

Punyashlok Ahilyadevi Holkar Solapur University, Solapur



Name of the Faculty: Science & Technology

(As per New Education Policy 2020)

Subject: - Electronics and Computer Engineering

Name of the Course: T.Y.B.Tech.(Sem.–V & VI)

(Syllabus to be implemented from-2026-27)



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech. (Electronics and Computer Engineering)

NEP 2020 Compliant Curriculum With
effect from 2026-2027

SEMESTER - V

Distribution	Course Code	Name of the Course	Engagement Hours			Credits	FA	SA			Total
			L	T	P		ESE	ISE	ICA	OE/ POE	
PCC	ECEPCC-07	Analog and Digital Communication	3			03	70	30			100
PCC	ECEPCC-08	Database Management System	3		2	04	70	30	25		125
PCC	ECEPCC-09	Embedded System	3		2	04	70	30	25	25	150
PEC	ECEPEC-01	Programme Elective Course-I	3		2	04	70	30	25		125
AEC	AEC-02	Creativity and Design Thinking	1		2	02	50*		25		75
OE	OE-03	Interdisciplinary Mini Project	1		2	02			25	25	50
MDM	ECEMDM-03	ECE MD Minor-III	2		2	03	70	30	25		125
		Total	16		12	22	400	150	150	50	750

*** For AEC-02: MCQ- based examination to be conducted.**

PCC- Programme Core Course

PEC- Programme Elective Course

AEC- Ability Enhancement Course

OE- Open Elective

CC- Co-curricular Courses

MDM-Multidisciplinary Minor: It should be selected from other UG Engineering Minor Programme



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech. (Electronics and Computer Engineering)

NEP 2020 Compliant Curriculum

With effect from 2026-2027

SEMESTER - VI

Distribution	Course Code	Name of the Course	Engagement Hours			Credits	FA	SA			Total
			L	T	P		ESE	ISE	ICA	OE/ POE	
PCC	ECEPCC-10	Real Time Operating System	2			02	70	30			100
PCC	ECEPCC-11	Software Testing & Quality Assurance	2		2	03	70	30	25	25	150
PCC	ECEPCC-12	System Design using Arduino & Raspberry Pi	3		2	04	70	30	25		125
PEC	ECEPEC-02	Programme Elective Course-II	3		2	04	70	30	25	25	150
PEC	ECEPEC-03	Programme Elective Course-III	3	1		04	70	30	25		125
SEC	ECESec-02	Projects on Industrial Application			4	02			25	50	75
MDM	ECMDM-04	ECE MD Minor-IV	2		2	03	70	30	25		125
		Total	15	1	12	22	420	180	150	100	850

PCC- Programme Core Course

PEC- Programme Elective Course

AEC- Ability Enhancement Course

OE- Open Elective

CC- Co-curricular Courses

MDM-Multidisciplinary Minor: It should be selected from other UG Engineering Minor Programme

Programme Elective Course-I

Course Code	Name of the Course
ECEPEC – 01A	Machine Learning
ECEPEC – 01B	Cyber Security
ECEPEC – 01C	VLSI Design
ECEPEC – 01D	Digital Signal Processing

Programme Elective Course-II

Course Code	Name of the Course
ECEPEC – 02A	Deep Learning
ECEPEC – 02B	Data Visualization (Power BI/Tableau)
ECEPEC – 02C	Electronic System Design
ECEPEC – 02D	Digital Image Processing

Programme Elective Course-III

Course Code	Name of the Course
ECEPEC – 03A	Principles of Soft Computing
ECEPEC – 03B	Mobile Application Development using Android
ECEPEC – 03C	Blockchain Technology
ECEPEC – 03D	Computer Vision

ECEHON: ECE Honors:**ECE Honors in AI Technology:**

Semester	Course Code	Name of the Course	Engagement Hours			Credits	FA	SA		Total
			L	T	P		ESE	ISE	ICA	
V	ECEHON-03A	Soft computing	3		2	4	70	30	25	125
VI	ECEHON-04A	AI Applications	3		2	4	70	30	25	125

ECE Honors in the Internet of Things (IOT) and Smart Systems:

Semester	Course Code	Name of the Course	Engagement Hours			Credits	FA	SA		Total
			L	T	P		ESE	ISE	ICA	
V	ECEHON-03B	Cloud computing for IoT applications	3		2	4	70	30	25	125
VI	ECEHON-04B	IoT security and privacy	3		2	4	70	30	25	125

ECE Honors in Railway Engineering:

Semester	Course Code	Name of the Course	Engagement Hours			Credits	FA	SA			Total
			L	T	P			ESE	ISE	ICA	
V	ECEHON-03C	Applications of IT and Control Engineering in Railway	3		2	4	70	30	25	125	
VI	ECEHON-04C	Advanced Communication and Modern Signaling in Railway	3		2	4	70	30	25	125	

ECE Honors in Renewable Energy and Power Electronics:

Semester	Course Code	Name of the Course	Engagement Hours			Credits	FA	SA			Total
			L	T	P			ESE	ISE	ICA	
V	ECEHON-03D	Grid integration and smart grids	3		2	4	70	30	25	125	
VI	ECEHON-04D	AI in renewable energy optimization	3		2	4	70	30	25	125	

ECE Multidisciplinary Minor in Advanced Embedded Systems:

Semester	Course Code	Course Title
V	ECEMDM-03A	Microcontrollers and Applications
VI	ECEMDM-04A	Internet Of Things (IoT)

ECE Multidisciplinary Minor in Artificial Intelligence:

Semester	Course Code	Course Title
V	ECEMDM-03B	Fundamentals of Deep Learning
VI	ECEMDM-04B	Data Visualization Tools (Tableau & Power BI)



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-V

ECEPCC-07 – Analog and Digital Communication

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

Introduction:

This course helps students understand how information is transmitted using analog and digital communication systems. It explains key concepts like modulation, noise, sampling, and digital transmission in a simple and practical way. The subject builds a strong foundation for designing and analyzing real-world communication systems used in today's technology.

Prerequisite: Students shall have knowledge of Signals and Systems, Electronic Circuits or Network Theory, and Digital Electronics.

Course Objectives:

The course aims to:

1. To remember basic elements of analog communication systems including modulation techniques.
2. To understand principles of amplitude and angle modulation schemes.
3. To apply mathematical analysis to evaluate modulation efficiency and bandwidth requirements.
4. To analyze noise performance in analog and digital communication systems.
5. To design simple analog and digital modulation/demodulation circuits.
6. To evaluate digital modulation techniques like PSK and FSK for practical applications

Course Outcomes:

At the end of the course students will be able to

1. Explain analog modulation processes and their mathematical representations.
2. Analyze bandwidth and power efficiency of AM, FM, and digital schemes.
3. Design and implement basic communication systems using modulation techniques.
4. Evaluate performance of communication systems under noisy conditions.
5. Compare analog and digital communication advantages for specific scenarios.
6. Create simulation models for digital modulation schemes like ASK and QAM.

SECTION-I

Unit 1: Introduction and AM

(07)

Elements of communication systems, types of signals, amplitude modulation: DSB-FC, DSB-SC, SSB; modulation index, demodulation using envelope detector, derivations of modulation index and power relations.

Unit 2: Angle Modulation

(07)

Frequency modulation, phase modulation, FM generation (direct/indirect), demodulation (PLL, Foster-Seeley), bandwidth (Carson's rule), narrowband/wideband FM comparison, pre-emphasis/de-emphasis.

Unit 3: Noise in Analog Systems

(07)

Noise types, SNR, noise in AM/FM, threshold effect, figure of merit for modulation schemes.

SECTION-II

Unit 4: Sampling and Pulse Modulation

(07)

Sampling theorem, PAM, PWM, PPM, quantization (uniform/non-uniform with examples), PCM, DPCM, delta modulation, quantization noise analysis.

Unit 5: Digital Modulation

(07)

ASK, FSK, PSK, DPSK, QPSK, QAM basics; line coding (NRZ/AMI); constellation diagrams, coherent/non-coherent detection (MATLAB simulation concepts).

Unit 6: Information Theory Basics

(07)

Entropy, channel capacity, Shannon's theorem, error control coding: linear block codes, systematic codes, parity-check matrix, syndrome testing, Hamming code examples

.....

Text Books:

1. Principles of Communication Systems by H. Taub, D.L. Schilling, G. Singh (4th Edition, McGraw-Hill, 2013).
2. Communication Systems by B.P. Lathi, Zhi Ding (4th Edition, Oxford University Press, 2010).
3. Electronic Communication Systems by G. Kennedy, B. Davis (5th Edition, McGraw-Hill).

.....

Reference Books:

1. Modern Digital and Analog Communication Systems by B.P. Lathi (Oxford University Press).
2. Digital Communications by J.G. Proakis, M. Salehi (5th Edition, McGraw-Hill, 2008).
3. Analog and Digital Communication by K. Sam Shanmugam (Wiley).
4. Communication Systems: Analog and Digital by R.P. Singh, S.D. Sapre (McGraw-Hill).



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-V

ECEPCC-08 –Database Management System

Teaching Scheme

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

This course introduces the fundamentals of Database Management Systems and their role in managing data efficiently. It covers database design, SQL, normalization, transactions, and basic NoSQL concepts. The subject equips students with practical skills to design and handle real-world database applications.

Prerequisite: Students shall have knowledge of basic computer architecture, data structures, and algorithms.

Course Objectives:

The course aims to:

1. To introduce the fundamental concepts, architecture, and applications of database management systems.
2. To develop the ability to design databases using Entity–Relationship modeling and relational foundations.
3. To enable students to formulate database queries using relational algebra and Structured Query Language (SQL).
4. To understand database design principles, normalization techniques, and transaction management.
5. To introduce NoSQL databases and their applications in large-scale and real-time systems

Course Outcomes:

At the end of the course students will be able to

1. Explain the need, applications, and architecture of DBMS.
2. Design ER diagrams and map them to relational schemas.
3. Write SQL queries using DDL, DML, DCL, and TCL commands.
4. Apply normalization techniques to relational database design.
5. Describe transaction processing, concurrency control, and recovery mechanisms.

SECTION-I

Unit I: Introduction to Database Systems

(04)

Database concepts, characteristics of databases, file system vs DBMS, data abstraction, data independence, DBMS architecture, roles of DBA.

Unit II: Data Modeling and Relational Foundations

(10)

Entity–Relationship (ER) Model: entity types, attributes, keys, weak and strong entities, relationship types, constraints (cardinality and participation), ER diagram design. Relational Model: relational model concepts, relational schema, types of keys, integrity constraints. Mapping and Query Foundations: mapping ER model to relational schema, introduction to relational algebra operators, relational algebra queries.

Unit III: Structured Query Language (SQL)**(07)**

SQL overview, DDL commands, DML commands, integrity constraints, aggregate functions, joins, nested queries, views, triggers.

SECTION-II**Unit IV: Database Design and Normalization****(06)**

Functional dependencies, first normal form (1NF), second normal form (2NF), third normal form (3NF), Boyce–Codd Normal Form (BCNF).

Unit V: Transaction Management and Concurrency Control**(06)**

Transaction concepts, ACID properties, transaction states, concurrency problems, serializability, lock-based protocols, deadlocks, recovery techniques.

Unit VI: NoSQL Databases**(09)**

Introduction to NoSQL, limitations of relational databases, characteristics of NoSQL systems, CAP theorem, types of NoSQL databases: key-value stores, document-oriented databases, column-family stores, graph databases. Comparison of SQL vs NoSQL. Use cases of NoSQL in real-time and large-scale applications.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

1. Implement SQL DML commands (INSERT, UPDATE, DELETE and SELECT) on a database.
2. Draw an E-R diagram for a real-world database application.
3. Write simple SQL queries on a database schema for a specific application.
4. Write SQL queries using aggregate functions and nested subqueries.
5. Write SQL queries using views and different types of joins.
6. Implement SQL integrity constraints and authorization commands.
7. Implement embedded SQL, functions, procedures, and triggers.
8. Write a Java program for database connectivity using JDBC.
9. Normalize a database into 1NF, 2NF, 3NF, and BCNF.
10. Implement dynamic hashing on a database for efficient record operations.

Text Books:

1. Abraham Silberschatz, Henry F.Korth, S. Sudarshan Database System Concepts, McGraw-Hill, 6th Edition
2. Ramez Elmasri, Shamkant B. Navathe Fundamentals of Database Systems, Pearson Education
3. Raghu Ramakrishnan, Johannes Gehrke Database Management Systems, McGraw-Hill

Reference Books:

1. Henry Korth, Abraham Silberschatz, S. Sudarshan Database System Concepts, McGraw-Hill, 6th Edition
2. J. D. Ullman Principles of Database Systems, Galgotia Publications James R. Groff, Paul N. Weinberg, Andy Opper SQL: The Complete Reference, 3rd Edition
3. Peter Rob, Carlos Coronel Database Systems: Design, Implementation, and Management, Cengage Learning, 9th Edition



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-V

ECEPCC-09-Embedded System

Teaching Scheme

Lectures : - 3 Hrs/Week, 3 credits

Practical : - 2 Hrs/Week, 1 credit

Examination Scheme

ESE: 70 Marks

ISE : 30 Marks

ICA : 25 Marks

POE : 25 Marks

Introduction:

This course helps students understand how embedded systems work and where they are used in everyday devices. It focuses on the 8051 microcontroller, programming in Assembly and Embedded C, and connecting components like LEDs, sensors, and communication modules. By the end of the course, students will be able to build and debug simple real-time embedded projects with confidence.

Prerequisite: Students should have basic knowledge of C/C++ programming and Assembly language. Understanding of digital electronics concepts such as logic gates, number systems, and microprocessor basics is required.

Course Objectives:

The course aims to:

1. Understand core concepts, categories, and applications of embedded systems.
2. Explain microcontroller architecture, memory systems, and instruction execution.
3. Interface digital/analog peripherals using standard communication protocols.
4. Apply real-time operating system principles and develop task-based firmware.
5. Employ systematic design and debugging practices for embedded projects.
6. Analyze real-world embedded applications and emerging trends in the domain.

Course Outcomes:

At the end of the course students will be able to:

1. Describe the **fundamental concepts, architecture, and applications** of embedded systems.
2. Explain the **internal architecture and operation of microcontrollers**, including memory organization and instruction execution.
3. Apply **embedded C/assembly programming** to interface peripherals such as GPIO, timers, ADC, and communication modules.
4. Analyze and implement **serial communication protocols** (UART, SPI, I2C) and interrupt-driven systems for real-time applications.
5. Evaluate **embedded system design constraints** such as power consumption, performance, cost, and reliability using RTOS concepts.
6. Design and develop a **functional embedded system solution** for real-world applications using appropriate hardware and software tools.

SECTION-I

Unit 1: Introduction to Embedded Systems

(06)

Definition and characteristics of embedded systems, Embedded system components: hardware and software, Embedded vs general-purpose computing systems, Classification of embedded systems, Applications of embedded systems, Overview of embedded system design flow.

Unit 2: 8051 Microcontroller Architecture (07)
Evolution of microcontrollers, Features of 8051 microcontroller, Pin configuration and functions, internal architecture of 8051, Register organization, Memory organization: Program memory, Data memory, SFRs; I/O ports and their structure

Unit 3: 8051 Instruction Set & Assembly Language Programming (08)
Instruction format and addressing modes, Data transfer instructions, Arithmetic and logical instructions, branching and looping instructions, Bit-level programming, Simple Assembly language programs, Introduction to assembler and development tools.

SECTION-II

Unit 4: Embedded C Programming for 8051 (07)
Introduction to Embedded C, Data types and memory models, I/O port programming using C, Delay generation, Interrupt programming using C, Comparison of Assembly vs Embedded C, Keil C compiler.

Unit 5: Timers, Interrupts & Serial Communication (07)
Timers and counters: modes and applications, Interrupt structure and priority levels, External and internal interrupts, Serial communication fundamentals, Serial ports and registers of 8051, Baud rate generation, Interfacing with PC using UART.

Unit 6: Interfacing & Embedded System Applications (07)
Interfacing LEDs, switches, relays, LCD interfacing (16×2), ADC interfacing (e.g., ADC0804), Sensor interfacing (temperature, IR, LDR), Stepper/DC motor interfacing, Temperature monitoring system.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

Suggested Experiments

1. Study of 8051 microcontroller architecture and pin diagram
2. LED blinking using 8051 (Assembly/C)
3. Switch interfacing and LED control
4. Delay generation using timers
5. Timer-based square wave generation
6. External interrupt programming
7. Serial communication between 8051 and PC
8. LCD interfacing with 8051
9. ADC interfacing and analog data display
10. Temperature sensor interfacing
11. DC/Stepper motor control using 8051
12. Mini-project based on 8051 (automation/monitoring)

Text Books:

1. **Muhammad Ali Mazidi, Janice Gillispie Mazidi, Rolin D. McKinlay**
The 8051 Microcontroller and Embedded Systems – Pearson Education
2. **Raj Kamal**
Embedded Systems: Architecture, Programming and Design – McGraw Hill

Reference Books:

1. **Kenneth J. Ayala**
The 8051 Microcontroller – Penram International
2. **David Calcutt, Fred Cowan, Hassan Parchizadeh**
8051 Microcontroller: An Application-Based Introduction – Newnes
3. 8051 Manufacturer Datasheets (Intel/Atmel)



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-V

Programme Elective Course - I
ECEPEC 01A MACHINE LEARNING

Teaching Scheme:

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

Machine Learning is a branch of Artificial Intelligence that enables systems to learn from data and improve performance without being explicitly programmed. This course introduces fundamental concepts, algorithms, models, and practical applications of Machine Learning used in real-world domains such as healthcare, finance, robotics, and data analytics.

Course Prerequisite:

Students should have basic knowledge of:

- Mathematics (Linear Algebra, Probability, and Statistics)
- Programming (preferably Python)
- Basic understanding of Data Structures and Algorithms

Course Objectives:

1. To introduce various types of machine learning algorithms.
2. To enable designing of a model selecting appropriate machine learning algorithms for a given Problem.
3. To study methods to validate previously designed machine learning models.
4. To introduce methods to evaluate and tune machine learning models.

Course Outcomes:

At the end of the course students will be able to:

1. Demonstrate types of machine learning algorithms.
2. Design a model by selecting appropriate machine learning algorithm for a given problem.
3. Validate designed machine learning model.
4. Evaluate and tune machine learning model based on various parameters.
5. Design various applications using machine learning algorithm.

SECTION I

Unit 1: Introduction to Machine learning

(07)

Understanding Machine Learning: What Is Machine Learning?, Leveraging the Power of Machine Learning, The Roles of Statistics and Data Mining with Machine Learning, Putting Machine Learning in Context, Types of machine Learning, Applications of Machine Learning. Applying Machine Learning: Getting Started with a Strategy, Applying Machine Learning to Business Needs.

Unit 2: Offerings of Machine learning

(07)

The Impact of Machine Learning on Applications, Data Preparation, the Machine Learning Cycle. Getting Started with Machine Learning: Understanding How Machine Learning Can Help, Focus on the Business Problem, Requirement of Collaboration in Machine Learning, Executing a Pilot

Project, Determining the Best Learning Model.

Unit 3: Basic mathematics for Machine learning (07)

Getting Started with The Math Basics, Working with Data, Exploring the World of Probabilities, Describing the Use of Statistics, Interpreting Learning As Optimization, Exploring Cost Functions, Descending the Error Curve.

