristics of AC servo motor
10. Verification of system stability using Root locus, Bode Plot & Nyquist Plot
11. Verification of time response analysis for a given system
12. Verification of controllability and observability for a given system
13. Programmable logic controller – Study and verification of truth tables of logic gates, simple Boolean expressions
14. PLC Based Temperature Control
15. Speed Control of DC Motor Using PLC

AI Integration – Minimum two of the following experiments are to be conducted:

1. ML-based prediction of time response parameters
2. Smart PLC using AI decision-making
3. Neural network modelling of servo motor
4. AI-based PID tuning
5. Optimization based compensator design
6. Intelligent temperature control (Fuzzy / ANN)
7. Time response prediction using deep learning

Recommended Free & Open-Source AI/Computational Tools

Control & Simulation:

- GNU Octave
- Scilab + Xcos
- OpenModelica

AI / ML Tools:

- Python (NumPy, SciPy, scikit-learn)
- TensorFlow / PyTorch
- Jupyter Notebook / Google Colab

Optimization / Advanced AI:

- DEAP (Genetic Algorithm)
- PyGAD
- OpenAI Gym (Reinforcement Learning)

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26PC02208P	SE	AI-Assisted Control of Induction and Synchronous Machines Lab	0	0	0	3	1.5

Pre-Requisites: Basic knowledge of electrical circuits, electromagnetic principles, AC machines, power measurements, and fundamental electrical engineering concepts.

Course Objectives:

- To perform experimental investigations on induction motors, synchronous machines and AC series motors.
- To analyze experimental results to determine performance characteristics such as efficiency, regulation and power factor.
- To study different testing methods for evaluating electrical machine performance.
- To utilize computational tools for visualization and analysis of experimental data.
- To apply AI-based tools for intelligent monitoring, prediction and fault detection in electrical machines.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the operating principles, characteristics, testing procedures, and performance parameters of induction motors, synchronous motors, alternators, and AC series motors used in laboratory experiments. (L3)

CO2: Apply experimental methods to conduct brake tests, circle diagram analysis, speed control, power factor improvement, voltage regulation tests, efficiency determination, and parallel operation of AC electrical machines. (L3)

CO3: Analyze the performance characteristics, efficiency, voltage regulation, speed–torque behaviour, synchronous reactance, and operating conditions of AC machines using measured and simulated experimental data. (L4)

CO4: Analyze AI and machine learning techniques such as regression models, neural networks, anomaly detection, and predictive analytics for efficiency estimation, fault diagnosis, voltage regulation prediction, and machine condition monitoring. (L4)

CO5: Design AI-assisted monitoring, predictive maintenance, and intelligent performance evaluation models for rotating electrical machines using experimental data visualization, pattern recognition, and open-source computational tools. (L4)

CO – PO Articulation Matrix:

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	3	2	1	3	3	3		2	3	2
CO2	3	3	2	3	2	1	3	3	3		2	3	2
CO3	3	3	2	3	2	1	3	3	3		2	3	2
CO4	2	2	1	2	3	1	3	3	3		3	2	2
CO5	2	2	2	2	3	2	3	3	3		3	2	3

List of Experiments:

Any 10 of the following experiments are to be conducted:

1. Brake test on three phase Induction Motor
2. Circle diagram of three phase induction motor
3. Speed control of three phase induction motor by V/f method
4. Equivalent circuit of single-phase induction motor
5. Power factor improvement of single-phase induction motor by using capacitors
6. Load test on single phase induction motor
7. Regulation of a three-phase alternator by synchronous impedance & MMF methods
8. Regulation of three-phase alternator by Potier triangle method
9. V and Inverted V curves of a three-phase synchronous motor
10. Determination of Xd, Xq & Regulation of a salient pole synchronous generator
11. Determination of efficiency of three phase alternator by loading with three phase induction motor
12. Parallel operation of three-phase alternator under no-load and load conditions
13. Determination of efficiency of a single-phase AC series Motor by conducting Brake test

AI Integration – Minimum two of the following experiments are to be conducted:

1. Machine learning models to predict efficiency of induction motors and alternators
2. AI-based speed–torque characteristic prediction for induction motors
3. Predictive maintenance models for electrical machines using vibration/current data
4. Regression models to estimate voltage regulation of alternators
5. AI-based analysis of V and inverted V curves of synchronous motors
6. Neural network estimation of synchronous reactance (Xd and Xq)
7. Data visualization and pattern recognition for experimental machine characteristics

References:

1. Engineering Circuits Analysis, Jack Kemmerly, William Hayt and Steven Durbin, Tata McGraw Hill Education, 2005, Sixth Edition.