SECTION II

Unit 4: Validating Machine Learning Models (09)

Validating Machine Learning: Checking Out-of-Sample Errors, Getting to Know the Limits of Bias, Keeping Model Complexity in Mind and Solutions Balanced, Training, Validating, and Testing, Resorting to Cross Validation. Looking for Alternatives in Validation. Optimizing Cross-Validation Choices, Avoiding Sample Bias and Leakage Traps, Discovering the Incredible Perceptron Simplest learning strategies to learn from Data: Discovering the Incredible Perceptron

Unit 5: Improving Machine Learning Models (08)

Improving Machine Learning Models, Studying Learning Curves, Using Cross-Validation Correctly, Choosing the Right Error or Score Metric, Searching for the Best Hyper-Parameters, Testing Multiple Models, Averaging Models, Stacking Models, Applying Feature Engineering, Selecting Features and Examples.

Unit 6: Applications of Machine Learning (04)

Applying Learning to Real Problems, Classifying Images, Scoring Opinions and Sentiments, Recommending Products and Movies, Using Machine Learning to Provide Solutions to Business Problems, Future of Machine Learning.

Internal Continuous Assessment (ICA):

Student should implement the following (Use Python or R)

1. Basic mathematics for Machine Learning – Simulating solutions using Python to
 - i. Matrix operations
 - ii. Problems using Probability
 - iii. Statistical Estimations.
2. Introduction to Jupyter Notebook and Colab.
3. Working with data.
4. Data Exploration and Preprocessing.
5. Linear Regression
6. Introduction to Dimensionality Reduction
7. Logistic Regression
8. Decision Trees
9. Random Forest
10. Clustering (Unsupervised Learning)

Text Books:

1. Machine Learning For Dummies, IBM Limited Edition by Judith Hurwitz, Daniel Kirsch (Published by Wiley, First edition).
2. Machine Learning for Dummies by John Paul Mueller, Luca Massaron (Published by For Dummies; First edition).

Reference Books:

1. Introduction to Machine Learning (Second Edition) by Ethem Alpaydm (published by The MIT Press Cambridge, Massachusetts London, England
2. Machine Learning by Tom M. Mitchell (Publisher: McGraw Hill Education; First edition + New Chapters from Second edition).



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-V

Programme Elective Course - I

ECEPEC 01B Cyber Security

Teaching Scheme:

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

This course provides a comprehensive introduction to the fundamental principles of cyber security, focusing on protecting networks, systems, and data from digital attacks. It covers essential concepts such as cryptography, network security protocols, web security, and legal frameworks to prepare students for the evolving threat landscape in the digital era.

Course Prerequisite:

Students should have basic knowledge of:

- Computer Networks and Data Communication.
- Operating Systems fundamentals.
- Basic programming concepts (Python/C)

Course Objectives

The course aims to:

1. To understand the fundamental concepts and terminology of cyber security and cryptography.
2. To explore various types of cyber threats, vulnerabilities, and attack vectors.
3. To analyze and implement network and web security protocols.
4. To introduce digital forensics and incident response strategies.
5. To familiarize students with cyber laws, ethics, and privacy regulations.

Course Outcomes

Upon completion of the course, students will be able to:

1. Identify and explain various cyber security threats and vulnerabilities.
2. Apply cryptographic techniques to ensure data confidentiality and integrity.
3. Design and evaluate secure network architectures using firewalls and IDS.
4. Analyze web-based vulnerabilities and implement defensive measures.
5. Understand the legal and ethical implications of cyber-attacks under current regulations.

SECTION I

Unit 1: Introduction to Cyber Security and Cryptography

(07 hrs)

Cyber Security overview: Definition, CIA Triad (Confidentiality, Integrity, Availability), Cyber-attacks vs. Information Security. Introduction to Cryptography: Symmetric and Asymmetric Key Cryptography, DES, AES, RSA algorithms, Digital Signatures, and Hash functions.

Unit 2: Cyber Threats and Vulnerabilities

(07 hrs)

Types of Cyber-attacks: Malware (Viruses, Worms, Trojans), Phishing, Social Engineering, DoS/DDoS attacks, Man-in-the-Middle attacks. Vulnerability assessment: Understanding common vulnerabilities (CVE), port scanning, and footprinting.

Unit 3: Network Security (07 hrs)

Network layer security: IPsec, VPNs. Transport layer security: SSL/TLS. Firewalls: Types and configuration, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS), Honeypots

SECTION – II

Unit 4: Web and Application Security (07 hrs)

Web security threats: SQL Injection, Cross-Site Scripting (XSS), Session Hijacking. Secure coding practices, HTTPS protocol, Web Application Firewalls (WAF), and secure API development.

Unit 5: Digital Forensics and Incident Response (07 hrs)

Introduction to Digital Forensics: Evidence collection, preservation, and analysis. Incident Response Life Cycle: Preparation, detection, containment, eradication, and recovery. Tools for digital forensics.

Unit 6: Cyber Laws, Ethics, and Privacy (07 hrs)

Cyber Law overview: IT Act 2000 and its amendments, Intellectual Property Rights (IPR). Ethical Hacking vs. Cyber Crime. Privacy in the digital age: Data protection regulations (GDPR, Digital Personal Data Protection Act).

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

Suggested List of Practical:

1. Classical Encryption Techniques:
Implementation of traditional substitution and transposition ciphers, specifically the Caesar Cipher and Playfair Cipher, using C++ or Python.
2. Symmetric Key Cryptography:
Demonstration and implementation of the Advanced Encryption Standard (AES) algorithm to encrypt and decrypt data files using Python libraries (like cryptography or PyCryptodome).
3. Asymmetric Key Cryptography:
Implementation of the RSA algorithm, including key generation (public and private keys), encryption of a message, and decryption using the private key.
4. Network Reconnaissance and Scanning:
Using Nmap to perform host discovery, port scanning, and service version detection. Students will analyze open ports and potential entry points in a controlled environment.
5. Traffic Analysis and Packet Sniffing:
Capturing and analyzing network traffic using Wireshark. Students will identify protocols (HTTP, FTP, TCP) and observe how sensitive data appears in "plain text" over unencrypted connections.
6. Firewall Configuration and Rules:
Setting up and configuring a software firewall (such as Windows Firewall or iptables/UFW in Linux) to allow or block specific IP addresses, ports, and protocols.
7. Social Engineering Simulation:
Simulating a Phishing attack using tools like the Social-Engineer Toolkit (SET). The focus will be on identifying "red flags" in URLs/emails and documenting preventive measures.

8. Web Application Security (SQL Injection):
Identifying and demonstrating an SQL Injection (SQLi) vulnerability on a test website (like DVWA or a local lab setup) and implementing "Prepared Statements" to fix it.
9. Web Application Security (Cross-Site Scripting):
Demonstration of Stored or Reflected XSS attacks. Students will learn how malicious scripts are executed in a user's browser and how to implement input validation.
10. Digital Forensics – Data Recovery:
A practical exercise in recovering deleted files from a corrupted or formatted disk image using forensic tools like Autopsy or FTK Imager.
11. Password Cracking and Hash Analysis:
Understanding how passwords are stored as hashes (MD5/SHA-256) and using tools like John the Ripper or Hashcat to demonstrate the importance of "salting" and strong password policies.
12. Case Study – Real-World Breach Analysis:
A comprehensive analysis of a major historical cyber-attack (e.g., The WannaCry Ransomware or The Equifax Breach). Students must document the attack vector, the impact, and the security failures that led to the incident.

- **Text Books:**

1. Cryptography and Network Security, William Stallings, Pearson Education.
2. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Nina Godbole and Sunit Belapure, Wiley India.

- **Reference Books:**

1. Principles of Information Security, Michael E. Whitman and Herbert J. Mattord, Cengage Learning.
 2. Applied Cryptography, Bruce Schneier, Wiley.
 3. The IT Act, 2000, Government of India.
-



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-V

Programme Elective Course - I

ECEPEC-01C: VLSI Design

Teaching Scheme:

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

This course introduces how to design, simulate and test digital logic circuits using the hardware description language of Verilog HDL and CMOS logic. It also introduces the CPLD and FPGA architectures used to implement the digital logic circuits.

Course Prerequisite:

Students shall have knowledge of Digital Devices, combinational logic circuit design, and simulation.

Course Objectives:

1. To make students learn EDA Tools for Verilog HDL programming and CMOS Logic Design and simulation
2. To enable students to design Verilog HDL modules for combinational and sequential logic circuits.
3. To acquaint students with CPLD and FPGA architecture.
4. To introduce students to MOS Transistor Theory and CMOS Logic-based design of combinational logic circuits.

Course Outcomes:

At the end of the course, students will be able to -

1. Explain the different syntax of Verilog HDL language.
2. Analyze combinational circuits using Verilog HDL.
3. Analyze sequential logic circuits using Verilog HDL.
4. Describe MOS transistor theory and behavior of E-MOSFET
5. Analyze combinational logic circuit design using E-MOSFETs.
6. Describe the architecture and internal components of CPLD and FPGA.

Section I

Unit 1- Introduction to Verilog HDL

(08)

Structure of Verilog module, Operators, Data Types, Styles of Description. Verilog Models for Gate Propagation Delay (Inertial Delay), Time Scales for Simulation, Verilog Models for Net Delay (Transport Delay), Module Paths and Delays, Path Delays and Simulation, Inertial Delay Effects and

Pulse Rejection, Examples using Verilog.

Unit 2- Combinational logic using Behavioral Descriptions (07)

Structure, Variable Assignment Statement, Sequential Statements, Loop Statements, Verilog Behavioral Description of Adder, Subtractor, Comparator, Multiplexer, Demultiplexer, Encoder, Decoder.

Unit 3 – Sequential logic using Behavioral Descriptions (06)

Behavioral models of Flip-flops, counters and shift registers, Verilog HDL model using state machine for sequence detectors

Section II

Unit 4 - MOS Transistor Theory (07)

Physical structure of MOS transistor, accumulation, depletion & inversion modes, MOS device design equations, second order effects, Technology scaling, Static and dynamic behavior of CMOS inverter, power and energy delay, impact of technology scaling on inverter.

Unit 5 – Combinational Logic Design in CMOS (08)

Static CMOS design- complementary CMOS, Implementation of Boolean Expressions using CMOS Logic, Ratioed logic and pass transistor logic; dynamic CMOS design- dynamic logic basic principle, speed and power dissipation, issues in dynamic design, cascading dynamic gates, comparison of static and dynamic designs in CMOS, Timing Issues in Digital Circuits.

Unit 6: Architecture of Commercial Devices (06)

CPLD Architecture, Xilinx XC9500, Altera Max7000, FPGA organization and architecture, Altera Flex 10k.

Internal Continuous Assessment:

ICA shall be based on a minimum of eight experiments based on the above syllabus using any EDA software tool for Verilog HDL modules and CMOS logic design.

Suggested List of experiments:

I. Design and Implementation of the following using Verilog HDL and write test bench for-

1. Design of half adder and full adder
2. Design of 4 bit adder using structural style modelling
3. Design of code converters
4. Design of comparators
5. Design of encoder and decoder
6. Design of multiplexer and Demultiplexers
7. Design of flip flops
8. Design of asynchronous and synchronous counters
9. Design of sequence detector using state machine
10. Design of ADD and Shift multiplier using state machine

II. Design and Implementation of the Following using CMOS / Dynamic CMOS Logic-

1. Logic Gates
2. Universal Logic Gates
3. Boolean Expression
4. Half adder and full adder
5. Half subtractor and full subtractor

Text Books:

1. Digital Systems Design using Verilog, Charles H. Roth, Lizy Kurian John, Byeong Kil Lee- Cengage Learning.
 2. HDL Programming VHDL And Verilog, Nazeih M.Botros, Dreamtech Press
 3. HDL with Digital Design: VHDL and Verilog, Nazeih Botros. Mercury Learning and Information LLC. ISBN: 978-1-938549-81-6
 4. Modeling, Synthesis and Rapid Prototyping with the Verilog HDL, M.D. CILETTI, Prentice- Hall.
 5. Digital Integrated Circuits, Rabey, Chandrakasan, Nikolic, Pearson Education
 6. CMOS VLSI design, Neil H. E. Weste, David Harris, Ayan Banerjee, Pearson Education
-

Reference Books:

1. Digital Design Principles and Practices, John F. Wakerly, Printice Hall, 3rd Edition.
 2. Datasheets of CPLDs and FPGAs.
 3. CMOS digital integrated circuits, Analysis and Design, Sung-Mo Kang, Yusuf Leblebici, TATA McGRAW Hill
 4. Principles of CMOS VLSI Design, Neil Weste, Kamran Eshraghian Addison Wesley/Pearson Education
 5. Modern VLSI Design, Wayne Wolf, 2nd Edition, Prentice Hall, 1998
 6. Essentials of VLSI Circuits and Systems, Kamran Ehraghian, Dauglas A. Pucknell and Sholeh Eshraghian, PHI, EEE, 2005 Edition
-



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-V

Programme Elective Course - I

ECEPEC 01D Digital Signal Processing

Teaching Scheme:

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

Digital Signal Processing (DSP) deals with the analysis, manipulation, and processing of signals in digital form. This course covers discrete-time signals and systems, transforms, digital filter design, and practical DSP applications in communication, audio processing, image processing, and biomedical systems.

Course Prerequisite:

Students should have basic knowledge of:

- Engineering Mathematics (Laplace Transform, Fourier Series, Fourier Transform)
- Signals and Systems
- Basic programming concepts

Course Objectives:

The course aims to:

1. To introduce the fundamental principles of Digital Signal Processing and enable frequency-domain analysis of discrete-time signals using the Discrete Fourier Transform (DFT).
2. To introduce the application of the Discrete Fourier Transform (DFT) and Inverse Discrete Fourier Transform (IDFT) for performing linear filtering of discrete-time signals.
3. To introduce efficient algorithms for computing the Discrete Fourier Transform using Fast Fourier Transform (FFT) techniques such as DIT and DIF.
4. To develop skills in the design and implementation of digital filters (FIR and IIR) for various signal processing applications.
5. To introduce the implementation of discrete-time systems using direct, cascade, and parallel realization structures for efficient DSP system design.

Course Outcomes:

Upon completion of the course, students will be able to:

1. **Apply** fundamental principles of Digital Signal processing to analyze discrete-time signals and systems in both time and frequency domains using the DFT.
 2. **Analyze** and **Solve** problems related to long sequence filtering using **Overlap - Save** and **Overlap - Add** methods.
 3. **Explain and implement** FFT algorithms, such as DIT and DIF, to efficiently compute the Discrete Fourier Transform.
 4. **Design and analyze** Finite Impulse Response (FIR) digital filters using standard techniques to meet given frequency-domain specifications.
 5. **Design and analyze** (IIR) digital filters using the Impulse Invariance Method (IIM) and Bilinear Transformation (BLT) to meet given frequency specifications.
 6. **Implement and compare** different structures for the realization of digital signal processing systems to achieve efficient and stable system performance.
-

SECTION I

Unit 1: Introduction to DSP and DFT (Discrete Fourier Transform) (08)

Basic elements of DSP, Advantages of DSP over ASP, Correlation, Frequency domain sampling and reconstruction of discrete-time signals, Computation of DFT & IDFT, DFT as a linear transformation, Properties of DFT, Circular Convolution using DFT & IDFT, Matrix and Concentric Method, Relation between DFT and Z transform.

Unit 2: Linear filtering using DFT & IDFT (06)

Linear filtering using DFT & IDFT, Long Sequence Filtering Methods:

1) Overlap-Save Method

2) Overlap-Add Method, Frequency analysis of signals using DFT

Unit 3: FFT (Fast Fourier Transform) Algorithm (07)

Introduction to FFT, Bit Reversal Algorithm, Radix-2 FFT algorithm for the computation of DFT and IDFT, Decimation In Time (DIT) and Decimation In Frequency (DIF) algorithms. Difference between DFT and FFT.

SECTION II

Unit 4: FIR Filter Design (07)

Introduction to FIR filters, Gibbs phenomenon, Design of FIR filters using Fourier Series Method Frequency Sampling Method and Windows techniques (Rectangular, Hamming, Hanning, Trigular, and Blackmann).

Unit 5: IIR Filter Design (07)

Introduction to analog filter design, Characteristic of Butterworth filter, IIR Filter Design by Impulse Invariance Method (IIM), Bilinear Transformation (BLT), Warping effect. (Problems on filter design). Difference between FIR and IIR.

Unit 6: Implementation of Discrete Time Systems (07)

FIR Filter Design: Direct Form, Cascade Form Structure

IIR Filter Design: Direct Form, Cascade Form, Parallel form

Applications of DSP in Audio and Speech Processing, Image and Video Processing, Biomedical Signal Processing.

Internal Continuous Assessment (ICA):

ICA consists of **minimum eight** experiments based upon above curriculum.

Suggested List of Practical

1. Generation and analysis of basic discrete-time signals (unit impulse, step, ramp, sine, exponential).
2. Sampling and reconstruction of analog signals.
3. Computation of Cross/Auto correlation between two discrete-time signals.
4. Linear and circular convolution of discrete-time signals.
5. Verification of properties of Discrete-Time Fourier Transform (DTFT).
6. Computation of Discrete Fourier Transform (DFT) and Inverse DFT (IDFT).
7. Implementation of Fast Fourier Transform (FFT) algorithms (DIT and DIF).
8. Linear filtering using DFT and IDFT.
9. Design and analysis of FIR digital filters using window methods.
10. Design and analysis of IIR digital filters using Impulse Invariance Method (IIM).
11. Design and analysis of IIR digital filters using Bilinear Transformation (BLT).
12. Realization of DSP systems using direct, cascade, and parallel forms.

13. Application-based experiment such as noise removal from audio signal processing.

• **Recommended Text Books:**

1. Digital Signal Processing – Principles, Algorithms and Applications John G Proakis-4th edition, Pearson Education
 2. Digital Signal Processing by S Salivahanan, A Vallavaraj& C Gnanapriya –2nd edition, TMH.
 3. Discrete time signal Processing A.V. Oppenheim & R.W. Schafer.- Low price edition, John Wiley
-

• **Recommended Reference Books:**

1. Digital Signal Processing Ramesh Babu -4th Edition, Scientific Publication.
 2. Digital Signal Processing Dr. Shaila D. Apte, Second edition, Wiley India.
 3. Essentials of Digital Signal Processing using MATLAB Vinay K. Ingle & John G. Proakis, Cengage Learning, 2012
 4. Digital Signal Processing- A Practical Approach, E. C. Ifleachor and B. W. Jervis, Second Edition, Pearson education.
 5. Digital Signal Processing S. Palani& D. Kalaiyarasi, Ane's Student Edition, Ane Books Pvt. Ltd New Delhi
-



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-V

AEC-02: Creativity and Design Thinking

Teaching Scheme:

Lecture: 1 hrs/week, 1 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 50* Marks

ICA : 25 Marks

Introduction:

This course is intended for students from any discipline who require an understanding of design thinking for brand, product, and service development. Students will learn a series of design thinking concepts, methods and techniques that are used to bring about innovation in business and in the social sector

Course Prerequisite: Students shall have communication and collaboration skills, openness to creativity and innovation.

Course Objectives:

1. Understand the importance of creativity and design thinking in Engineering.
2. Apply design thinking principles to problem-solving in various problems.
3. Develop user-centric solutions using ideation and prototyping techniques.
4. Use various creativity-enhancing techniques for innovative technology solutions.
5. Implement usability testing, feedback loops, and data-driven design decisions.
6. Work on real-world projects to apply design thinking methodologies effectively.

Course Outcomes:

Students will be able to:

1. Demonstrate an understanding of design thinking frameworks and methodologies.
2. Develop innovative and user-friendly applications using creative problem-solving.
3. Use prototyping and wire framing tools to build interfaces.
4. Conduct usability testing and refine designs based on user feedback.
5. Apply design thinking principles in computing fields.
6. Develop and present a capstone project that showcases creative and technical skills.

SECTION-I

UNIT-1 Introduction to Creativity and Design Thinking

(03)

Definition and importance of creativity in computer science, overview of the design thinking process (empathize, define, ideate, prototype, test), human-centered design approach, case studies of innovative tech products.

UNIT-2 Problem Identification and Ideation Techniques

(03)

Understanding complex engineering problems, brainstorming and idea generation techniques (SCAMPER, mind mapping, lateral thinking), TRIZ (theory of inventive problem solving), convergent vs. divergent thinking in engineering.

UNIT-3 Prototyping and Concept Development

(02)

Prototyping techniques: low-fidelity vs. high-fidelity prototypes, engineering design and product development, design for manufacturing (DFM) and sustainable engineering, role of ai and simulation in engineering prototyping.

SECTION II

UNIT-4 Innovation in Engineering Applications

(03)

Creativity in mechanical, electrical, and civil engineering, design thinking for smart cities and sustainable engineering, role of IoT, AI, and emerging technologies in engineering innovation, ethical and social considerations in engineering design

UNIT-5 Testing, Iteration, and Implementation

(02)

Usability testing and performance evaluation, failure analysis and iterative design improvement, agile methodologies and lean engineering principles, data-driven decision making in engineering design

UNIT-6 Capstone Project and Future Trends

(02)

Real-world engineering challenges and case studies, team-based project using design thinking methodology, emerging trends in engineering innovation, presentation and peer review of final designs.

Internal Continuous Assessment (ICA):

ICA consists of **minimum eight** experiments based upon above curriculum.

Text books:

1. Kelley, Tom & Kelley, David – Creative Confidence: Unleashing the Creative Potential Within Us All, Crown Business, 2013.
 2. Brown, Tim – Change by Design: How Design Thinking Creates New Alternatives for Business and Society, Harper Business, 2009.
 3. Plattner, Hasso, Meinel, Christoph, & Leifer, Larry – Design Thinking: Understand – Improve – Apply, Springer, 2011.
-

Reference Books:

1. **Cross, Nigel** - Design Thinking: Understanding How Designers Think and Work, Berg Publishers, 2011.
2. **Liedtka, Jeanne & Ogilvie, Tim** – Designing for Growth: A Design Thinking Toolkit for Managers, Columbia Business School Publishing, 2011.
3. **Buxton, Bill** – Sketching User Experiences: Getting the Design Right and the Right Design, Morgan Kaufmann, 2007.
4. **Norman, Donald A.** – The Design of Everyday Things, Basic Books, 2013.
5. **IDEO.org** - Design Kit: The Human-Centered Design Toolkit (Available online at www.designkit.org).



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-V

OE-03: Interdisciplinary Mini Project

Teaching Scheme

Lecture: 1Hr/Week, 1 Credit

Practical: 2 Hrs/week, 1 Credit

Examination Scheme

ICA: 25 Marks

POE: 25 marks

Introduction:

The **Interdisciplinary Mini Project** course introduces students to collaborative problem-solving across multiple disciplines. It encourages the integration of diverse knowledge areas, enabling students to apply theoretical concepts in real-world applications. Through this course, students gain hands-on experience in designing, developing, and implementing innovative solutions while fostering teamwork, creativity, and technical expertise.

Prerequisite:

An interdisciplinary mini project involves integrating knowledge from multiple fields to solve a problem or create an innovative solution.

Course Objectives

1. Identify problems and propose innovative, efficient solutions.
2. Develop project management skills, including planning, designing, and executing a small- scale project.
3. Improve proficiency in programming, hardware design, or other domain-specific skills.
4. Work effectively in a team, distributing tasks and responsibilities.
5. Encourage critical thinking and experimentation with new technologies.
6. Gain hands-on experience relevant to industry practices.
7. Consider ethical, environmental, and sustainability factors in project development.

Course Outcomes:

At the end of this course, student will be able to

1. To encourage students to work on real-world problems through interdisciplinary collaboration.
2. To integrate computing skills with other domains such as IoT, AI, Healthcare, Finance, and Sustainability.
3. To develop problem-solving, research, teamwork, and project management skills.
4. To implement industry-relevant technologies using experiential learning.
5. To prepare students for innovation, entrepreneurship, and research.

Note:

1. There should be a group of preferably 4/5 students.
2. At least two branches must be combined to form a group.
3. Students should be given projects in Hardware, Software, Embedded or any contemporary topic.
4. One guide should be allocated per group.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-V

Multidisciplinary Minor in Advanced Embedded Systems

ECEMDM-03A: Microcontrollers and Applications

Teaching Scheme

Lectures—2 Hours/week, 2 Credits
Practical – 2 Hours/week, 1 Credit

Examination Scheme

ESE –70 Marks
ISE – 30 Marks
ICA– 25 Marks

Introduction:

This course introduces students to the fundamental concepts of microcontrollers and their role in embedded systems. It covers architecture, programming (Assembly and Embedded C), peripheral interfacing, and real-time applications using 8051 and PIC microcontrollers. The course emphasizes practical implementation and system-level design for real-world embedded applications.

Course Prerequisite:

Basic knowledge of Digital Electronics, Understanding of Microprocessors (8085/8086 basics), Fundamentals of C Programming, Basic concepts of Electronic Circuits.

Course Objectives

By the end of the course, students will be able to:

1. Understand the fundamental concepts of microcontrollers, architecture, and classification.
 2. Compare and contrast CISC & RISC architectures and features of common embedded controllers.
 3. Develop and analyze assembly and C programs for microcontrollers.
 4. Design and implement I/O interfacing for various peripherals.
 5. Interpret and configure timers, interrupts, and serial communication modules.
 6. Apply microcontroller solutions to real-world embedded applications.
-

Course Outcomes (COs)

After successful completion, students will be able to:

CO1: Explain microcontroller architecture, terms, family types, and features.

CO2: Illustrate memory structures, I/O ports, and addressing modes of 8051/PIC controllers.

CO3: Develop basic assembly and embedded C programs for data processing tasks.

CO4: Analyze timer/counter, interrupt, and serial communication operations and programs.

SECTION - I

Unit 1 — Introduction to Microcontrollers

(05 hrs)

Overview of Embedded Systems and Controllers, Microprocessor vs. Microcontroller, CISC vs. RISC, Harvard & von Neumann architecture, Selection criteria for microcontrollers, Development tools and IDEs

Unit 2 — MCS-51 Architecture & Programming

(05 hrs)

8051 architecture, memory map, registers, Addressing modes, instruction set overview, Timers/Counters, serial port, and interrupt structure, Assembly level programming examples

Unit 3 — Embedded C for 8051**(05 hrs)**

C programming in embedded systems, Data types, I/O operations, timers & delay generation, Functions, pointers, interrupts in embedded C, Code generation and debugging

SECTION – II**Unit 4 — PIC Microcontroller Fundamentals****(05 hrs)**

Overview of PIC microcontrollers (e.g., PIC16F877A), Architecture, oscillator, reset, SFRs, Input/Output and timers, Introduction to PIC instruction set

Unit 5 — Peripheral Interfacing & Communication**(05 hrs)**

Interfacing LEDs, switches, LCD, keypad, ADC/DAC, sensors, relays, motor drivers, Serial communication (UART/SPIs basics), Interrupt handling and priority

Unit 6 — Embedded Applications & Design**(05 hrs)**

Design of simple embedded systems, Real-time control systems, traffic light controller, data logger, System integration

Internal Continuous Assessment (ICA)

ICA shall consist of minimum eight practicals must be conducted for the **Microcontroller** practical. Similarly, four practicals must also be conducted in line with course outcome.

Suggested List of experiments:

1. Study of Microcontroller Kits and IDE Setup
2. 8051 Assembly Program — Data Transfer Operations
3. Timer programming (8051) for Delay Generation
4. Serial communication (UART) – Basic Tx/Rx
5. Embedded C: I/O Port operations
6. LCD Interfacing & Message Display (C/Assembly)
7. Keypad Interfacing and Scanning
8. ADC Interfacing and Analog Data Read

Recommended Textbooks

1. “The 8051 Microcontroller & Embedded Systems” – Muhammad Ali Mazidi, Janice Gillispie Mazidi, Rolin McKinlay
2. “Microcontrollers – Principles and Applications” – Ajay V. Deshmukh
3. “Programming 16-bit PIC Microcontrollers in C” – Lucio Di Jasio
4. “Embedded C Programming and the 8051” – Michael J. Pont

Reference Books

- “Design with PIC Microcontrollers” – John B. Peatman
- “Microcontroller Theory and Applications” – A. V. Deshmukh
- Datasheet & Application Notes for 8051 / PIC16F877A



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-V

Multidisciplinary Minor in Artificial Intelligence

ECEMDM-03B Fundamentals of Deep learning

Teaching Scheme

Lectures–2 Hours/week, 2 Credits

Practical – 2 Hours/week, 1 Credit

Examination Scheme

ESE –70 Marks

ISE – 30 Marks

ICA– 25 Marks

Introduction:

This course provides comprehensive knowledge of deep learning concepts, neural network architectures, and modern AI frameworks. It focuses on mathematical foundations, model training, optimization techniques, and real-world applications in image processing, signal analysis, and interdisciplinary engineering domains.

Course Prerequisite:

Students shall have knowledge of basic Artificial Intelligence, Machine Learning, Reinforcement Learning, Natural Language Processing, and Python Programming.

Course Objectives:

The course aims to

1. Introduce fundamental concepts and architectures of deep learning.
2. Provide mathematical and computational foundations of neural networks.
3. Enable students to design and train deep learning models for real-world problems.
4. Familiarize students with deep learning frameworks and tools.
5. Apply deep learning techniques to image, signal, and data-driven applications.
6. Develop problem-solving and analytical skills using deep learning approaches.

Course Outcomes:

Upon completion of the course, students will be able to:

- CO1: Explain the fundamentals of neural networks and deep learning concepts.
- CO2: Implement and train deep learning models using standard frameworks.
- CO3: Analyze and optimize model performance using suitable techniques.
- CO4: Apply Convolutional Neural Networks for image and signal processing tasks.
- CO5: Apply Recurrent Neural Networks for sequence and time-series data.
- CO6: Solve interdisciplinary engineering problems using deep learning models.

SECTION I

Unit 1: Introduction to Deep Learning

(5 Hours)

Artificial Intelligence, Machine Learning, and Deep Learning, Limitations of traditional machine learning, Biological inspiration of neural networks, History and evolution of deep learning, Applications in Electronics and Computer Engineering, Overview of deep learning tools and frameworks

Unit 2: Neural Network Fundamentals

(5 Hours)

Perceptron model, artificial neuron and activation functions, Feedforward neural networks, Loss functions and cost optimization, Gradient descent and variants, Backpropagation algorithm, Overfitting and under fitting.

Unit 3: Training Deep Neural Networks

(5 Hours)

Weight initialization techniques, Optimization algorithms: SGD, Momentum, RMSProp, Adam, Regularization methods: L1, L2, Dropout, Batch normalization, Hyperparameter tuning, Model evaluation metrics

SECTION II

Unit 4: Convolutional Neural Networks (CNNs) (5 Hours)

Convolution operation and filters, CNN architecture, Pooling techniques, Popular CNN architectures (LeNet, AlexNet, VGG – overview), Image classification and feature extraction, CNN applications in vision and electronics systems

Unit 5: Recurrent Neural Networks (RNNs) and Sequence Models (5 Hours)

Sequence modeling and temporal data, Recurrent Neural Networks, Vanishing and exploding gradients, Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), Applications: time-series analysis, speech, signals

Unit 6: Advanced Topics and Applications (5 Hours)

Autoencoders and representation learning, Introduction to Transformers, Transfer learning and fine-tuning, Deep learning for IoT and embedded systems, Ethics, bias, and limitations of deep learning, Case studies in healthcare, automation, and smart systems.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

Suggested List of Practical:

- 1) Installation and setup of deep learning environment
 - 2) Python basics for deep learning
 - 3) Implementation of single-layer perceptron
 - 4) Implementation of feedforward neural network
 - 5) Backpropagation using a deep learning framework
 - 6) Training and evaluation of deep neural networks
 - 7) Implementation of CNN for image classification
 - 8) Feature visualization using CNNs
 - 9) Implementation of RNN for sequence prediction
 - 10) Time-series forecasting using LSTM
 - 11) Hyperparameter tuning experiment
 - 12) Transfer learning using pre-trained models
-

❖ Recommended Text Books:

1. **Ian Goodfellow, Yoshua Bengio, Aaron Courville**, Deep Learning, MIT Press
 2. **François Chollet**, Deep Learning with Python, Manning Publications
-

❖ Recommended Reference Books:

1. **Nikhil Buduma**, Fundamentals of Deep Learning, O'Reilly Media
 2. **Michael Nielsen**, Neural Networks and Deep Learning
 3. **Bishop, C. M.**, Pattern Recognition and Machine Learning
 4. Online documentation and tutorials of TensorFlow and PyTorch
-

❖ Software Tools:

1. Python
2. TensorFlow / Keras
3. PyTorch
4. Jupyter Notebook



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

Semester – V

Honors in AI TECHNOLOGY

ECEHON-03A- Soft Computing

Teaching Scheme

Lectures: - 3 Hrs/Week, 3 credits

Practical: - 2 Hrs/Week, 1 credit

Examination Scheme

ESE: 70 Marks

ISE: 30 Marks

ICA: 25 Marks

Introduction:

This course introduces the fundamental concepts and techniques of Soft Computing and Intelligent Systems, including neural networks, fuzzy logic, evolutionary algorithms, and machine learning models. The course also highlights real-world applications of soft computing in data science and intelligent decision-making systems.

Prerequisite: Students should have basic knowledge of

- Engineering Mathematics (Linear Algebra, Probability and Statistics)
- Fundamentals of Artificial Intelligence
- Data Structures and Algorithms
- Programming using C/C++ or Python

Course Objectives (COs):

The objectives of this course are to:

1. Introduce the concepts, scope, and importance of soft computing techniques.
2. Provide in-depth understanding of artificial neural networks and their learning mechanisms.
3. Explain supervised, unsupervised, and associative learning networks.
4. Develop knowledge of fuzzy logic theory, fuzzy inference, and control systems.
5. Explain evolutionary computation techniques such as genetic algorithms and differential evolution.
6. Familiarize students with hybrid soft computing techniques and real-world engineering applications.

Course Outcomes (COs):

At the end of the course, the student will be able to:

1. Explain the fundamentals, scope, and components of soft computing.
2. Analyze artificial neural network models and learning algorithms.
3. Design supervised, unsupervised, and associative neural network models.
4. Apply fuzzy logic concepts for modeling, decision making, and control systems.
5. Implement genetic algorithms and differential evolution techniques for optimization problems.
6. Analyze hybrid soft computing techniques and apply them to real-world engineering problems.

SECTION I

Unit 1: Introduction to Soft Computing and Intelligent Systems

(07)

Introduction to soft computing: concepts, characteristics, and limitations of hard computing, Relationship between soft computing, machine learning, deep learning, and data science, Fundamental concepts of artificial neural networks, Basic models of artificial neural networks, Important terminologies of ANNs, McCulloch–Pitts neuron model, Linear separability and Hebbian learning

Unit 2: Supervised Learning and Regression Models

(07)

Machine learning paradigm: supervised vs unsupervised learning, linear regression and multiple linear regression, Logistic regression and classification problems, Gradient descent optimization, Perceptron learning algorithm, Adaptive Linear Neuron (ADALINE), Introduction to bias–variance trade-off

Unit 3: Neural Networks and Deep Learning Models (07)

Multi-layer perceptron (MLP), Back propagation algorithm, Activation functions and loss functions, Deep neural networks (DNNs), Convolutional Neural Networks (CNNs): architecture and applications, Introduction to recurrent neural networks (RNNs)

SECTION II

Unit 4: Unsupervised Learning and Clustering Techniques (07)

Unsupervised learning concepts, K-means clustering, Hierarchical clustering, Self-Organizing Maps (SOM), Dimensionality reduction: PCA overview, Applications of clustering in data science

Unit 5: Fuzzy Logic and Decision-Making Systems (07)

Classical sets and fuzzy sets, Membership functions and fuzzification, Fuzzy relations and operations, Fuzzy rule base and inference systems, Defuzzification techniques, Fuzzy decision-making and fuzzy logic controllers, Applications of fuzzy systems in intelligent decision support

Unit 6: Evolutionary Algorithms, Hybrid Models and Data Science Applications (07)

Genetic algorithms: operators and optimization process, Differential evolution, Hybrid soft computing models: neuro-fuzzy and genetic-neural systems, Introduction to ensemble learning, Stability considerations in intelligent systems, Applications of soft computing in machine learning and data science

Internal Continuous Assessment (ICA):

ICA consists of minimum ten Practicals based upon above curriculum.

List of Experiments:

1. Introduction to the MATLAB / Python environment for soft computing.
 2. Implementation of McCulloch–Pitts neuron model.
 3. Perceptron learning algorithm implementation.
 4. ADALINE and MADALINE network implementation.
 5. Back-propagation neural network implementation.
 6. Radial basis function network implementation.
 7. Self-organizing feature map (SOM) implementation.
 8. Hopfield network for pattern storage and recall.
 9. Fuzzy set operations and membership function design.
 10. Fuzzy inference system implementation.
 11. Fuzzy logic controller design.
 12. Genetic algorithm for function optimization.
 13. Differential evolution algorithm implementation.
 14. Neuro-fuzzy hybrid system implementation.
 15. Mini project based on real-world soft computing application.
-

Textbooks:

1. S. N. Sivanandam and S. N. Deepa, *Principles of Soft Computing*, Wiley India.
 2. J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Pearson.
 3. Simon Haykin, *Neural Networks and Learning Machines*, Pearson.
-

Reference Books:

1. Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, Wiley.
 2. David E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Pearson.
 3. Marco Dorigo and Thomas Stützle, *Ant Colony Optimization*, MIT Press.
 4. MathWorks Documentation on Neural Network, Fuzzy Logic, and GA Toolboxes.
-



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

T. Y. B. Tech. (Electronics & Computer Engineering)

SEMESTER-V

Honors in Internet of Things (IOT) and Smart Systems

ECEHON-03B– Cloud computing for IoT applications

Teaching Scheme:

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

Cloud computing plays a vital role in enabling Internet of Things (IoT) systems by providing scalable computing power, storage, and data analytics capabilities. IoT devices generate large volumes of data that require efficient cloud-based processing, storage, and management. This course introduces the concepts of cloud computing, virtualization, cloud service models, and integration of IoT with cloud platforms. It also covers cloud security, data management, and real-world IoT cloud applications.

Prerequisites:

Students should have basic knowledge of: Programming concepts, Computer Networks, Operating Systems and Database fundamentals.

Course Objectives:

The course aims to:

1. Understand the fundamental concepts and architecture of cloud computing.
2. Explore cloud service models and deployment models used for IoT applications.
3. Study virtualization technologies and resource provisioning in cloud environments.
4. Analyze the integration of IoT devices with cloud platforms and services.
5. Evaluate security, privacy, and data management challenges in cloud-enabled IoT systems.
6. Apply cloud platforms to develop scalable IoT applications.

Course Outcomes:

At the end of the course students will be able to:

1. Explain cloud computing concepts, architectures, and service models.
2. Analyze virtualization and resource provisioning mechanisms in cloud environments.
3. Evaluate different cloud platforms for IoT data processing and storage.
4. Integrate IoT devices with cloud services for real-time monitoring applications.
5. Identify security, privacy, and compliance issues in cloud-based IoT systems.
6. Design simple cloud-enabled IoT solutions using modern cloud platforms.