2. Network Analysis, M. E. Van Valkenburg, Pearson Education, 2019, Revised Third Edition.

Web Resources & NPTEL Links :

- Virtual Labs – Electrical Machines: https://em-coep.vlabs.ac.in/List%20of%20experiments.html
- NPTEL – Electrical Machines: https://nptel.ac.in/courses/108105017
- MIT OpenCourseWare – Electric Machines: https://ocw.mit.edu/courses/electric-machines/
- Python for Engineering Applications: https://scipy.org
- Machine Learning with Python: https://scikit-learn.org/stable/tutorial/index.html
- OpenModelica Simulation Course: https://openmodelica.org/education

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26SE02201S	SE	Electrical Simulation Tools	1	0	0	2	2

Prerequisites: Electrical Circuits, Electrical Machines, AI Tools & Applications.

Course Objectives:

- To introduce the fundamentals of Electronic Design Automation (EDA), P-SPICE simulation techniques, and electrical circuit analysis methods.
- To develop the ability to analyze DC, AC, and power electronic circuits using P-SPICE simulation tools.
- To provide knowledge of MATLAB programming, numerical computation, visualization, and engineering problem solving.
- To familiarize students with MATLAB/Simulink applications in control systems, signal processing, neural networks, and fuzzy logic systems.
- To develop practical skills in E-CAD tools for schematic capture, PCB layout design, routing, and integration of simulation with PCB development.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the fundamentals of the P-SPICE environment, DC circuit analysis, and network theorems to simulate and analyze electrical circuits. (L3)

CO2: Apply P-SPICE tools to simulate and analyze AC circuit behavior, semiconductor device characteristics, and transient/frequency responses. (L3)

CO3: Apply MATLAB programming concepts, matrix operations, and numerical methods for solving electrical engineering problems. (L3)

CO4: Analyze control systems, signal processing models, neural networks, and fuzzy logic applications using MATLAB/Simulink. (L4)

CO5: Design PCB layouts and electrical schematics using E-CAD tools by following industrial design standards and verification techniques. (L5)

CO – PO Articulation Matrix

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2		1	3		3	3	3		2	3	2
CO2	3	3	1	2	3		3	3	3		2	3	2
CO3	3	2	2	2	3		3	3	3		2	3	2
CO4	3	3	3	3	3		3	3	3		3	3	3
CO5	3	3	3	2	3		3	3	3		3	3	3

Unit I: P-SPICE Fundamentals and DC Circuit Simulation:

Introduction to Electronic Design Automation (EDA) – Overview of P-SPICE environment – Circuit components, libraries, and schematic capture – DC analysis: Ohm's Law and Kirchhoff's Laws, Nodal and Mesh Analysis, Voltage and Current Divider Circuits, DC Sweep and Bias Point Analysis – Network theorems using P-SPICE: Superposition Theorem, Thevenin's Theorem & Norton's Theorem.

Unit II: P-SPICE for AC and Power Electronics Applications:
AC circuit simulation using P-SPICE – Transient and Frequency Response Analysis – RLC circuits and resonance – Introduction to semiconductor devices: Diodes, BJT, MOSFET – Harmonic analysis and waveform visualization.

Unit III: MATLAB Fundamentals for Electrical Engineering Applications:
Introduction to MATLAB environment – MATLAB commands, scripts, and functions – Matrix operations and array manipulation – Plotting and visualization tools – Numerical methods in MATLAB – Solving electrical engineering equations using MATLAB – Applications in circuit analysis and signal representation.

Unit IV: MATLAB for Control Systems and Signal Processing:
Introduction to Simulink – Modeling of dynamic systems – Transfer functions and state-space models – Time-domain analysis: Step response, Impulse response – Frequency-domain analysis: Bode plots, Nyquist plots – MATLAB applications in electrical drives and power systems – Neural Network Toolbox: Structure of neurons, transfer functions, representation of networks, simulating networks, Adaline, Backpropagation, self-organizing networks – Fuzzy logic toolbox: demonstration of fuzzy logic, Membership functions, Rule editor, rule viewer and surface viewer.

Unit V: E-CAD Tools for PCB Design and Electrical Layouts:
Introduction to E-CAD tools – PCB design flow and methodology – Schematic capture and netlist generation – PCB layout fundamentals: Layer management, Routing techniques, Ground planes – Design Rule Check (DRC) – Industrial PCB Design Standards – Integration of Simulation and PCB Design.