SECTION I

Unit 1: Fundamentals of Cloud Computing

(07)

Overview of cloud computing, evolution of cloud technologies, characteristics and benefits of cloud computing, cloud architecture, layers of cloud computing, service models (IaaS, PaaS, SaaS), deployment models (public, private, hybrid, community cloud), advantages and challenges of cloud computing.

Unit 2: Virtualization and Cloud Infrastructure**(07)**

Concept of virtualization, types of virtualization, hypervisors, virtual machines, containerization concepts, VM provisioning, VM migration techniques, resource allocation and management, virtualization tools and platforms.

Unit 3: Cloud Platforms for IoT**(07)**

Introduction to IoT cloud platforms, integration of IoT with cloud computing, cloud storage and computing for IoT data, overview of AWS IoT, Microsoft Azure IoT, Google Cloud IoT services, IoT gateways and edge computing.

SECTION II**Unit 4: Cloud Security and Privacy****(07)**

Security issues in cloud computing, cloud security architecture, identity and access management (IAM), encryption and authentication mechanisms, data protection, cloud compliance and risk management, security challenges in IoT-cloud integration.

Unit 5: Cloud Applications and Business Perspectives**(07)**

Cloud computing in business, cost models and ROI analysis, service level agreements (SLA), cloud adoption strategies, cloud-based application deployment, case studies of cloud-enabled IoT applications in smart cities, healthcare, and agriculture.

Unit 6: Legal and Ethical Issues in Cloud Computing**(07)**

Data privacy issues, regulatory compliance, jurisdiction and data location concerns, cloud service contracts, legal considerations for cloud adoption, ethical issues in cloud data management.

Internal Continuous Assessment (ICA):

ICA shall consist of minimum 8–10 practical assignments based on the following topics:

1. Introduction to cloud platforms and services
2. Creating and managing virtual machines using VirtualBox/VMware
3. Setting up an account on AWS / Azure / Google Cloud
4. Deploying and managing cloud storage services
5. Hosting a simple web application on cloud infrastructure
6. Introduction to Docker and container-based deployment
7. Deploying IoT data to cloud using ThingSpeak or AWS IoT Core
8. Creating serverless functions using AWS Lambda or Azure Functions
9. Implementing cloud-based database services
10. Simple IoT data visualization dashboard using cloud platform

Text Book:

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi – Mastering Cloud Computing, McGraw Hill.
2. Thomas Erl, Ricardo Puttini, Zaigham Mahmood – Cloud Computing: Concepts, Technology & Architecture, Pearson..

Reference Book:

1. Arshdeep Bahga, Vijay Madiseti – Internet of Things: A Hands-on Approach, Universities Press.
2. Michael Miller – Cloud Computing: Web-Based Applications that Change the Way You Work, Que Publishing.
3. Anthony T. Velte, Toby J. Velte – Cloud Computing: A Practical Approach, McGraw Hill.
4. Eric Bauer, Randee Adams – Reliability and Availability of Cloud Computing, Wiley.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

T. Y. B. Tech. (Electronics & Computer Engineering)

SEMESTER-V

Honors in Railway Engineering

ECEHON-03C–Applications of IT and Control Engineering in Railway

Teaching Scheme:

Lecture: 3 hrs/week, 3 credits

Practical : 2 hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

This course provides an overview of the Digital Railway Ecosystem, focusing on smart freight operations, next-generation passenger services, automation, embedded safety systems, and mission-critical telecommunications.

Prerequisite:

Students should have basic knowledge of Electrical and Electronics Engineering, Control Systems, and Microprocessors/Microcontrollers. Familiarity with Computer Networks, Communication Systems, Sensors & Instrumentation, and basic programming concepts will help in understanding railway IT infrastructure, automation, and intelligent control applications.

Course Objectives:

The course aims to:

1. To analyze the evolution of Freight (FOIS) and Passenger (PRS/UTS) systems into cloud native architectures.
2. To master Control Engineering principles applied to high-speed rolling stock and signaling.
3. To explore the integration of AI, Edge Computing, and IoT in modern Indian Railway assets.
4. To understand the implementation of Indigenous safety systems (KAVACH) and 5G-R communication.
5. To compare Indian Railway standards with Global High-Speed Rail (HSR) protocols.

Course Outcomes:

At the end of this course, students will be able to:

1. Design and troubleshoot IoT-enabled freight and logistics tracking modules.
2. Evaluate the security and scalability of digital ticketing and passenger service platforms.
3. Apply control theory to stabilize and automate railway subsystems.
4. Implement microprocessor-based safety and protection logic (Kavach/ETCS).
5. Architect hybrid communication networks using 5G-R and Satellite integration.

SECTION I

Unit 1: Digital Freight Ecosystem & Smart Logistics

(07)

FOIS 2.0: Mission Statement, shift from Legacy to Cloud Architecture, **Smart Freight:** Integration of RFID, GPS, and IoT for real-time wagon tracking, **E-Way Bill Integration:** Interoperability with GSTN and National Logistics Portal, **Asset Management:** Introduction to FMM (Freight Maintenance Management) and digital prototyping of rolling stock.

Unit 2: Next-Gen Passenger Services & UTS (07)

Modern PRS: Distributed Micro-services architecture, High-availability (HA) server clusters, **Digital Ticketing:** Evolution of UTS-on-Mobile, QR-based ticketing, and NCMC (National Common Mobility Card) integration, **AI in Passenger Flow:** Using Big Data for waitlist prediction, **Security:** Cybersecurity protocols for preventing ticket botting and data breaches.

Unit 3: Control Systems & Automation (07)

Fundamentals: Feedback loops in Traction Control and Regenerative Braking, **Railway Specific Control:** Anti-skid/Wheel Slide Protection (WSP) systems, Liquid level control for bio-toilets, and HVAC automation, **Advanced Monitoring:** Data Loggers, Event Recorders (Black Box), and Cloud-based SCADA for Traction Power Supply, **Robotics:** Automated Under-gear Surveillance Robots (R-NAV) and AI-based track inspection drones.

SECTION II

Unit 4: Embedded Systems & Safety Integrity (07)

Micro-controllers in Rolling Stock: Architecture of TCMS (Train Control and Management System) in Vande Bharat trains, **KAVACH (ATP):** Logic and hardware for the indigenously developed Train Collision Avoidance System, **On-board Electronics:** Integration of LiDAR, DSLR-based OHE monitoring, and MEMS sensors for vibration analysis, **Edge Computing:** Processing sensor data at the "Edge" for hot-axle detection and derailment prevention.

Unit 5: Mission-Critical Railway Telecommunications (07)

Evolution of Control: Transition from 4-wire analog circuits to VoIP-based Train Control, **OFC Networks:** SDH/DWDM backbone in railways, Dark Fiber utilization, **Test Protocols:** Routine diagnostics of control circuits, MTTR (Mean Time To Repair) optimization in telecom faults, **Emergency Communications:** Satellite-based emergency phones and Tunnel Communication systems.

Unit 6: Future Communication & Passenger Information Systems (07)

Wireless Standards: Transition from GSM-R to 5G-R (Future Railway Mobile Communication System - FRMCS), **PIS/PAS:** IP-based Public Address Systems, Real-time GPS-synced Passenger Information Displays, **Network Convergence:** TETRA and DECT for yard communications; Wi-Fi 6 implementation in stations and trains, **Metro Specific:** CBTC (Communication Based Train Control) and Driverless Train Operation (DTO) protocols.

Internal Continuous Assessment (ICA):

ICA shall consist of a minimum of eight assignments/lab exercises, including:

- 1. Simulation:** Modeling a feedback control loop for a railway braking system using MATLAB/Simulink.
- 2. Case Study:** Analysis of the KAVACH safety logic and signal interlocking.
- 3. Project:** Designing a prototype IoT sensor node for track temperature monitoring.
- 4. Field Visit/Virtual Tour:** Study of a Divisional Control Office or a Modern Signaling Relay Room.
- 5. Technical Report:** Comparative study of Indian Railway standards vs. UIC (International Union of Railways) standards for 2026.

References:

RDSO. Handbook on LTE Network & its applications in Indian Railways.

RDSO. (2023, February). Technical Advisory Note (TAN) on Implementation of LTE on Indian Railways.

South Central Railway. (2023, August 9). IRISSET & IIT-Madras Partner for India 5G Testbed for Railways.

CRIS/ESCAP. (2023, April 6). Digitalization of INDIAN RAILWAYS' freight BUSINESS. Indian Railways. (2022, January 20). FOIS-INNOVATIONS.

NFR & IIT Guwahati. (2025, November 10). Collaboration on AI-Based Surveillance System DRISHTI for wagon security.

RDSO. (2025, October 21). Expression of Interest (EOI) for proto-development of Machine Vision-based Inspection System (MVIS).

RDSO. (2023, February 6). Technical Specification for Machine Vision based Inspection System for Rolling Stocks.



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
T. Y. B. Tech. (Electronics & Computer Engineering)

SEMESTER-V
Honors in Renewable Energy and Power Electronics
ECEHON-03D- Grid integration and Smart grids

Teaching Scheme

Lectures: - 3 Hrs/Week, 3 credits
Practical: - 2 Hrs/Week, 1 credit

Examination Scheme

ESE: 70 Marks
ISE: 30 Marks
ICA: 25 Marks

Introduction:

Grid Integration and Smart Grids deal with modernizing conventional power systems using advanced communication, control, and power electronic technologies to enable reliable, efficient, and sustainable operation with renewable energy integration.

Prerequisite:

1. Power System Fundamentals 2. Electrical Machines 3. Power Electronics etc.

Course Prerequisite: Students should have basic knowledge of

- Electrical Power Systems (generation, transmission and distribution)
- Fundamentals of Renewable Energy Sources (solar, wind, etc.)
- Power Electronics devices and converters
- Electrical Machines and Control basics
- Basic concepts of Communication Systems and Control Systems

Course Objectives (COs):

The objectives of this course are to:

1. Introduce the concept, need and evolution of smart grids from conventional power systems.
2. Provide understanding of smart grid architecture, components and operational principles.
3. Explain the challenges and solutions for grid integration of renewable energy sources.
4. Develop knowledge of power electronics interfaces used in renewable energy integration.
5. Familiarize students with smart grid communication, monitoring, control and protection systems.
6. Expose students to emerging technologies and applications in smart grid systems.

Course Outcomes (COs):

By the end of the course, the student will be able to:

1. Explain the structure, operation and limitations of conventional power grids and the need for smart grids.
2. Identify and describe key components and architecture of smart grid systems.
3. Analyze the challenges associated with grid integration of renewable energy sources.
4. Apply power electronics concepts for grid-connected renewable energy systems.
5. Describe communication, monitoring, control and cyber security aspects of smart grids.
6. Evaluate smart grid applications, energy storage systems and future trends in grid modernization.

SECTION I

Unit-1: Introduction to Power Grid & Smart Grid

(07)

Conventional Power Grid – overview of generation, transmission, distribution, Limitations of conventional grids, Evolution from traditional grids to Smart Grids, Definition, characteristics,

advantages of smart grids over conventional grids, Stakeholders and drivers for Smart Grid adoption, Examples of Smart Grid implementation scenarios

Unit-2: Smart Grid Architecture and Components (07)

Smart grid architecture: generation–transmission–distribution–consumption, Key components: Smart Meters, AMI, PMU, IEDs, sensors, smart appliances, Outage management systems, real-time pricing, demand response, Electric vehicles (EV), Vehicle-to-Grid (V2G) concepts, Smart substations, feeder automation and data concentrators

Unit-3: Renewable Energy Integration (07)

Renewable sources: Solar PV, Wind, Biomass, small scale DGs, Mini-grid, microgrid and distributed generation (DG) systems, Grid-interconnection standards and requirements, Synchronization of renewables with grid, Power quality issues from renewables, Impact of variability and intermittency on grid stability

SECTION II

Unit-4: Power Electronics in Grid Integration (07)

Power electronic converters – basic principles and classifications, Inverters: grid-tied vs grid-forming converters, MPPT techniques for PV systems, Converter control for wind turbines (DFIG/PMSG), Bidirectional converters for storage and EV interfaces, Power electronics for reactive power compensation and power quality improvement

Unit-5: Smart Grid Communication & Control Systems (07)

Communication infrastructures: HAN, NAN, WAN, Wired and wireless technologies (Zigbee, Wi-Fi, BPL etc.), SCADA, Energy Management Systems (EMS) and distributed control, Cyber security basics and protection of grid data, Integration of IoT and real-time monitoring system

Unit-6: Applications, Monitoring & Emerging Technologies (07)

Wide Area Measurement Systems (WAMS) & Phasor Data Concentrators (PDC), Demand Side Management (DSM) and Advanced Energy Management, Energy storage technologies (batteries, pumped hydro, SMES), Case studies of smart grids and hybrid renewable systems, Future trends: AI, machine learning, blockchain in grid operations

Internal Continuous Assessment (ICA):

ICA consists of minimum eight experiments based upon above curriculum.

List of Experiments:

1. Study of conventional power grid structure and identification of limitations using block diagrams.
 2. Study of smart grid architecture and its key components (AMI, Smart Meter, PMU, IEDs).
 3. Simulation of a grid-connected solar PV system using MATLAB/Simulink.
 4. Study and analysis of grid-connected inverter characteristics and synchronization methods.
 5. Implementation of Maximum Power Point Tracking (MPPT) algorithms for solar PV systems.
 6. Simulation of wind energy conversion system integrated with the grid.
 7. Analysis of power quality issues (harmonics, voltage sag/swell) due to renewable energy integration.
 8. Study of micro grid and distributed generation (DG) system configurations.
 9. Study of communication technologies used in smart grids (HAN, NAN, WAN).
 10. Case study and analysis of a real-world smart grid or renewable energy grid-integration project.
 11. Mini-project: Design and simulation of a smart grid model integrating renewable energy sources, storage and load management.
-

Textbooks

1. Smart Grid – Fundamentals of Design and Analysis – James Momoh
 2. Design of Smart Power Grid Renewable Energy Systems – Ali Keyhani
-

Reference Books

1. Smart Grids: Technology and Applications – Janaka Ekanayake et al.
 2. Power Electronics for Renewable Energy Systems – Haitham Abu-Rub et al.
-



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPCC-10 – Real Time Operating System

Teaching Scheme

Lectures – 2 Hrs/week, 2 Credits

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

Introduction:

This course provides an overview of Real-Time Operating Systems (RTOS) and their role in embedded and industrial systems, covers task scheduling, synchronization, inter-task communication, and RTOS services with real-time constraints, and enables students to gain practical skills to design, analyze, and optimize RTOS-based embedded applications.

Prerequisite: Student shall have knowledge of basic programming.

Course Objectives:

The course aims to:

1. Understand the basic concepts of operating systems and the need for real-time systems.
2. Explain task scheduling, synchronization, and communication in real-time systems.
3. Analyze the design and performance of various real-time operating systems.
4. Apply RTOS concepts to develop embedded applications for microcontrollers.
5. Evaluate task priorities, resource management, and timing constraints in real-time applications.
6. Propose optimized solutions for real-time systems in industrial and embedded applications.

Course Outcomes:

At the end of the course students will be able to

1. Recall fundamental concepts of operating systems and RTOS. (Remember)
2. Interpret scheduling policies, task synchronization, and inter-task communication. (Understand)
3. Apply RTOS features to design multitasking embedded applications. (Apply)
4. Analyze system performance, task response times, and resource utilization. (Analyze)
5. Evaluate different RTOS architectures for specific embedded applications. (Evaluate)

SECTION-I

Unit 1: Introduction to RTOS

(05)

Definition and characteristics of real-time systems, Hard vs. soft real-time systems, Overview of embedded systems and microcontrollers, RTOS vs. General Purpose OS, Applications of RTOS in industry and electronics.

Unit 2: Task Management and Scheduling

(05)

Task states and types (periodic, aperiodic, sporadic), Task control block (TCB) and context switching
Scheduling policies: Rate Monotonic, Earliest Deadline First, Priority-based, Preemptive vs. non-Preemptive scheduling, Multitasking and task management in RTOS

Unit 3: Inter-task Communication and Synchronization

(05)

Semaphores, mutexes, and events, Message queues and mailboxes, Shared memory and data

Consistency, Priority inversion and priority inheritance, Deadlocks and avoidance techniques

SECTION-II

Unit 4: RTOS Services and APIs

(05)

RTOS kernel services, Timers, clocks, and delays, Memory management in RTOS, RTOS API functions Overview, Example RTOS platforms: FreeRTOS, VxWorks, μ C/OS

Unit 5: RTOS Programming and Real time examples

(05)

RTOS-based embedded system design, Task creation, deletion, and scheduling using Free RTOS, Interrupt handling and ISR design, Example applications: motor control, sensor interfacing, IoT devices, Debugging and performance analysis

Unit 6: Advanced Topics and Future Trends

(05)

Real-time Linux and RT-preempt patch, Safety-critical and industrial RTOS, Distributed RTOS and Networked embedded systems, Security in RTOS-based systems, Future trends in IoT and real-time Applications

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

- 1) RTOS installation and configuration (FreeRTOS / μ C/OS)
- 2) Creating and running multiple tasks
- 3) Task scheduling and priority assignment demonstration
- 4) Semaphore and mutex usage for synchronization
- 5) Message queue implementation between tasks
- 6) Timer and delay implementation in RTOS
- 7) Interrupt Service Routine (ISR) handling
- 8) Context switching analysis and measurement
- 9) Task deletion and memory management demonstration
- 10) Motor control using RTOS-based multitasking
- 11) Sensor interfacing using RTOS
- 12) IoT device simulation using RTOS

(Tools: FreeRTOS, Keil μ Vision IDE, STM32 / Arduino / Raspberry Pi platforms, simulators for embedded RTOS)

Text Books:

1. Jane W. S. Liu, Real-Time Systems, Pearson, 2000.
2. Raj Kamal, Embedded Systems: Architecture, Programming and Design, 3rd Edition, McGraw-Hill.

Reference Books:

1. Andrew S. Tanenbaum, Modern Operating Systems, 4th Edition, Pearson.
2. RTOS official manuals: FreeRTOS, μ C/OS-II/III documentation.
3. Tammy Noergaard, Embedded Systems Architecture, 2nd Edition, Elsevier.
4. K. J. Ayala, The 8051 Microcontroller Architecture, Programming and Applications, 3rd Edition, Penram International.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPCC-11 – Software testing & Quality Assurance

Teaching Scheme

Lectures – 2 Hrs/week, 2 Credits

Practical – 2 Hrs/week, 1 Credit

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

ICA - 25 Marks

POE - 25 Marks

Introduction:

Software testing is a crucial phase of the software development life cycle (SDLC) that ensures the reliability, functionality, and performance of software applications. This course provides a foundational understanding of software testing principles, methodologies, and best practices. It covers various testing approaches, including black-box and white-box testing, verification and validation, and different levels of testing such as unit, integration, system, and acceptance testing.

Prerequisite:

Basic Software Engineering Concepts – Concepts like requirements analysis, design, coding, and maintenance Programming Fundamentals – Knowledge of at least one programming language (e.g., Java, Python) to understand test automation and debugging.

Course Objectives:

The course aims to:

1. To provide students with a comprehensive understanding of software testing principles, methodologies, and practices. To draw implications from basic primitives.
2. To introduce software quality assurance, industry-standard testing tools, and automation techniques.
3. To introduce the fundamental testing concepts, verification and validation techniques, levels of testing, and test planning.

Course Outcomes:

At the end of the course students will be able to

1. Recall fundamental concepts, techniques, and methodologies of software testing.
2. Discuss testing levels and review system testing types to assess software performance.
3. Design effective test plans, test cases, and bug-tracking report using software quality assurance principles and standards.
4. Demonstrate automated testing tools to test software.

SECTION-I

Unit 1: Fundamentals of Software Testing

(05)

Introduction, Basics of Software Testing, Approaches to Testing, Testing During Development Life Cycle, Essential of Software Testing, Features of Testing, Misconceptions About Testing, Principles of Software Testing, Test Policy, Strategy, Planning, Process, Challenges in Testing, Test Team Approach, Methods.

Unit 2: Methods of Testing**(05)**

Software Verification and Validation, Black-Box and White-Box Testing, Static and Dynamic Testing, Black-Box Testing Techniques-Equivalence Partitioning, Data Testing, State Testing, Other Black Box Test Techniques. White-Box Testing Techniques.

Unit 3: Levels of Testing**(05)**

Verification and Validation Model, Levels of Testing, Proposal Testing, Requirement Testing, Design Testing, Code Review, Unit Testing, Module Testing, Integration Testing, Big-Bang Testing, Sandwich Testing, System Testing- GUI Testing, Compatibility Testing.

SECTION-II**Unit 4: Test Planning & Documentation****(06)**

Test Planning-The goal of Test Planning, Test Planning Topics, Writing and Tracking Test Cases. The Goal of Test Case Planning, Test Case Planning Overview, Test Case Organization and Tracking, Reporting Bugs- Getting Your Bugs Fixed

Unit 5: Quality Concepts & Software Quality Assurance**(05)**

Quality Concepts-What is Quality?, Software Quality, The Software Quality Dilemma, Achieving Software Quality, Software Quality Assurance-Background Issues, Elements of Software Quality Assurance.

Unit 6: Automated Testing and Testing Tools**(04)**

Introduction, The Benefits of Automation and Tools, Test Tools, Software Test Automation, Random Testing, Realities of Using Test Tools and Automation, Open-Source Testing Tools, Case Studies on Testing Tools-Selenium.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

1. Introduction to Software Testing
2. Develop a comprehensive test plan for a simple application (e.g., login system or calculator).
3. Write Test Cases for a Login Page
4. Apply black-box techniques (boundary value analysis, equivalence partitioning) to a sample problem.
5. Apply White-Box Testing techniques (Control Flow Graph, Condition Coverage, Path Coverage).
6. Use a Bug Tracking System (e.g., Bugzilla, JIRA, or open-source tool) to log and manage bugs for a given project.
7. Prepare a Test Plan and Test Case Documentation for a sample software project (include Test Objectives, Strategy, and Environment).
8. Perform GUI Testing and Usability Testing on a web application.
9. Demonstrate Automated Testing using Selenium – write and execute test scripts for login functionality of a sample website.
10. Use a tool like JMeter or LoadRunner to test the performance of a web application.
11. Write unit tests for given code snippets in Java, Python, or C# using JUnit, PyTest, or NUnit.
12. End-to-End Test Strategy for a Web application.

Text Books:

1. Software Testing Principles, Techniques and Tools By M G Limaye, Published by Tata McGraw-Hill Education Private Limited, Published 2009, ISBN (13): 978-0-07-013990-9, ISBN (10): 0-07-013990-3 (Chapter 1 & 3)
2. Software Testing, Second Edition By: Ron Patton, Published by SAMS, ISBN-13: 978- 0672327988 ISBN-10: 0672327988 (Chapter 2, 4 & 6)
3. Software Engineering: A Practitioner's Approach by Roger S Pressman, 8th Edition Publisher McGraw Hill (Chapter 5)

Reference Books:

1. Software Testing Principle and Practices by Ramesh Desikan, Gopaldaswamy Ramesh, Pearson Education, ISBN 978-81-7758-121-8
2. Software Testing Principles and Practices by Naresh Chauhan, Publisher OXFORD UNIVERSITY PRESS-NEW DELHI, ISBN 0-19-806184-6
3. Beautiful Testing: Leading Professionals Reveal How They Improve Software By Adam Goucher, Tim Riley, Publisher O'reilly
4. Foundations of Software Testing By Rex Black, Dorothy Graham, Erik Van Veenendaal IsabelEvans, Published by Cengage Learning India Pvt Ltd.
5. Lessons Learned in Software Testing by Cem Kaner, James Bach, Bret Pettichord, Publisher Wiley
6. Testing Computer Software Cem Kaner, Jack Falk, Hung Q. Nguyen, Publisher Wiley
7. Selenium Testing Tools Cookbook By Unmesh Gundecha Published by Packt, ISBN: 978- 1-84951-574-0
8. Dr. K.V.K.K. Prasad, "Software Testing Tools: Covering WinRunner, Silk Test, LoadRunner, JMeter and TestDirector with Case Studies", Dreamtech Publications ISBN:10:81-7722-532-4

Reference tutorials:

1. Spoken Tutorials on Selenium Software Testing Framework at http://spokentutorial.org/tutorialsearch/?search_foss=Selenium&search_language=English



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI
ECEPCC-12 – System Design Using Arduino & Raspberry Pi

Teaching Scheme

Lectures – 3 Hrs/week, 3 Credits

Practical – 2 Hrs/week, 1 Credit

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

ICA – 25 Marks

Introduction:

This course introduces system design using Arduino and Raspberry Pi with a focus on embedded systems and hardware–software integration, covers programming, GPIO interfacing, sensors, actuators, and communication protocols for real-world applications, and prepares students to design industrial and IoT-based electronic systems through hands-on experiments and mini projects.

Prerequisite: Students should have basic knowledge of Basic Electronics, Digital Electronics, Microcontrollers / Embedded Systems, Fundamentals of C / Python Programming

Course Objectives:

The course aims to:

1. To understand the architecture of Arduino and Raspberry Pi
2. To design electronic systems using microcontroller and single-board computer platforms
3. To interface sensors, actuators, and communication modules
4. To develop embedded applications using C/C++ and Python
5. To design industrial and IoT-based electronic systems
6. To implement complete electronic systems with hardware–software integration

Course Outcomes:

At the end of the course students will be able to

1. Explain the architecture and features of Arduino and Raspberry Pi.
2. Interface sensors and actuators with Arduino and Raspberry Pi.
3. Implement serial communication protocols for system design.
4. Develop embedded programs using Arduino IDE and Python
5. Design industrial and IoT-based electronic systems.
6. Integrate hardware and software to realize complete electronic systems

SECTION-I

Unit 1: Introduction to Embedded Systems

(07)

Definition and characteristics of embedded systems, Embedded system components: hardware and software, Embedded vs general-purpose computing systems, Classification of embedded systems, Applications of embedded systems, Overview of embedded system design flow

Unit 2: Programming and GPIO Interfacing

(07)

Arduino IDE and program structure, Embedded C/C++ programming for Arduino, Digital and analog I/O programming, GPIO programming on Raspberry Pi using Python, PWM generation and control, Interfacing LEDs, push buttons, relays, and buzzers

Unit 3: Sensor and Actuator Interfacing (07)

Interfacing sensors with Arduino and Raspberry Pi: Temperature sensors (LM35, DS18B20) Humidity sensors, Light sensors (LDR), Proximity sensors Interfacing actuators: DC motors, Stepper motors, Servo motors Motor driver circuits and power considerations

SECTION-II

Unit 4: Communication Interfaces (07)

Serial communication fundamentals: UART communication, I2C communication, SPI communication, Interfacing external ADC/DAC, Interfacing displays (LCD, OLED)

Unit 5: Industrial & IoT System Design (07)

Data acquisition systems using Arduino and Raspberry Pi, Signal conditioning and data logging Introduction to IoT architecture, Web-based monitoring using Raspberry Pi, Cloud connectivity and data visualization, Security considerations in IoT systems

Unit 6: System Integration and Applications (07)

Electronic system design methodology, Hardware–software co-design approach, Case studies: Smart home automation, Industrial monitoring system, Energy management system Smart agriculture system Debugging, testing, and documentation, Mini project design and implementation

Internal Continuous Assessment (ICA):

ICA consists of a minimum 8 experiments based upon the above curriculum.

- 1. Introduction to Arduino IDE and Raspberry Pi OS**
Installation, board configuration, Linux commands, and tool chain setup.
- 2. Digital I/O Programming using Arduino**
LED blinking, push button interfacing, and logic control.
- 3. Analog Input Interfacing using Arduino**
Interfacing potentiometer and analog sensors using ADC.
- 4. PWM Signal Generation and Control**
LED brightness control and speed control of DC motor using PWM.
- 5. Temperature Sensor Interfacing**
Interfacing LM35 / DS18B20 with Arduino and Raspberry Pi.
- 6. Light and Proximity Sensor Interfacing**
LDR and IR / ultrasonic sensor interfacing with threshold-based control.
- 7. Relay and Buzzer Interfacing**
Designing simple control systems using relays and alarm indicators.
- 8. DC Motor Control using Motor Driver IC**
Direction and speed control using L293D / L298 with Arduino.
- 9. Servo and Stepper Motor Interfacing**
Position and speed control of servo and stepper motors.
- 10. LCD / OLED Display Interfacing**
Displaying sensor data on 16×2 LCD or OLED using Arduino / Raspberry Pi.

11. **UART Communication Interface**
Serial communication between Arduino and Raspberry Pi.
 12. **I2C Communication Interface**
Interfacing I2C-based sensors and peripherals.
 13. **SPI Communication Interface**
SPI-based device interfacing and data transfer.
 14. **Data Logging System using Raspberry Pi**
Sensor data storage using files or database with timestamp.
 15. **Mini Project based on System Design**
Design, implementation, testing, and documentation of a complete system
-

Text Books:

1. Simon Monk, Programming Arduino, McGraw-Hill
 2. Eben Upton & Gareth Halfacree, Raspberry Pi User Guide, Wiley
 3. Derek Molloy, Exploring Raspberry Pi, Wiley
-

Reference Books:

1. Raj Kamal, Embedded Systems, McGraw-Hill
2. Muhammad Ali Mazidi, The AVR Microcontroller and Embedded Systems, Pearson
3. Raspberry Pi and Arduino Official Documentation



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPEC-02A: Deep learning

Teaching Scheme:

Lecture: 3 Hrs/week, 3 credits

Practical : 2 Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

POE: 25 Marks

Introduction:

Deep Learning is a subfield of Machine Learning that focuses on artificial neural networks with multiple layers to model complex patterns in data. This course introduces the fundamentals of neural networks, deep architectures, training techniques, and applications in computer vision, natural language processing, speech recognition, and intelligent systems.

Prerequisite:

Linear Algebra and Probability, Machine Learning fundamentals, Python programming, Basic understanding of data handling and statistics

Course Objectives:

The course aims to:

1. Introduce fundamental concepts and architectures of deep learning.
2. Provide mathematical and computational foundations of neural networks.
3. Enable students to design and train deep learning models for real-world problems.
4. Familiarize students with deep learning frameworks and tools.
5. Apply deep learning techniques to image, signal, and data-driven applications.
6. Develop problem-solving and analytical skills using deep learning approaches

Course Outcomes:

At the end of the course students will be able to

1. Explain the fundamentals of neural networks and deep learning concepts.
2. Implement and train deep learning models using standard frameworks
3. Analyze and optimize model performance using suitable techniques.
4. Apply Convolutional Neural Networks for image and signal processing tasks.
5. Apply Recurrent Neural Networks for sequence and time-series data.
6. Solve interdisciplinary engineering problems using deep learning models.

SECTION-I

Unit 1: Introduction to Deep Learning

(07)

Artificial Intelligence, Machine Learning, and Deep Learning, Limitations of traditional machine learning, Biological inspiration of neural networks, History and evolution of deep learning, Applications in Electronics and Computer Engineering, Overview of deep learning tools and frameworks

Unit 2: Neural Network Fundamentals

(07)

Perceptron model, artificial neuron and activation functions, Feedforward neural networks, Loss functions and cost optimization, Gradient descent and variants, Back propagation algorithm, Over fitting and under fitting.

Unit 3: Training Deep Neural Network

(07)

Weight initialization techniques, Optimization algorithms: SGD, Momentum, RMSProp, Adam, Regularization methods: L1, L2, Dropout, Batch normalization, Hyper parameter tuning, Model evaluation metrics

SECTION-II

Unit 4: Convolutional Neural Networks (CNNs)

(07)

Convolution operation and filters, CNN architecture, Pooling techniques, Popular CNN architectures (LeNet, AlexNet, VGG – overview), Image classification and feature extraction, CNN applications in vision and electronics systems

Unit 5: Recurrent Neural Networks (RNNs) and Sequence Models

(07)

Sequence modeling and temporal data, Recurrent Neural Networks, Vanishing and exploding gradients, Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), Applications: time-series analysis, speech, signals

Unit 6: Advanced Topics and Applications

(07)

Autoencoders and representation learning, Introduction to Transformers, Transfer learning and fine-tuning, Deep learning for IoT and embedded systems, Ethics, bias, and limitations of deep learning, Case studies in healthcare, automation, and smart systems.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

- 1) Installation and setup of deep learning environment
- 2) Python basics for deep learning
- 3) Implementation of single-layer perceptron
- 4) Implementation of feedforward neural network
- 5) Backpropagation using a deep learning framework
- 6) Training and evaluation of deep neural networks
- 7) Implementation of CNN for image classification
- 8) Feature visualization using CNNs
- 9) Implementation of RNN for sequence prediction
- 10) Time-series forecasting using LSTM
- 11) Hyper parameter tuning experiment
- 12) Transfer learning using pre-trained model

Text Books:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press
2. François Chollet, Deep Learning with Python, Manning Publications

Reference Books:

1. Nikhil Buduma, Fundamentals of Deep Learning, O'Reilly Media
2. Michael Nielsen, Neural Networks and Deep Learning
3. Bishop, C. M., Pattern Recognition and Machine Learning
4. Online documentation and tutorials of TensorFlow and PyTorch



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPEC-02B: Data Visualization (Power BI & Tableau)

Teaching Scheme:

Lecture: 3 Hrs/week, 3 credits

Practical : 2 Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

POE: 25 Marks

Introduction:

This course focuses on data visualization concepts and the practical use of industry-leading tools such as Tableau and Microsoft Power BI. It enables students to transform raw data into meaningful dashboards, reports, and visual insights for effective decision-making. The course emphasizes data preparation, visualization design principles, dashboard development, and business analytics applications.

Prerequisite:

Fundamentals of data analysis, Basic statistics, Spreadsheet tools (e.g., Excel), Basic computer operations

Course Objectives:

The course aims to:

1. Introduce students to fundamentals of data visualization and business intelligence.
2. Enable students to understand data types, data sources, and data preparation techniques.
3. Develop skills to create interactive dashboards using Power BI and Tableau.
4. Apply visualization techniques for data-driven decision making.
5. Interpret and communicate insights effectively using visual analytics.
6. Prepare students for industry-oriented BI and analytics roles.

Course Outcomes:

At the end of the course students will be able to

1. Understand principles, importance, and applications of data visualization.
2. Prepare, clean, and transform datasets for visualization tools.
3. Create basic and advanced visualizations using Power BI and Tableau.
4. Design interactive dashboards and reports for real-world datasets.
5. Analyse data patterns and trends to support decision-making.
6. Apply visualization techniques in engineering and interdisciplinary domains.

SECTION-I

Unit 1: Introduction to Data Visualization & Business Intelligence

(07)

Introduction to data visualization, Importance of data visualization in engineering and business, Data visualization vs data analysis, Basics of Business Intelligence (BI), Visualization best practices and principles, Types of data: structured, semi-structured, unstructured, Overview of Power BI and Tableau, Applications in Electronics, Computer Engineering, and Industry

Unit 2: Data Preparation and Data Modelling

(07)

Data sources: Excel, CSV, databases, web data, Data collection and data quality issues, Data cleaning and transformation concepts, Introduction to Power Query / Tableau Prep, Data modelling concepts, Relationships,

cardinality, and normalization, Calculated columns and measures (introductory level)

Unit 3: Visualization Fundamentals and Charts (07)

Visual perception and design principles, choosing the right chart, Basic charts: bar, column, line, pie, area, Advanced charts: stacked charts, scatter plots, maps, Filters, slicers, and sorting, Formatting and customization Of visuals, Data story telling concepts.

SECTION-II

Unit 4: Power BI – Dashboard and Report Development (07)

Power BI architecture and workflow, Importing and transforming data, DAX basics: measures, functions, aggregations, creating interactive reports, Dashboard creation and layout design, Publishing and sharing reports, Power BI service overview

Unit 5: Tableau – Visualization and Analytics (07)

Tableau interface and architecture, connecting to data sources, Dimensions and measures calculated fields and parameters, Creating interactive worksheets, Dashboards and actions, Tableau storytelling feature

Unit 6: Advanced Visualization & Case Studies (07)

Comparative study: Power BI vs Tableau, KPI design and performance dashboards, Time-series and trend analysis, Visualization for IoT, electronics data, and system logs, Case studies in engineering, finance, healthcare, Introduction to real-time and big data visualization, Ethical issues and data privacy in visualization

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

1. Introduction to Power BI and Tableau interface
2. Importing datasets from Excel/CSV
3. Data cleaning and transformation using Power Query / Tableau Prep
4. Creating basic charts and visuals
5. Implementing filters and slicers
6. Creating calculated fields and measures
7. Designing interactive dashboards in Power BI
8. Dashboard creation in Tableau
9. KPI and performance dashboard design
10. Case study-based visualization (engineering dataset)
11. Comparative dashboard using Power BI and Tableau

Text Books:

1. Murray, D., Beginning Power BI, Apress (<https://content.e-bookshelf.de/media/reading/L-13393976-1ef18135ae.pdf>)
2. Ryan Sleeper, Practical Tableau, O'Reilly Media (<https://dynamicsystemindia.in/pdfs/pdfs/tableau.pdf>)

Reference Books:

1. Stephen Few, Show Me the Numbers, Analytics Press
2. Cole Nussbaumer Knaflic, Storytelling with Data, Wiley
3. Alberto Cairo, The Functional Art, New Riders
4. Microsoft Power BI and Tableau official documentation



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPEC-02C- Electronic System Design

Teaching Scheme:

Lecture: 3 Hrs/week, 3 credits

Practical : 2 Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

POE: 25 Marks

Introduction:

Electronic System Design focuses on the design, development, and implementation of complete electronic systems by integrating hardware and software components. The course covers system-level design methodology, embedded systems, PCB design, testing, and prototyping. It emphasizes practical approaches to developing reliable, efficient, and cost-effective electronic products used in industrial and consumer applications.