List of Experiments:

1. Nodal and Mesh analysis of a given electrical network
2. Calculation of Maximum Power dissipation of a given circuit
3. Obtain transient response of the given circuit
4. Determination of transfer function, Poles and zeros of a given circuit
5. Determination of frequency response of a given circuit
6. Determination of transmission parameters of a given cascaded network
7. Generation of Full wave rectifier waveform
8. Determination of diode characteristics
9. Determination of output characteristics of BJT and NPN transistor
10. Design of a single-phase transformer with the given specifications
11. Design of an Induction motor with the given specifications
12. Determination of mmf required for the air gap of a machine
13. Neural Network-Based Prediction System
14. Fuzzy Logic-Based Intelligent Control System
15. Hybrid Neuro-Fuzzy System
16. Design of PCB layout for regulated power supply
17. Schematic capture and PCB routing using E-CAD tool

Text Books:
1. Simulation of Electric and Electronic Circuits using PSPICE by M.H. Rashid, PHI Publications.
2. MATLAB: The Language of Technical Computing, MathWorks Inc., 2005.
3. PCB Design Tutorial by David L. Jones.

B.Tech.–II Year II Semester

Course Code	Category	Name of the Course	L	T	Pr	P	C
26HMHS206A	MC	Technical Paper Writing and IPR (Common to all branches of Engineering)	2	0	0	0	0

Prerequisites: Electrical Circuits, Electrical Machines, AI Tools & Applications.

Course Objective:

- To develop fundamental skills in technical communication, professional writing, and effective documentation.
- To enable students to prepare technical reports, research documents, proposals, and professional correspondence. To introduce the fundamental concepts of Intellectual Property Rights (IPR), including patents, copyrights, trademarks, and trade secrets.
- To develop awareness of IP protection, patent filing procedures, technology commercialization, and IP strategy.
- To familiarize students with AI-powered writing and research tools for improving productivity, clarity, and quality while maintaining academic integrity and ethical standards.

Course Outcomes (COs):

On successful completion of the course, Student will be able to

CO1: Apply the principles, types, and structure of technical reports and IPR fundamentals to prepare basic technical documents and identify appropriate forms of intellectual property protection. (L3)

CO2: Apply the principles of technical writing to produce well-structured laboratory reports, project reports, and technical proposals. (L3)

CO3: Analyse technical documents for clarity, coherence, accuracy, and compliance with formatting standards using AI-powered review tools. (L4)

CO4: Evaluate patent documents, prior art, and IP claims to assess novelty and inventiveness of an engineering innovation. (L5)

CO5: Create a complete technical report and draft a patent disclosure for an engineering project, integrating AI tools for writing assistance. (L6)

CO – PO Articulation Matrix

Scale: 3 – Strong, 2 – Moderate, 1 – Slight

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	1	1	1	1			2	1	3	2	2	2	1
CO2	2	2	2	2			2	2	3	2	3	2	1
CO3	2	3	2	2			2	2	3	2	3	1	2
CO4	2	3	2	3			3	2	2	2	3	2	2
CO5	2	2	3	2			2	3	3	3	3	2	2

Unit I: Foundations of Technical Communication:

Core Concepts:

- Technical Communication vs. General Communication: Precision, objectivity, and audience-awareness
- Types of Technical Documents: Reports, Proposals, Manuals, Specifications, Abstracts
- Writing Process for Technical Documents: Planning, Drafting, Revising, Proofreading
- Audience Analysis: Primary and secondary audiences; adapting tone and technicality
- Language in Technical Writing: Active vs. Passive voice, Nominalisations, Clarity and conciseness

AI Integration:

- AI Tool: Grammarly (free) – analyse a technical paragraph for passive voice overuse and nominalization
- AI Tool: Hemingway Editor – paste a technical section and improve readability score to Grade 10 or below
- Activity: Compare two versions of a technical document (before and after AI editing) and document changes

Unit II: **Technical Report Writing:**

Core Concepts:

- Structure of a Formal Report: Title page, Abstract, Table of Contents, Introduction, Methodology, Results, Discussion, Conclusion, References, Appendices
- Abstract Writing: Informative vs. Descriptive abstracts; 150–250 word limit
- Literature Review: Searching, Evaluating, and Synthesising sources
- Data Presentation: Tables, Figures, Charts – placement, numbering, and captioning
- Citation and Referencing: IEEE, APA, and MLA styles; avoiding plagiarism

AI Integration:

- AI Tool: Elicit.org (free) – AI-powered literature search; find and summarise 5 relevant papers
- AI Tool: Zotero (open-source reference manager) – manage citations and auto-generate bibliographies
- AI Tool: ChatGPT – generate a first-draft abstract; critique and rewrite to meet technical writing standards
- Plagiarism Check: Use Turnitin (institutional access) or Copyleaks (free tier) to verify originality
-