Prerequisite:

Electronic Devices and Circuits, Digital Electronics, Microprocessors / Microcontrollers, Basic programming concepts

Course Objectives:

The course aims to:

1. To describe the concept, construction, and characteristics of power electronic devices
2. To understand the applications of thyristors
3. To study the working and applications of chopper circuits
4. To design and analyze timers, frequency counters, and digital voltmeters
5. To design signal conditioning circuits using sensors
6. To design and analyze controllers for industrial applications

Course Outcomes:

At the end of the course students will be able to

1. Describe construction, working, and characteristics of thyristors
2. Analyze AC and DC power control circuits using thyristors
3. Explain the operation of DC chopper circuits
4. Design and implement timers, frequency counters, digital voltmeters, and frequency synthesizers
5. Design signal conditioning circuits using industrial sensors
6. Design and analyze controllers and PLC-based industrial applications

SECTION-I

Unit 1: Introduction to Modern Power Semiconductor Devices

(07)

Power IGBT: Construction and working, Transfer characteristics, Output characteristics Applications

SCR: Construction and working, V-I characteristics, Turn-On and Turn-Off methods

TRIAC: Construction, working, and V-I characteristics

DIAC: Construction, working, and V–I characteristics

Unit 2: Power Electronics Applications

(07)

Single-phase half-wave controlled rectifier, Center-tapped full-wave controlled rectifier, Fully controlled bridge rectifier

Unit 3: Chopper Circuits

(07)

DC choppers: Classification and types, Control strategies, Single quadrant chopper, Two quadrant chopper, Four quadrant chopper, Circuit diagram, Operation with waveforms
Morgan chopper: Circuit diagram, Operation with waveforms

SECTION-II

Unit 4: Timers, Counters, and Digital Voltmeter

(07)

Design of timer using XR-2240, Design of counter using IC 74C926 for time and event counting
Design of 3-digit multi-range Digital Voltmeter (DVM) using discrete components

Unit 5: Signal Conditioning Circuits

(07)

Signal conditioning for sensors: PT-100, LM-35, Thermocouples (J-type and K-type) Current loop interface: 4 mA – 20 Ma Zero and span adjustment circuits, Offset circuits Voltage-to-Current (V-I) converter, Current-to-Voltage (I-V) converter, Voltage-to-Voltage (V-V) converter

Unit 6: Design of Controllers and PLC Applications

(07)

Design of Analog Controllers: ON–OFF Controller: principle, characteristics, and applications
Proportional (P) Controller: operation, steady-state error, and applications, Process Control Applications: Role of controllers in industrial process control, Control of temperature, level, and flow processes Programmable Logic Controller (PLC) Fundamentals: Introduction to PLC, Basic architecture and components of PLC, Advantages of PLC over conventional control systems PLC-Based Industrial Applications: Bottle filling plant control system, Elevator control system

Internal Continuous Assessment (ICA):

ICA consists of minimum 8 Experiments based upon above curriculum.

List of Experiments:

1. Characteristics of Power IGBT

Plotting the Transfer and Output characteristics of an Insulated Gate Bipolar Transistor (IGBT) and calculating the threshold voltage and transconductance.

2. V-I Characteristics of SCR, DIAC, and TRIAC

Study and plot the Static V-I characteristics of the Silicon Controlled Rectifier (SCR), DIAC, and TRIAC to determine holding and latching currents.

3. SCR Turn-On and Turn-Off Methods

Demonstration of different triggering (Gate triggering) and commutation (Forced/Natural) circuits for SCR operation.

4. Single-Phase Half-Wave and Full-Wave Controlled Rectifiers

To observe and analyze the output voltage waveforms of half-wave and center-tapped full-wave controlled rectifiers with Resistive (R) and Inductive (RL) loads.

5. Single-Phase Fully Controlled Bridge Rectifier

Study of a full-bridge controlled rectifier using SCRs and observing the effect of varying the firing angle (α) on the average DC output voltage.

6. Step-Down (Buck) and Step-Up (Boost) DC Choppers

Implementation of a basic DC chopper circuit to demonstrate single-quadrant operation and control strategies like Pulse Width Modulation (PWM).

7. Multi-Quadrant Chopper Operation

Analysis of two-quadrant or four-quadrant chopper circuits to understand regenerative braking and bidirectional power flow in DC motor control.

8. Design of Timer and Event Counter

Designing a precision timer circuit using IC XR-2240 and a 4-digit event counter using IC 74C926 to display time intervals or pulse counts.

9. Signal Conditioning for Temperature Sensors

Designing and testing signal conditioning circuits for LM-35 (Voltage output) and PT-100 (Resistance change using a Bridge circuit) to provide standardized output.

10. Voltage-to-Current (V-I) and Current-to-Voltage (I-V) Converters

Implementation of an industrial standard 4mA to 20mA current loop interface for long-distance signal transmission using Op-Amps.

11. Design and Analysis of an ON-OFF and P-Controller

Building an Analog ON-OFF controller for a temperature control system and observing the "hunting" effect or steady-state error in a Proportional (P) controller.

12. PLC-Based Industrial Automation Simulation

Programming a Programmable Logic Controller (PLC) using Ladder Logic to automate a process such as a **Bottle Filling Plant** or a **Simple Elevator System**.

Text Books:

1. Rashid M. H., Power Electronics – Circuits, Devices and Applications, Pearson
2. B. W. Williams, Power Electronics, McGraw-Hill
3. D. V. Hall, Microprocessors and Interfacing, McGraw-Hill

Reference Books:

1. M. D. Singh & K. B. Khanchandani, Power Electronics, Tata McGraw-Hill
2. Ramakant A. Gayakwad, Op-Amps and Linear ICs, Pearson
3. Frank D. Petruzella, Programmable Logic Controllers, McGraw-Hill



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI
ECEPEC-02D- Digital Image Processing

Teaching Scheme:

Lecture: 3 Hrs/week, 3 credits

Practical : 2 Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

POE: 25 Marks

Introduction:

Digital Image Processing deals with the manipulation and analysis of digital images using computational techniques. This course covers image representation, enhancement, restoration, and compression, segmentation, and feature extraction. It emphasizes practical applications in medical imaging, remote sensing, computer vision, surveillance, and multimedia systems.

Prerequisite:

Students should have basic knowledge of: Engineering Mathematics (Linear Algebra, Probability, and Transforms), Signals and Systems, Programming (preferably Python or MATLAB)

Course Objectives:

The course aims to:

1. To introduce the fundamental concepts and mathematical principles of digital images, including image formation, sampling, quantization, and basic image representation techniques.
2. To understand and apply image transform techniques for efficient image analysis, representation, and processing in the transform domain.
3. To develop the ability to improve the visual quality and interpretability of digital images using spatial and frequency domain enhancement techniques.
4. To understand and apply techniques for restoring degraded images and processing colour images to improve image quality and fidelity.
5. To learn techniques for segmenting digital images and applying morphological operations to extract, analyze, and refine image structures.
6. To understand and apply image compression techniques that reduce data redundancy while preserving acceptable image quality for efficient storage and transmission.

Course Outcomes:

At the end of the course students will be able to

1. Explain the basic concepts of digital image representation, sampling, and quantization
2. Apply image transform techniques to analyze and process images in the transform domain.
3. Apply image enhancement techniques to improve the visual quality of digital images.
4. Apply image restoration and colour processing techniques to improve degraded images and manage colour information effectively.
5. Apply segmentation and morphological techniques to extract and analyze meaningful regions and structures from digital images.
6. Explain and apply basic image compression methods to reduce image size while maintaining acceptable visual quality.

SECTION-I

Unit 1: Digital Image Fundamentals (05)

Introduction to Digital Image Processing & Applications, Image Digitalization, Sampling and quantization, Relationship between pixels.

Unit 2: Image Transforms (08)

Fourier transformation, Properties of FT, 2-D FFT, Properties. Walsh transforms, Hadamard Transform, Discrete cosine Transform, KL Transform, Discrete Wavelet Transform.

Unit 3: Image Enhancement (08)

Introduction, Image Enhancement in Spatial Domain, Enhancement Through Point Operation, Types of Point Operation, Histogram Manipulation, gray level Transformation, local or neighbourhood operation, median filter, spatial domain high- pass filtering

SECTION-II

Unit 4: Image Restoration and Colour Image Processing (08)

Degradation Model, Algebraic Approach to Restoration, Inverse Filtering, Least Mean Square Filters, Constrained Least Squares Restoration, difference between image enhancement and image restoration, Primary and Secondary colours, colour characteristics, chromaticity diagram, RGB colour model, HIS colour model, conversion from one model to another,

Unit 5: Image segmentation & Morphological Image Processing (07)

Image segmentation: Detection of discontinuities. Edge linking and boundary detection, Thresholding, Region oriented segmentation

Morphological Image Processing: Dilation and Erosion, Dilation, Structuring Element Decomposition, Erosion, Combining Dilation and Erosion, Opening and Closing, the Hit or Miss Transformation

Unit 6: Image Compression (06)

Redundancies and their Removal Methods, Fidelity Criteria, Image Compression Models, Huffman and Arithmetic Coding, Error Free Compression, Lossy Compression, Lossy and Lossless Predictive Coding, Transform Based Compression, JPEG 2000 Standards.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

1. Simulation and Display of an Image, Negative of an Image (Binary & Gray Scale)
2. Implementation of Relationships between Pixels
3. Implementation of Transformations of an Image
4. Contrast stretching of a low contrast image, Histogram, and Histogram Equalization
5. Display of bit planes of an Image
6. Display of FFT (1-D & 2-D) of an image
7. Computation of Mean, Standard Deviation, Correlation coefficient of the given Image
8. Implementation of Image Smoothing Filters (Mean and Median filtering of an Image)
9. Implementation of image sharpening filters and Edge Detection using Gradient Filters
10. Image Compression by DCT, DPCM, HUFFMAN coding

11. Implementation of image restoring techniques
12. Implementation of Image Intensity slicing technique for image enhancement
13. Canny edge detection Algorithm

Text Books:

1. Digital Image Processing- Rafeal C. Gonzalez, Richard E. Woods, 3 rd Edition, Pearson, 2008
2. Digital Image Processing- S Jayaraman, S. Essakkirajan, T. Veerakumar-TMH,2010

Reference Books:

1. Introduction to Digital Image Processing with Matlab, Alasdair McAndrew,Thomson Course Technology
2. Fundamentals of Digital Image Processing-A.K. Jain, PHI, 1989
3. Digital Image Processing and computer Vision-Somka, Halavac, Boyle - Cengage learning (Indian edition)2008
4. Digital Image Processing using Matlab, Rafeal C. Gonzalez, Richard E. Woods, Steven L. Eddins, Pearson Education.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI
ECEPEC-03A – Principles of Soft Computing

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Tutorial: - 1 Hrs/Week, 1 credit

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

ICA - 25 Marks

Introduction:

Principles of Soft Computing introduces intelligent computational techniques such as neural networks, fuzzy logic, and evolutionary algorithms to solve complex real-world problems that are difficult to model using conventional hard computing methods.

Prerequisite:

Students should have basic knowledge of Engineering Mathematics (Linear Algebra, Probability and Statistics), Fundamentals of Artificial Intelligence, Data Structures and Algorithms, Programming using C/C++ or Python.

Course Objectives:

The course aims to:

1. Introduce the concepts, scope, and importance of soft computing techniques.
2. Provide in-depth understanding of artificial neural networks and their learning mechanisms.
3. Explain supervised, unsupervised, and associative learning networks.
4. Develop knowledge of fuzzy logic theory, fuzzy inference, and control systems.
5. Explain evolutionary computation techniques such as genetic algorithms and differential evolution.
6. Familiarize students with hybrid soft computing techniques and real-world engineering applications.

Course Outcomes:

At the end of the course students will be able to

1. Explain the fundamentals, scope, and components of soft computing.
2. Analyze artificial neural network models and learning algorithms.
3. Design supervised, unsupervised, and associative neural network models.
4. Apply fuzzy logic concepts for modeling, decision making, and control systems.
5. Implement genetic algorithms and differential evolution techniques for optimization problems.
6. Analyze hybrid soft computing techniques and apply them to real-world engineering problems.

SECTION-I

Unit 1: Introduction to Soft Computing and Intelligent Systems (07 Hours)

Introduction to soft computing: concepts, characteristics, and limitations of hard computing, Relationship between soft computing, machine learning, deep learning, and data science, Fundamental concepts of artificial neural networks, Basic models of artificial neural networks, Important terminologies of ANNs, McCulloch–Pitts neuron model, Linear separability and Hebbian learning

Unit 2: Supervised Learning and Regression Models (07 Hours)

Machine learning paradigm: supervised vs unsupervised learning, Linear regression and multiple linear

regression, Logistic regression and classification problems, Gradient descent optimization, Perceptron learning algorithm, Adaptive Linear Neuron (ADALINE), Introduction to bias–variance trade-off

Unit 3: Neural Networks and Deep Learning Models (07 Hours)

Multi-layer perceptron (MLP), Back propagation algorithm, Activation functions and loss functions, deep neural networks (DNNs), Convolutional Neural Networks (CNNs): architecture and applications, Introduction to recurrent neural networks (RNNs)

SECTION II

Unit 4: Unsupervised Learning and Clustering Techniques (07 Hours)

Unsupervised learning concepts, K-means clustering, Hierarchical clustering, Self-Organizing Maps (SOM), Dimensionality reduction: PCA overview, Applications of clustering in data science

Unit 5: Fuzzy Logic and Decision-Making Systems (07 Hours)

Classical sets and fuzzy sets, Membership functions and fuzzification, Fuzzy relations and operations, Fuzzy rule base and inference systems, Defuzzification techniques, Fuzzy decision-making and fuzzy logic controllers, Applications of fuzzy systems in intelligent decision support

Unit 6: Evolutionary Algorithms, Hybrid Models and Data Science Applications (07 Hours)

Genetic algorithms: operators and optimization process, Differential evolution, Hybrid soft computing models: neuro-fuzzy and genetic-neural systems, Introduction to ensemble learning

Stability considerations in intelligent systems, Applications of soft computing in machine learning and data science

Internal Continuous Assessment (ICA):

ICA consists of minimum 8 Practical's based upon above curriculum:

List of Practicals:

1. Introduction to MATLAB / Python environment for soft computing.
2. Implementation of McCulloch–Pitts neuron model.
3. Perceptron learning algorithm implementation.
4. ADALINE and MADALINE network implementation.
5. Back-propagation neural network implementation.
6. Radial basis function network implementation.
7. Self-organizing feature map (SOM) implementation.
8. Hopfield network for pattern storage and recall.
9. Fuzzy set operations and membership function design.
10. Fuzzy inference system implementation.
11. Fuzzy logic controller design.
12. Genetic algorithm for function optimization.
13. Differential evolution algorithm implementation.
14. Neuro-fuzzy hybrid system implementation.
15. Mini project based on real-world soft computing application.

Text Books:

1. S. N. Sivanandam and S. N. Deepa, Principles of Soft Computing, Wiley India.
2. J.-S. R. Jang, C.-T. Sun, and E. Mizutani, Neuro-Fuzzy and Soft Computing, Pearson.
3. Simon Haykin, Neural Networks and Learning Machines, Pearson.

Reference Books:

1. Timothy J. Ross, Fuzzy Logic with Engineering Applications, Wiley.
2. David E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Pearson.
3. MathWorks Documentation on Neural Network, Fuzzy Logic, and GA Toolboxes.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPEC-03B – Mobile Application Development using Android

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Tutorial: - 1 Hrs/Week, 1 credit

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

ICA - 25 Marks

Introduction:

Mobile Application Development using Android focuses on designing, developing, testing, and deploying Android-based mobile applications using modern development tools and frameworks.

Prerequisite:

Students should have basic knowledge of computer programming using C/Java, Object-Oriented Programming concepts, basics of Data Structures, fundamentals of Operating Systems, and basic understanding of databases and computer networks.

Course Objectives:

The course aims to:

1. Introduce the fundamentals of mobile computing and mobile application development.
2. Provide understanding of Android platform architecture, operating system, and application framework.
3. Familiarize students with Android application components and their lifecycle management.
4. Develop skills to design intuitive and user-friendly mobile application interfaces using Android UI tools.
5. Expose students to data storage, networking, and advanced features in Android applications.
6. Emphasize testing, deployment, performance optimization, and security aspects of mobile applications.

Course Outcomes:

At the end of the course students will be able to

1. Explain the concepts of mobile computing and the architecture of the Android platform.
2. Develop Android applications using core components such as activities, intents, services, and broadcast receivers.
3. Design user-friendly mobile application interfaces using layouts, UI widgets, and Material Design principles.
4. Implement data storage, database, and networking functionalities in Android applications.
5. Integrate advanced Android features such as fragments, sensors, GPS, notifications, and Google Maps.
6. Test, deploy, optimize, and secure Android applications following industry best practices.

SECTION-I

Unit 1: Introduction to Mobile Application Development

(07)

Mobile computing overview, Mobile application types, Android platform overview Android architecture, Android OS components, Android SDK and Android Studio Android application structure

Unit 2: Android Application Components (07)
Activities and Activity lifecycle, Intents (explicit & implicit), Services, Broadcast Receivers, Content Providers, Android Manifest file, Resource management

Unit 3: User Interface Design (07)
Views and ViewGroups, Layouts: LinearLayout, RelativeLayout, ConstraintLayout, FrameLayout, UI widgets: TextView, EditText, Button, ImageView, Event handling, Menus and dialogs, Material Design concepts

SECTION II

Unit 4: Data Storage and Networking (07)
Shared Preferences, Internal & External storage, SQLite database, Content Providers with SQLite, Networking basics, RESTful web services, JSON parsing, Permissions and security

Unit 5: Advanced Android Concepts (07)
Fragments, RecyclerView and ListView, Adapters, Background processing (Threads, AsyncTask) Notifications, Location-based services (GPS), Sensors, Google Maps integration

Unit 6: Testing, Deployment and Security (07)
Debugging Android applications, Testing (Unit testing, UI testing), APK generation and signing, App deployment on Google Play Store, Performance optimization, Mobile application security

Internal Continuous Assessment (ICA):

ICA consists of a minimum **ten** experiments based upon the above curriculum.

List of Practicals:

1. Installation and Configuration of Android Studio
Creating and running a simple Android application (Hello Android).
2. Study of Android Project Structure and Manifest File
Understanding activity lifecycle and resource management
3. Design of User Interface using Layouts
Implementation using LinearLayout, RelativeLayout, and ConstraintLayout.
4. Implementation of UI Widgets and Event Handling
TextView, EditText, Button, ImageView with event listeners.
5. Menus and Dialogs Implementation
Options menu, context menu, and alert dialogs.
6. Intent-Based Application Development
Implementation of explicit and implicit intents.
7. Service and Broadcast Receiver Implementation
Creating background services and handling system broadcasts.
8. Shared Preferences Implementation
Storing and retrieving simple data.
9. SQLite Database Application
CRUD operations using SQLite database.
10. RecyclerView / ListView Implementation
Displaying dynamic data using adapters.
11. Networking and API Integration
Consuming RESTful web services and JSON parsing.

12. Location-Based Application Development
GPS-based application using Google Maps.
 13. Sensor-Based Android Application
Accessing accelerometer, proximity, or light sensors.
 14. Notification and Background Processing
Implementation of notifications using threads / AsyncTask.
 15. Mini Project using Android Platform
Design, development, testing, and documentation of a complete Android application.
-

Text Books:

1. Reto Meier, Professional Android Development, Wrox
 2. Bill Phillips, Android Programming: The Big Nerd Ranch Guide
-

Reference Books:

1. Wei-Meng Lee – Beginning Android Application Development
2. Charlie Collins et al. – Android in Practice
3. Android Developers Documentation – developer.android.com



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI
ECEPEC-03C – Blockchain Technology

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Tutorial: - 1 Hrs/Week, 1 credit

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

ICA - 25 Marks

Introduction:

Blockchain Technology focuses on distributed ledger systems, cryptographic security mechanisms, consensus protocols, and smart contract development for building secure and decentralized applications.

Prerequisite:

Students should have basic knowledge of computer networks, cryptography fundamentals, data structures, database systems, and programming (Python/Java preferred).

Course Objectives:

The course aims to:

1. To understand the fundamentals of block chain technology
2. To explain the architecture and working of block chain systems.
3. To analyse different types of block chains
4. To evaluate block chain security, privacy, and scalability issues
5. To explore applications of block chain technology
6. To learn bitcoin and its process also the blockchain technology in allied technologies.
7. To develop problem-solving skills using block chain concepts

Course Outcomes:

At the end of the course students will be able to

1. Explain the fundamental concepts of block chain technology, including distributed ledgers, Cryptography, and consensus mechanisms.
2. Analyse the architecture and working of different block chain systems such as Bitcoin, Ethereum, and Hyperledger.
3. Apply cryptographic techniques and hashing algorithms used in blockchain for secure transaction Processing.
4. Design and implement basic smart contracts and decentralized applications (DApps) using Suitable blockchain platforms.
5. Evaluate blockchain security, scalability, privacy challenges, and propose suitable mitigation Strategies.
6. Understand and illustrate Blockchain with allied technologies such as cloud computing, AI, IoT, Robotics

SECTION-I

Unit 1: Introduction & Core Concepts

(07)

What is Blockchain? Definitions & motivation, Distributed ledgers vs centralized systems, History and evolution of blockchain technology, Components of a blockchain (blocks, chains, nodes), Types of

blockchains (public, private, consortium), Key features: immutability, decentralization, transparency, Cryptographic foundations: hash functions and digital signatures basics

Unit 2: Cryptography Primitives & Consensus Mechanisms (07)

Cryptographic primitives used in blockchain: Hash functions, Asymmetric cryptography (public/private keys), Digital signatures, Consensus mechanisms: Proof of Work (PoW), Proof of Stake (PoS), other consensus approaches (overview), Distributed consensus and fault tolerance fundamentals

Unit 3: Bitcoin Basics Fundamentals (07)

Bitcoin architecture and how Bitcoin works, Transaction structure and block mining, Bitcoin ledger, wallets, addresses, Other cryptocurrencies overview, Tokens vs coins, Transaction validation & basic economics of mining

SECTION II

Unit 4: Smart Contracts & Blockchain Platforms (09)

Smart contracts: concept and characteristics ,Ethereum platform overview, Ethereum Virtual Machine (EVM) basics, Introduction to smart contract development, Tools & languages (e.g., Solidity, Remix) — basic exposure, Hyperledger frameworks / permissioned blockchains (concepts)

Unit 5: Applications, Challenges (06)

Real-world applications of blockchain (finance, supply chain, healthcare, etc.), Blockchain in industry and enterprise settings, Security, privacy, scalability, and regulatory issues, Limitations of blockchain and future trends, Decentralized applications (DApps)

Unit 6: Blockchain and Allied Technologies (06)

Blockchain and Cloud Computing, Characteristics of Blockchain Cloud, Blockchain and Artificial Intelligence, Blockchain and IoT, Blockchain and Machine Learning, Blockchain and Robotic Process Automation

Internal Continuous Assessment (ICA):

ICA consists of minimum eight experiments based upon above curriculum.

List of Practical:

1. SHA-256 Hashing: To implement SHA-256 hashing for data integrity.
Tools: Python, Procedure: Install Python, Import hashlib, Convert input to bytes, Generate hash
2. Digital Signature (RSA): To generate and verify digital signatures.
Tools: Python, cryptography library
3. Merkle Tree Implementation: To verify transaction integrity using Merkle Tree.
4. Simple Blockchain Implementation: To implement a basic blockchain structure
Create a basic blockchain in Python / Java, Block creation, chaining, and validation
5. Proof of Work (PoW): Implement PoW consensus algorithm, Difficulty adjustment and nonce discovery
6. Proof of Stake (PoS) – Conceptual / Simulation, Stake-based validator selection, Comparison with PoW
7. Smart Contract Basics. Introduction to Solidity, Write and deploy a basic smart contract
8. Ethereum Development Environment:Install and configure:Ganache,Truffle / Hardhat, MetaMask
9. Smart Contract Deployment:Deploy contract on local Ethereum blockchain
10. Decentralized Application (DApp),Develop a simple DApp (frontend + smart contract)
Example: Voting / Token system(Interact using Web3.js / Ethers.js)

11. Token Creation: Implement ERC-20 or ERC-721 (NFT) token (Transfer and balance checking)

Text Books:

1. Kumar Saurabh and Ashutosh Saxena., —Blockchain Technology: Concepts and Applications, Wiley Publications
2. Yathish R , Tejaaswini N, Blockchain For Beginners, Publisher: Shroff/X-Team 2019 Edition
3. Don Tapscott, author of Wikinomics, Alex Tapscott, — Blockchain Revolution: How the technology behind bitcoin and other cryptocurrencies is changing the world, Penguin Publishing Group
4. Narayanan, Bonneau, Felten, Miller and Goldfeder, “Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction”, Princeton University Press.

Reference Books:

1. Narayanan, Bonneau, Felten, Miller and Goldfeder, —Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction, Princeton University Press.
2. Josh Thompson, ‘Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming’, Create Space Independent Publishing Platform, 2017.
3. Imran Bashir, —Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained, Packt Publishing.
4. Merunas Grincalaitis, —Mastering Ethereum: Implement Advanced Blockchain Applications Using Ethereum-supported Tools, Services, and Protocols, Packt Publishing



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECEPEC – 03D– Computer Vision

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Tutorial: - 1 Hours/week, 1 Credits

Examination Scheme

ESE - 70 Marks

ISE - 30 Marks

ICA – 25 Marks

Introduction:

This course is an introduction to the field of Computer Vision (CV), Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs — and take actions or make recommendations based on that information. If AI enables computers to think, computer vision enables them to see, observe and understand.

Prerequisite:

Students should have knowledge of basic programming, mathematics, and basic machine learning concepts.

Course Objectives:

The course aims to:

1. To understand the fundamentals of computer vision and digital image processing
2. To introduce the processes involved image enhancement and restoration.
3. To facilitate the students to gain understanding color image processing and morphology.
4. To impart the knowledge of image segmentation and object recognition techniques.

Course Outcomes:

At the end of the course students will be able to

1. Explain the fundamentals of computer vision and its applications.
2. Apply the image enhancement techniques for smoothing and sharpening of images.
3. Compare the different image restoration and segmentation techniques.
4. Demonstrate the smoothing and sharpening techniques for color images.
5. Explain morphological, feature extraction, and pattern classification techniques for object recognition.

SECTION-I

Unit – 1 Introduction:

(05)

What is computer vision? A brief history. Digital Image fundamentals, Image Sensing and acquisition, Sampling and Quantization, Image formation models, Overview of Computer Vision, Applications of Image processing and Computer Vision.

Unit – 2 Image Enhancement;

(09)

Image enhancement in spatial domain, Basic grey level Transformations, Histogram Processing Techniques, Spatial Filtering, Image smoothing and Image Sharpening, Image enhancement process in frequency domain, Low pass filtering, High pass filtering

Unit – 3 Image Segmentation:**(07)**

Fundamentals, point, line and edge detection, Thresholding (Foundation & Basic global thresholding only), Segmentation by region growing & region splitting & merging, Edge linking and boundary detection.

SECTION-II**Unit – 4 Feature Extraction:****(06)**

Importance of Features, Feature extraction techniques, Histogram of Oriented Gradient (HOG), Scale Invariant Feature Transform (SIFT), Background subtraction techniques, Image Matching, Principal Component Analysis (PCA)

Unit – 5 Object Recognition and Motion Estimation:**(09)**

Object Recognition techniques: Viola-Jones, Yolo, Deep learning algorithms for Object Recognition. Optical Flow, Gaussian Mixture Model (GMM), Structure of Motion, Motion Estimation

Unit – 6 Applications of Computer Vision:**(06)**

Machine Learning Algorithms and their Applications in Medical, Image Segmentation, Motion Estimation and Object Tracking, Face and Facial Expression Recognition, Gesture Recognition, Image Fusion, Programming Examples

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical's / Tutorials:

1. Implement various grey level transformations for Image Enhancement.
2. Implement Histogram Equalization technique.
3. Write a Program to apply convolution processes on an input image for image smoothing.
4. Implement Histogram of Oriented Gradient (HOG) for Feature extraction.
5. Write a Program to apply Scale Invariant Feature Transform on input image.
6. Implement frame differencing technique for background subtraction from video.
7. Implement Principal Component Analysis for the computation of Eigenvector to reduce the dimensionality.
8. Implement object detection algorithm YOLO.
9. Implement R-CNN algorithms for object detection.
10. Implement motion estimation using optical flow technique.
11. Implement Object recognition.
12. Implement Facial Expression Recognition.

Text Books:

1. Digital Image Processing- Refael C. Gonzalez and Richard E. Woods, Wesley
2. Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.
3. Computer Vision and Image Processing, Manas Kamal Bhuyan, CRC Press

Reference Books:

1. Introductory Techniques for 3D Computer Vision, by E. Trucco and A. Verri, Publisher: Prentice Hall.
2. Computer Vision, D. H. Ballard, C. M. Brown, Prentice-Hall, Englewood Cliffs, 1982.
3. Image Processing, Analysis, and Machine Vision, Milan Sonka, Vaclav Hlavac, Roger Boyle, Thomson Learning



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B. Tech (Electronics and Computer Engineering)

SEMESTER-VI

ECESEC-02: Projects on Industrial Application

Teaching Scheme

Practical–4 Hours/week, 2 Credits

Examination Scheme

ICA – 25 Marks

POE –50 Marks

Introduction:

Projects focusing on industrial applications aim to bridge the gap between academic learning and real-world industry needs. These projects provide students with hands-on experience in designing, developing, and implementing solutions that address practical industrial challenges.

Course Prerequisite:

To successfully undertake projects on industrial applications, students should have a strong foundation in relevant technical, analytical, and problem-solving skills.

Course Objectives:

1. **Understanding Industrial Requirements** – Identify challenges faced in industries like healthcare, finance, logistics, or manufacturing.
2. **Technology Implementation** – Develop a working solution using programming, databases, networking, cloud computing, IoT, etc.
3. **Efficiency & Automation** – Improve productivity, reduce manual effort, and optimize processes.
4. **Security & Compliance** – Ensure the system is secure and follows industry regulations.
5. **Scalability & Performance** – Design a system that can handle increasing workloads efficiently.
6. **User-Friendly Design** – Develop an intuitive and easy-to-use interface for end-users.

Course Outcomes:

At the end of this course, students will be able to

1. **Develop a Fully Functional Software/System** – A web app, mobile app, database system, or automation tool applicable to an industrial sector.
2. **Apply Engineering Concepts** – Use programming, data structures, algorithms, networking, and security in a real-world scenario.
3. **Improve Problem-Solving Skills** – Learn to analyze industrial challenges and create effective solutions.
4. **Understand Deployment & Maintenance** – Implement CI/CD pipelines, cloud integration, and post-deployment monitoring.
5. **Enhance Teamwork & Communication** – Work collaboratively, document findings, and present a professional-level project report.
6. **Meet Industry Standards** – Ensure the project aligns with software development best practices, including security, performance, and compliance.

Note:

1. **Group Size:** Each group should consist of **3 to 5 students** (or as per faculty guidelines).
2. **Diverse Skill Sets:** Teams should have a mix of skills (coding, database management, documentation, testing, etc.).
3. **Faculty Approval:** Group formation and project topics should be approved by the faculty before starting.
4. **No Changes after Formation:** Once groups are finalized, switching members is not allowed unless there is a valid reason.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI

ECEMDM-04A - Internet of Things (IoT)

Teaching Scheme

Lectures : - 2 Hrs/Week, 2 credits

Practical : - 2 Hrs/Week, 1 credit

Examination Scheme

ESE: 70 Marks

ISE: 30 Marks

ICA: 25 Marks

Introduction:

This course introduces the fundamental concepts of Internet of Things (IoT), enabling technologies, communication protocols, cloud integration, and security aspects. It emphasizes practical implementation of IoT systems using embedded platforms like Arduino and Raspberry Pi for real-world smart applications.

Prerequisite:

Basic knowledge of Microcontrollers, Embedded C programming, Computer Networks fundamentals, and Sensors & Actuators.

Course Objectives:

The course aims to:

1. To understand core IoT concepts, architectures, and enabling technologies like sensors and networking protocols.
2. To apply microcontroller interfacing skills from prior courses to develop IoT devices and systems.
3. To analyze communication protocols and data processing for IoT deployment in real-world scenarios.
4. To evaluate security challenges and design IoT solutions for applications like smart systems.

Course Outcomes:

Upon completion of the course, students will be able to:

1. Explain IoT fundamentals, components, and architectures.
2. Design and interface sensors/actuators with microcontrollers for IoT prototypes.
3. Implement IoT communication using protocols like MQTT/CoAP on embedded platforms.
4. Develop and assess IoT applications integrating cloud services and security.

SECTION I

Unit 1: IoT Introduction

(05)

IoT definition, characteristics, evolution from M2M, functional blocks, physical/logical design, applications in domains like smart homes/agriculture.

Unit 2: Sensors and Actuators

(05)

Classification, Types, characteristics, interfacing with microcontrollers (ADC/DAC), signal conditioning

Unit 3: Embedded Platforms

(05)

Arduino/Raspberry Pi boards, IDE setup, embedded C programming for I/O, timers, interrupts; prototyping basics building on prior microcontroller knowledge

SECTION II

Unit 4: IoT Communication (05)
Wireless protocols (WiFi, Bluetooth LE, Zigbee, LoRa, RFID, 6LoWPAN), MAC/routing in WSNs, gateway concepts, IPv6 addressing; power management and network topologies.

Unit 5: Protocols and Cloud (06)
Application protocols (MQTT, CoAP, HTTP), data aggregation, cloud platforms (AWS IoT, ThingSpeak), APIs, edge computing, SDN basics for IoT; integration with Raspberry Pi/Arduino.

Unit 6: Security and Applications (04)
IoT challenges (privacy, scalability), authentication, encryption (AES), case studies (smart grid, healthcare, agriculture).

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

1. Interface LED and button with Arduino; blink at intervals.
2. Read DHT11 (temperature/humidity) sensor on Arduino and display on LCD.
3. Control DC motor speed using PWM and potentiometer on Arduino.
4. Ultrasonic sensor for obstacle detection with buzzer alert on Raspberry Pi.
5. PIR motion sensor alarm system using Arduino.
6. Soil moisture sensor for automatic watering using NodeMCU/ESP8266.
7. MQTT publish/subscribe temperature data to cloud (ThingSpeak).
8. RFID-based door access control with servo lock on Arduino.
9. Gas/LPG sensor (MQ-2) alert system with buzzer/LED on Raspberry Pi.
10. IR sensor for line-following robot prototype on Arduino.
11. Light intensity (LDR) control for streetlight automation on Arduino.
12. Integrate multiple sensors (DHT11 + PIR) and log data to the cloud via WiFi.

Textbooks:

1. Raj Kamal, “Internet of Things: Architecture and Design Principles”, McGraw Hill Education, latest edition.
2. Arshdeep Bahga, Vijay Madisetti, “Internet of Things: A Hands-on Approach”, Universities Press, latest edition

Reference Books:

1. Cuno Pfister, “Getting Started with the Internet of Things: Connecting Sensors and Microcontrollers to the Cloud”, O’Reilly.
2. Adrian McEwen, Hakim Cassimally, “Designing the Internet of Things”, Wiley.
3. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things: Key Applications and Protocols”, Wiley.
4. Pethuru Raj, Anupama C. Raman, “The Internet of Things: Enabling Technologies, Platforms, and Use Cases”, CRC Press.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI

ECEMDM-04B -Data Visualization Tools (Tableau & Power BI)

Teaching Scheme

Lectures: - 2 Hrs/Week, 2 credits

Practical: - 2 Hrs/Week, 1 credit

Examination Scheme

ESE: 70 Marks

ISE: 30 Marks

ICA: 25 Marks

Introduction:

This course focuses on principles of data visualization and business intelligence using modern tools such as Power BI and Tableau. It enables students to transform raw data into meaningful insights through interactive dashboards, reports, and visual analytics for engineering and industry applications.

Prerequisite:

Basic computer operations, fundamental understanding of data, introductory statistics, spreadsheets (Excel), and basic SQL/Python knowledge.

Course Objectives:

The course aims to:

1. Introduce students to fundamentals of data visualization and business intelligence.
2. Enable students to understand data types, data sources, and data preparation techniques.
3. Develop skills to create interactive dashboards using Power BI and Tableau.
4. Apply visualization techniques for data-driven decision making.
5. Interpret and communicate insights effectively using visual analytics.
6. Prepare students for industry-oriented BI and analytics roles.

Course Outcomes:

At the end of the course students will be able to

7. Understand principles, importance, and applications of data visualization.
8. Prepare, clean, and transform datasets for visualization tools.
9. Create basic and advanced visualizations using Power BI and Tableau.
10. Design interactive dashboards and reports for real-world datasets.
11. Analyse data patterns and trends to support decision-making.
12. Apply visualization techniques in engineering and interdisciplinary domains.

SECTION-I

Unit 1: Introduction to Data Visualization & Business Intelligence (05)

Introduction to data visualization, Importance of data visualization in engineering and business, Data visualization vs data analysis, Basics of Business Intelligence (BI), Visualization best practices and principles, Types of data: structured, semi-structured, unstructured, Overview of Power BI and Tableau, Applications in Electronics, Computer Engineering, and Industry

Unit 2: Data Preparation and Data Modelling (05)

Data sources: Excel, CSV, databases, web data, Data collection and data quality issues, Data cleaning and transformation concepts, Introduction to Power Query / Tableau Prep, Data modelling concepts, Relationships, cardinality, and normalization, Calculated columns and measures (introductory level)

Unit 3: Visualization Fundamentals and Charts

(05)

Visual perception and design principles, choosing the right chart, Basic charts: bar, column, line, pie, area, Advanced charts: stacked charts, scatter plots, maps, Filters, slicers, and sorting, Formatting and customization of visuals, Data story telling concepts.

SECTION-II

Unit 4: Power BI – Dashboard and Report Development

(05)

Power BI architecture and workflow, Importing and transforming data, DAX basics: measures, functions, aggregations, creating interactive reports, Dashboard creation and layout design, Publishing and sharing reports, Power BI service overview

Unit 5: Tableau – Visualization and Analytics

(05)

Tableau interface and architecture, connecting to data sources, Dimensions and measures calculated fields and parameters, Creating interactive worksheets, Dashboards and actions, Tableau storytelling feature

Unit 6: Advanced Visualization & Case Studies

(05)

Comparative study: Power BI vs Tableau, KPI design and performance dashboards, Time-series and trend analysis, Visualization for IoT, electronics data, and system logs, Case studies in engineering, finance, healthcare, Introduction to real-time and big data visualization, Ethical issues and data privacy in visualization

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

1. Introduction to Power BI and Tableau interface
 2. Importing datasets from Excel/CSV
 3. Data cleaning and transformation using Power Query / Tableau Prep
 4. Creating basic charts and visuals
 5. Implementing filters and slicers
 6. Creating calculated fields and measures
 7. Designing interactive dashboards in Power BI
 8. Dashboard creation in Tableau
 9. KPI and performance dashboard design
 10. Case study-based visualization (engineering dataset)
 11. Comparative dashboard using Power BI and Tableau
-

Text Books:

1. Murray, D., Beginning Power BI, Apress (<https://content.e-bookshelf.de/media/reading/L-13393976-1ef18135ae.pdf>)
 2. Ryan Sleeper, Practical Tableau, O'Reilly Media (<https://dynamicsystemindia.in/pdfs/pdfs/tableau.pdf>)
-

Reference Books:

1. Stephen Few, Show Me the Numbers, Analytics Press
2. Cole Nussbaumer Knaflic, Storytelling with Data, Wiley
3. Alberto Cairo, The Functional Art, New Riders
4. Microsoft Power BI and Tableau official documentation



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

Honors in AI TECHNOLOGY

ECEHON-04A AI Applications

Teaching Scheme:

Lecture: 3Hrs/week, 3 credits

Practical : 2Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

Artificial Intelligence (AI) has become a core enabling technology across industries such as healthcare, finance, manufacturing, transportation, agriculture, and smart systems. This course on **AI Applications** focuses on how AI techniques are practically applied to solve real-world engineering and societal problems. The course bridges the gap between theoretical AI concepts and their implementation in diverse application domains using modern tools and frameworks.