Unit III: **Research Proposals and Scientific Writing:**

Core Concepts:

- Research Proposal Structure: Background, Problem Statement, Objectives, Methodology, Timeline, Budget
- Scientific Paper Format: IMRAD Structure (Introduction, Methods, Results and Discussion)
- Writing Results vs. Discussion: Factual reporting vs. Interpretation
- Peer Review Process: Understanding blind review, revision, and resubmission
- Conference Papers vs. Journal Articles: Scope, length, and submission guidelines

AI Integration:

- AI Tool: Semantic Scholar (free) – search for open-access papers in your domain; analyse IMRAD structure
- AI Tool: Writefull (free academic writing app) – get AI feedback on scientific language in a draft methods section
- Activity: Write a 1-page research proposal for a small project; use AI for structural review and language polishing

Unit IV: **Intellectual Property Rights – Fundamentals:**

Core Concepts:

- Introduction to IPR: Definition, importance, and types – Patents, Trademarks, Copyrights, Trade Secrets, Geographical Indications
- Patent System in India: Indian Patent Act 1970, Patent Office, Patent application process

- Patentability Criteria: Novelty, Inventive Step (Non-obviousness), Industrial Applicability
- Patent Search: Prior art search using databases
- Copyright in Engineering: Software copyright, Open-source licenses (GPL, MIT, Apache)

AI Integration:

- AI Tool: Google Patents (free) – conduct a prior art search on a given invention idea
- AI Tool: Lens.org (free, open-source patent database) – search and analyse patent documents
- Activity: Identify 3 patents related to a chosen engineering domain and analyse their claims section
- Discussion: How does AI help in patent drafting? Use AI tools like PatSnap or Examiner (where accessible)
-

Unit V: **Patent Drafting and IP Strategy:**

Core Concepts:

- Structure of a Patent Application: Title, Abstract, Background, Summary, Detailed Description, Claims, Drawings
- Types of Claims: Independent and Dependent claims; method, product, system claims
- Filing a Provisional Patent Application (PPA) in India
- IP Strategy for Startups and Engineers: Trade secrets vs. patents, licensing, technology transfer
- Case Studies: Famous patent disputes (Apple vs. Samsung, Monsanto seed patents, pharmaceutical patents)

AI Integration:

- AI Tool: ChatGPT – draft a set of patent claims for a simple mechanical invention; critique for clarity and scope
- AI Tool: IP India Online Portal (ipindia.gov.in) – navigate and understand the e-filing system
- Capstone Project: Write a complete patent disclosure document for a project-based invention idea
- Activity: Use AI to convert a project report executive summary into a patent abstract; compare the two

Text Books:

1. T1. Markel, Mike & Selber, Stuart. (2022). Technical Communication (13th ed.). Bedford/St. Martin's, New York.
2. T2. Ahuja, B.N. & Ahuja, S.S. (2021). Intellectual Property Rights: Patents, Trade Marks, Copyrights and Related Topics (4th ed.). Khanna Publishers, New Delhi.

Reference Books:

1. R1. Lannon, John & Gurak, Laura. (2020). Technical Communication (15th ed.). Pearson Education.
2. R2. Anderson, Paul V. (2019). Technical Communication: A Reader-Centred Approach (9th ed.). Cengage.
3. R3. Meganathan, R. (2019). Intellectual Property Rights for Engineers. Universities Press (India).
4. R4. Cornish, William R. & Llewelyn, David. (2019). Intellectual Property: Patents, Copyright, Trademarks and Allied Rights (9th ed.). Sweet & Maxwell.
5. R5. Eisenberg, Anne. (2018). Effective Technical Communication. Tata McGraw-Hill.
6. R6. World Intellectual Property Organization (WIPO). (2022). WIPO Intellectual Property Handbook. WIPO Geneva.

Web Resources:

- IP India – Patent Office: www.ipindia.gov.in
- Google Patents: patents.google.com
- Lens.org – Free Patent and Scholar Search: www.lens.org
- WIPO – Patent Resources: www.wipo.int
- Purdue OWL – Technical Writing: owl.purdue.edu/owl/subject_specific_writing/professional_technical_writing

YouTube / Video Resources:

- YouTube: NPTEL – Technical English for Engineers – www.youtube.com/c/nptel
- YouTube: IP India – Patent Filing Guide – search on YouTube
- YouTube: WIPO – Intellectual Property for Engineers – www.youtube.com/@WIPOChannel

- YouTube: How to Write a Patent Application – Patent File – search on YouTube
- YouTube: Scholarly Writing Tips – Academic English – search on