Prerequisite: Students should have basic knowledge of Python Programming, Data Structures and Algorithms, Probability and Statistics, Fundamentals of Machine Learning, Linear Algebra (basic level)

Course Objectives:

The course aims to:

1. Understand the role of AI in modern engineering applications.
2. Apply AI techniques to solve domain-specific problems.
3. Analyze real-world case studies involving AI systems.
4. Design and implement AI-based solutions using suitable tools.
5. Evaluate ethical, societal, and performance aspects of AI applications.
6. Develop practical skills required for industry-oriented AI projects.

Course Outcomes:

At the end of the course students will be able to

1. Explain fundamental concepts and architectures used in AI applications.
2. Apply machine learning and deep learning techniques to real-world problems.
3. Develop AI-based solutions for image, speech, and text data.
4. Analyze AI applications in healthcare, smart cities, and industrial automation.
5. Implement AI models using standard programming frameworks.
6. Evaluate ethical, legal, and social implications of AI technologies.

SECTION-I

Unit 1: Introduction to AI Applications

(07)

Overview of Artificial Intelligence and its evolution ,AI vs Machine Learning vs Deep Learning, Types of AI Applications, AI Application Architecture, Data-driven decision making, AI development lifecycle, Case studies of successful AI deployments

Unit 2: AI Applications in Data Analytics and Business (07)

AI in data analytics and predictive modeling ,Recommendation systems ,Customer behavior analysis ,Fraud detection systems, AI in finance and banking, Natural Language Processing (NLP) for business intelligence, Chatbots and virtual assistants, Case study: AI-driven business automation

Unit 3: AI Applications in Computer Vision (07)

Fundamentals of computer vision ,Image classification and object detection ,Face recognition systems ,Video analytics and surveillance ,Optical Character Recognition (OCR), Medical image analysis, Autonomous vision systems, Case study: Smart surveillance and healthcare imaging

SECTION-II

Unit 4: AI Applications in Natural Language and Speech Processing (07)

Basics of Natural Language Processing, Text preprocessing and feature extraction, Sentiment analysis, Machine translation, Speech recognition systems, Voice-based assistants, Case study: AI-powered customer support systems

Unit 5: AI Applications in Smart Systems and IoT (07)

AI-enabled IoT architecture, Smart homes and smart cities, Predictive maintenance, AI in industrial automation, Robotics and intelligent control, Edge AI concepts, Case study: Smart energy management systems

Unit 6: Emerging AI Applications and Ethical Issues (07)

AI in healthcare and medical diagnosis ,AI in autonomous vehicles ,AI in agriculture and environmental monitoring ,Generative AI applications ,Ethical issues in AI ,Bias, fairness, and explainable AI ,AI security and privacy ,Future trends in AI applications

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Practical:

1. Introduction to Python libraries for AI applications
 2. Data preprocessing and visualization using real datasets
 3. Implementation of a recommendation system
 4. Fraud detection using machine learning
 5. Image classification using CNN
 6. Face detection using OpenCV
 7. OCR implementation using Tesseract
 8. Sentiment analysis on social media data
 9. Speech-to-text conversion using AI tools
 10. Chatbot development using NLP techniques
 11. AI-based IoT predictive maintenance model
 12. Mini-project on real-world AI application (team-based)
-

Text Books:

1. Artificial Intelligence: A Modern Approach, Stuart Russell & Peter Norvig, Pearson Education, 4th Edition.

2. Deep Learning, Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, 1st Edition.
3. Machine Learning, Tom M. Mitchell, McGraw-Hill Education, 1st Edition.
4. A First Course in Artificial Intelligence, Deepak Khemani, McGraw-Hill Education, 1st Edition.
5. Generative Deep Learning, David Foster, O'Reilly Media, 2nd Edition.

Reference Books:

1. Machine Learning: A Probabilistic Perspective, Kevin P. Murphy, MIT Press, 1st Edition.
2. Python Machine Learning, Sebastian Raschka, Packt Publishing, 3rd Edition.
3. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurélien Géron, O'Reilly Media, 2nd Edition.
4. Introduction to Artificial Intelligence, Nils J. Nilsson, Morgan Kaufmann, 1st Edition.
5. The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World, Pedro Domingos, Basic Books, 1st Edition.
6. Online Resources a. Generative AI for Beginners – Microsoft b. DeepLearningBook.org



Punyashlok Ahilyadevi Holkar Solapur University, Solapur
Faculty of Science and Technology
Third Year B.Tech (Electronics and Computer Engineering)
SEMESTER-VI
Honors in Internet of Things (IOT) and Smart Systems
ECEHON-04B IoT Security and Privacy

Teaching Scheme:

Lecture: 3Hrs/week, 3 credits

Practical : 2Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

The Internet of Things (IoT) connects billions of devices that collect and exchange data through networks and cloud platforms. As IoT applications grow in areas such as smart homes, healthcare, industrial automation, and smart cities, ensuring security and privacy becomes critical. This course introduces security challenges in IoT systems and explores various mechanisms to protect IoT devices, communication networks, and user data. The course also covers cryptographic techniques, authentication protocols, secure communication methods, and privacy preservation strategies for IoT environments.

Prerequisite:

Students should have basic knowledge of:

- Computer Networks
- Embedded Systems and IoT fundamentals
- Operating Systems
- Basic Cryptography concepts

Course Objectives

The course aims to:

1. Understand the security challenges and vulnerabilities in IoT systems.
2. Study cryptographic techniques used for securing IoT communication.
3. Analyze authentication and access control mechanisms in IoT networks.
4. Explore privacy protection techniques for IoT data and users.
5. Understand security frameworks and standards for IoT applications.
6. Apply security principles to design secure IoT systems.

Course Outcomes:

At the end of the course students will be able to

1. Explain the security requirements and challenges in IoT environments.
 2. Analyze threats, attacks, and vulnerabilities in IoT systems.
 3. Apply cryptographic and authentication techniques to secure IoT communication.
 4. Evaluate privacy preservation mechanisms for IoT data.
 5. Implement basic security mechanisms for IoT devices and networks.
 6. Design secure IoT architectures for real-world applications.
-

SECTION-I

Unit 1: Introduction to IoT Security

(07)

Overview of IoT architecture and components, security requirements in IoT systems, security challenges in resource-constrained devices, threat models in IoT, types of attacks in IoT (physical attacks, network attacks, software attacks), security risks in IoT applications.

Unit 2: Cryptography for IoT

(07)

Basic cryptographic concepts, symmetric key cryptography, asymmetric cryptography, hash functions, digital signatures, and lightweight cryptographic algorithms for IoT, key management techniques for IoT devices.

Unit 3: Authentication and Access Control

(07)

Authentication mechanisms in IoT systems, device authentication protocols, identity management, access control models (RBAC, ABAC), secure device provisioning, and authentication challenges in large-scale IoT networks.

SECTION-II

Unit 4: Secure Communication in IoT

(07)

Security protocols for IoT communication, Transport Layer Security (TLS) and Datagram TLS (DTLS), secure MQTT and CoAP protocols, secure routing in IoT networks, data integrity and confidentiality techniques.

Unit 5: Privacy Protection in IoT

(07)

Privacy issues in IoT applications, data privacy and protection mechanisms, anonymization techniques, privacy preserving data collection, secure data storage and cloud privacy concerns, privacy regulations and policies.

Unit 6: IoT Security Frameworks and Case Studies

(07)

IoT security frameworks and standards, security in smart home systems, industrial IoT security, healthcare IoT security challenges, blockchain applications in IoT security, emerging trends and future research directions.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

1. Study of IoT security threats and attack models.
2. Implementation of basic encryption techniques for IoT communication.
3. Simulation of secure IoT communication using MQTT/CoAP protocols.
4. Implementation of authentication mechanisms for IoT devices.
5. Secure data transmission between IoT device and cloud platform.
6. Study of lightweight cryptographic algorithms for embedded systems.
7. IoT network vulnerability analysis using security tools.
8. Design of a secure IoT architecture for a smart application.
9. Case study on security challenges in smart city or healthcare IoT.
10. Mini project on secure IoT communication system.

Text Books:

1. Brian Russell, Drew Van Duren – **Practical Internet of Things Security**, Packt Publishing.
2. Arshdeep Bahga, Vijay Madisetti – **Internet of Things: A Hands-On Approach**, Universities Press.

Reference Books:

1. Fei Hu – **Security and Privacy in Internet of Things (IoTs)**, CRC Press.
2. William Stallings – **Cryptography and Network Security**, Pearson.
3. Dieter Gollmann – **Computer Security**, Wiley.
4. Shancang Li, Li Da Xu – **Securing the Internet of Things**, Elsevier.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

Honors in Railway Engineering

ECEHON-04C Advanced Communication and Modern Signaling in Railway

Teaching Scheme:

Lecture: 3Hrs/week, 3 credits

Practical : 2Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

Modern railway systems depend on advanced communication technologies, intelligent signaling mechanisms, and secure networking frameworks to ensure safe and efficient train operations. This course introduces modern railway communication systems, AI-enabled monitoring technologies, and next-generation signaling systems used in high-speed railways and metro networks. It also focuses on digital railway infrastructure using IoT, AI, and software-defined networking.

Prerequisite:

Students should have basic knowledge of digital communication systems, computer networks, and railway signaling concepts.

Course Objectives:

The course aims to:

1. To understand modern railway communication technologies and signaling frameworks.
2. To study next-generation railway mobile communication systems such as FRMCS and 5G-R.
3. To explore applications of Artificial Intelligence and IoT in railway monitoring and safety systems.
4. To understand intelligent signaling and collision avoidance systems used in modern railways.
5. To study secure networking and reliability techniques used in railway communication systems.

Course Outcomes:

1. At the end of the course students will be able to
2. Understand modern railway communication systems and architectures.
3. Analyze advanced signaling technologies such as Kavach and ETCS.
4. Evaluate AI-based monitoring systems used in smart railway stations.
5. Apply IoT and wireless communication technologies for railway asset monitoring.
6. Design secure and reliable railway communication networks.
7. Analyze cybersecurity and network management techniques in railway communication systems.

SECTION-I

Unit I: Fundamentals of Railway Communication Systems

(06)

Introduction to railway communication systems, evolution from conventional railway communication to digital communication networks. Overview of railway signaling and communication architecture. Basic concepts of train control communication systems. Role of communication systems in railway safety, operational efficiency, and passenger information systems.

Unit II: Next Generation Railway Mobile Communication (07)
Future Railway Mobile Communication System (FRMCS). Transition from GSM-R to LTE-R and 5G-R communication systems. 3GPP architecture for railway networks. Mission critical communication services including MCPTT, MCVideo, and MCDData. High-speed communication requirements for modern trains and integration of satellite communication systems.

Unit III: Smart Stations and IoT-Based Railway Infrastructure (07)
Concept of smart railway stations and digital ecosystems. AI-based video surveillance systems for security monitoring. Edge analytics for crowd detection, face recognition, and anomaly detection. IoT-based railway asset monitoring including OHE monitoring, axle temperature sensors, and smart passenger announcement systems. Wireless communication technologies used in railway yards.

SECTION-II

Unit IV: Intelligent Signaling and Train Control Systems (07)
Introduction to advanced railway signaling systems. Electronic interlocking systems and signaling logic. Communication-Based Train Control (CBTC) for metro and driverless trains. Concepts of cab signaling and moving block technology. Automatic train protection systems and their role in railway safety.

Unit V: Collision Avoidance and Modern Signaling Technologies (06)
KAVACH (Indigenous Automatic Train Protection system) architecture and operation. European Train Control System (ETCS) Level 2 and Level 3 signaling. Digital axle counters and advanced block signaling systems. Signal optimization and train throughput improvement using intelligent signaling techniques.

Unit VI: Secure Railway Networking and System Reliability (07)
Reliability requirements for railway communication systems. Safety Integrity Levels (SIL) for signaling systems. Predictive maintenance using AI and digital twin technology. Software defined networking for railway communication infrastructure. Cybersecurity challenges in railway signaling networks and protection of signaling data networks. Network management systems for monitoring and control.

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

- 1. Study of Railway Communication Systems**
Study the architecture of modern railway communication networks including GSM-R, LTE-R, and 5G-R systems and prepare a technical report.
- 2. Simulation of Electronic Interlocking System**
Design and simulate the logic of a **four-line railway electronic interlocking system** using digital logic simulation tools.
- 3. Study of FRMCS Architecture**
Analyze the structure and working of **Future Railway Mobile Communication System (FRMCS)** and compare it with GSM-R communication.
- 4. Configuration of VoIP Based Train Control Communication**
Configure a basic **VoIP communication system** using SIP protocol for train control communication.
- 5. Case Study on Kavach System**
Study the architecture and working of **Kavach (Automatic Train Protection)** system and prepare a comparative analysis with ETCS.

6. **Analysis of Communication-Based Train Control (CBTC)**
Study the working of **CBTC signaling system** used in metro rail networks and prepare a functional block diagram.
 7. **Design of IoT-Based Railway Monitoring System**
Design a conceptual **IoT system for railway asset monitoring** such as axle temperature monitoring or track condition monitoring.
 8. **AI-Based Video Surveillance System Study**
Analyze the role of **AI-based video surveillance systems** in railway stations for crowd monitoring and security.
 9. **Wireless Communication in Railway Yards**
Study the application of **Wi-Fi 6 / wireless communication systems** used for railway yard management.
 10. **Digital Axle Counter System Study**
Study the working principle of **Multi-Section Digital Axle Counter (MSDAC)** used for train detection and signaling.
 11. **Railway Network Security Analysis**
Prepare a **cybersecurity checklist** for signaling relay rooms and railway communication networks.
 12. **Railway Communication Network Topology Design**
Design a **network topology for a modern railway section** using LTE-R / 5G-R communication including base stations and control centers.
-

Texts Books:

1. **S. R. Vaidyanathan**, *Railway Signalling and Interlocking Systems*, Technical Publications, Pune.
 2. **U. Kumar and V. K. Verma**, *Modern Railway Signalling Systems*, McGraw Hill Education, India.
 3. **CAMTECH / RDSO**, *Communication Based Train Control (CBTC) – Introductory Handbook*, Indian Railways Publications.
 4. **A. H. Jazwinski**, *Railway Safety, Reliability and Risk Assessment*, CRC Press, Taylor & Francis Group.
 5. **H. H. Nguyen and Ed Shwedyk**, *A First Course in Digital Communications*, Cambridge University Press.
 6. **William Stallings**, *Data and Computer Communications*, Pearson Education.
-

Reference Books:

1. **J. D. Gibson**, *The Communications Handbook*, CRC Press.
2. **K. Sam Shanmugam**, *Digital and Analog Communication Systems*, Wiley India.
3. **RDSO (Research Design and Standards Organization)**, *Indian Railway Signal Engineering Manual*.
4. **International Union of Railways (UIC)**, *Railway Telecommunications and Signalling Standards*.
5. **IRISSET Technical Journal**, *Grade of Operations and Driverless Train Operation in Railways*.



Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Third Year B.Tech (Electronics and Computer Engineering)

SEMESTER-VI

Honors in Renewable Energy and Power Electronics

ECEHON-04D AI in Renewable Energy Optimization

Teaching Scheme:

Lecture: 3Hrs/week, 3 credits

Practical : 2Hrs/week, 1 credit

Examination Scheme:

ESE : 70 Marks

ISE : 30 Marks

ICA : 25 Marks

Introduction:

This course introduces the application of artificial intelligence and machine learning techniques in renewable energy systems to address challenges in forecasting, optimization, and energy management, enables students to analyze renewable energy data and develop AI-based models for grid integration, smart grids, and energy storage, and exposes learners to real-world case studies and future trends in sustainable and intelligent energy systems.

Prerequisite:

Students should have basic knowledge of: Basics of Renewable Energy Systems (Solar, Wind, Storage) Fundamentals of Power Systems, Basic knowledge of Mathematics and Probability, Introduction to Programming / Data Analysis, Basic understanding of Control Systems (desirable)

Course Objectives:

The course aims to:

1. To introduce artificial intelligence concepts relevant to energy systems.
2. To apply AI techniques for optimization of renewable energy generation and utilization.
3. To understand forecasting methods for renewable energy resources.
4. To develop AI-based solutions for grid integration and energy management.
5. To expose students to real-world applications and case studies in smart and sustainable energy systems.

Course Outcomes:

At the end of the course students will be able to

1. Explain the role of AI and machine learning in renewable energy systems.
2. Apply AI techniques for forecasting renewable energy generation.
3. Analyze optimization problems in renewable energy systems using AI tools.
4. Develop AI-based models for energy management and grid integration.
5. Evaluate real-world renewable energy optimization case studies using AI.
6. Identify future trends and challenges in AI-driven sustainable energy systems.

SECTION-I

Unit-1: Introduction to AI and Renewable Energy Systems

(07)

Overview of renewable energy systems: solar, wind, hybrid systems, Challenges in renewable energy integration and optimization, Introduction to Artificial Intelligence and Machine Learning, AI vs conventional optimization techniques, Applications of AI in renewable energy systems

Unit-2: Data Analytics for Renewable Energy**(07)**

Renewable energy data types and sources, Data acquisition from sensors, smart meters and IoT devices
Data pre-processing, normalization and feature extraction, Statistical analysis and visualization of energy data, Introduction to energy datasets and benchmarks

Unit-3: Machine Learning Techniques for Energy Forecasting**(07)**

Supervised and unsupervised learning concepts, Solar irradiance and PV power forecasting, Wind speed and wind power prediction, Regression, decision trees and support vector machines, Performance evaluation metrics for forecasting models

SECTION-II**Unit-4: Optimization of Renewable Energy Systems Using AI****(07)**

Optimization problems in renewable energy systems, Genetic Algorithms (GA) for energy optimization

Particle Swarm Optimization (PSO), Fuzzy logic and neural networks for system optimization, Hybrid AI techniques for renewable energy applications

Unit-5: AI in Smart Grids and Energy Management**(07)**

AI-based energy management systems, Load forecasting and demand response using AI, Optimization of energy storage systems, AI applications in micro grids and distributed generation, Fault detection and predictive maintenance using AI

Unit-6: Advanced Applications and Case Studies**(07)**

Deep learning applications in renewable energy, Reinforcement learning for energy scheduling, AI-based optimization of hybrid renewable systems, Case studies of AI implementation in smart grids, Future trends: AI, digital twins and autonomous energy systems

Internal Continuous Assessment (ICA):

ICA consists of a minimum **eight** experiments based upon the above curriculum.

List of Experiments:

1. Study of renewable energy systems and identification of optimization challenges using real-world scenarios.
2. Collection and analysis of renewable energy datasets (solar irradiance, wind speed, load demand).
3. Data pre-processing techniques for renewable energy data: cleaning, normalization and feature selection.
4. Exploratory data analysis and visualization of renewable energy datasets.
5. Implementation of linear and nonlinear regression models for solar power prediction.
6. Wind speed and wind power forecasting using machine learning techniques.
7. Performance comparison of different forecasting models using error metrics (MAE, RMSE, MAPE).
8. Implementation of artificial neural networks (ANN) for renewable energy output prediction.
9. Application of genetic algorithm (GA) for optimization of renewable energy system parameters.
10. Implementation of particle swarm optimization (PSO) for optimal energy scheduling.
11. Fault detection and predictive maintenance of renewable energy systems using machine learning.
12. Case study analysis of AI applications in smart grids and renewable energy optimization.
13. Mini-project: Design and simulation of an AI-based optimization model for a hybrid renewable energy system

Text Books:

1. Artificial Intelligence Applications in Renewable Energy Systems – Elsevier
 2. Machine Learning for Energy Systems – Springer
 3. Smart Grids and Renewable Energy Systems – CRC Press
-

Reference Books:

1. IEEE Transactions on Smart Grid
2. IEEE Transactions on Sustainable Energy
3. NPTEL Courses on AI and Renewable Energy.