**R24**

## SCHEME OF INSTRUCTION & EXAMINATION
## B.E. V- Semester
## (ELECTRONICS AND COMMUNICATION ENGINEERING)

| S. No | Course Code | Course Title | Scheme of Instruction | | | | Scheme of Examination | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P/D | Contact Hrs/Wk | CIE | SEE | Duration in Hrs | |
| **Theory Course** | | | | | | | | | | |
| 1 | PC 501 EC | Microcontroller & Interfacing | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 2 | PC 502 EC | Analog Communication | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 3 | PC 503 EC | Digital System Design Using Verilog | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 4 | PC 504 EC | Automatic Control Systems | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 5 | PC 505 EC | IoT System Design And Applications | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 6 | | Professional Elective-I | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| **Practical / Laboratory Course** | | | | | | | | | | |
| 7 | PC 551EC | Microcontroller & Interfacing Lab | - | - | 2 | 2 | 25 | 50 | 3 | 1 |
| 8 | PC 552EC | Digital System Design Using Verilog Lab | - | - | 2 | 2 | 25 | 50 | 3 | 1 |
| 9 | SD 553EC | Skill Development Course-1 | - | - | 2 | 2 | 50 | - | - | 1 |
| **Total** | | | **18** | **0** | **6** | **24** | **280** | **520** | **24** | **21** |

| Professional Elective-I | | |
|---|---|---|
| **S.No.** | **Course Code** | **Course Title** |
| 1 | PE511EC | Artificial Neural Networks |
| 2 | PE512EC | Data Science with Applications in ECE |
| 3 | PE513EC | Python Programming and Applications |
| 4 | PE514EC | Industry 4.0 and Smart Electronic Systems |

**R24**

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| **PC 501 EC** | **MICROCONTROLLER & INTERFACING** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | | | |
| **Prerequisite** | **L** | **T** | **D** | **P** | **CIE** | **SEE** | **Credits** |
| **PC 402 EC** | **3** | **-** | **-** | **-** | **30** | **70** | **3** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Understand the architecture, addressing modes and programming of 8086 microprocessor.
2. Learn the architecture, pin diagram & memory organization of 8051 microcontroller.
3. Understand the 8051-instruction set and Addressing modes.
4. Describe the programming of Timers/Counters, UART and interrupts using 8051 MC.
5. Develop interfacing of real time devices like DAC, ADC, LCD and stepper motor with 8051.

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

1. Describe the architecture of 8086 microprocessor, instruction set and recognize different types of addressing modes.
2. Analyze the architecture of 8051 microcontroller and memory organization of 8051 MC.
3. Write assembly language programming using 8051 microcontrollers.
4. Program Timers/Counters, UART and interrupts using 8051 microcontrollers.
5. Interface different peripheral modules to 8051 microcontrollers.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO 11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 1 | 1 | 1 | 2 | 1 | - | - | - | - | - | - |
| **CO2** | 2 | 1 | 1 | 1 | - | - | - | - | - | - | - | - |
| **CO3** | 2 | 1 | 1 | 1 | 2 | 2 | - | - | - | - | - | - |
| **CO4** | 2 | 1 | 2 | 1 | 2 | 2 | - | - | 2 | - | 2 | 2 |
| **CO5** | 2 | 1 | 2 | 1 | 2 | 2 | 1 | - | 2 | - | 2 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**8086 Microprocessor:** Intel 8086/8088 architecture, Segmented memory, Minimum and Maximum modes of operation, addressing modes, Instruction set, Assembly Language Programming (ALP) using data transfer, arithmetic, logical, branching & string manipulation instructions.

**UNIT II**

**8051 Microcontroller: 8051** Internal architecture and pin configuration, Differentiation

| |
|---|
| between 8086 MP & 8051 MC, 8051 features.<br>**8051 Memory Organization**: On-chip memory, Internal RAM, Registers, bit addressable memory, Special function Registers (SFRs), Register banks, Program Status Word (PSW). |
| **UNIT III** |
| **8051 Instructions & Programming**: 8051 addressing modes, instruction set, I/O port structures, Instruction formats, Assembly Language Programming (ALP) using data transfer, arithmetic, logical and branch instructions. |
| **UNIT IV** |
| **8051 Timers, Serial Port & Interrupts:** 8051 Timers/Counters and its programming, Serial data communication, Serial port and its programming, 8051 interrupts, Interrupt Vector Table (IVT), Interrupts programming. |
| **UNIT V** |
| **8051 Interfacing:** Interfacing of 8051 with Digital-Analog Convertor (DAC), Analog-Digital Convertor (ADC), Liquid Crystal Display (LCD), Stepper Motor & External Memory. |

| **Suggested Text Books:** |
|---|
| 1. Ray A.K and Bhurchandi K.M , Advanced Microprocessors and Peripherals ‖,3/e,Tata McGraw Hill Education Pvt Ltd, 2013. |
| 2. MazidiM.A,Mazidi J.G and RolinD.Mckinlay , The 8051 Microcontroller & Embedded Systems Using Assembly and C ‖,2/e, Pearson Education, 2008. |
| 3. DouglasV. Hall,Microprocessors and Interfacing Programming and Hardware ‖, 2nd Edition, Tata McGraw Hill publishing company Limited, New Delhi, 2008. |
| **Reference Text Books:** |
| 1. Ayala K.J,The 8051 Microcontroller Architecture, programming & Applications ‖, Penram International, 2007. |
| 2. Scott Mackenzie and Raphael C. W. Phan., The 8051 Microcontroller‖, 4th Edition, Pearson education, 2008. |

| **Course Code** | **Course Title** | | | | | | **Core/PE/OE** |
|---|---|---|---|---|---|---|---|
| **PC 502 EC** | **ANALOG COMMUNICATION** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC 301 EC** | **3** | | | **3** | **30** | **70** | **3** |

**Course Objectives :** The course is taught with the objectives of enabling the student to:

1. To understand the fundamental concepts of communication systems and the need for modulation.
2. To study amplitude modulation techniques and their generation and detection methods.
3. To learn angle modulation concepts including frequency and phase modulation and their practical implementations.
4. To analyze pulse modulation techniques, both analog and digital, used in modern communication systems.
5. To evaluate the impact of noise on communication systems and assess system performance using SNR and related parameters.

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

1. Analyze amplitude modulation techniques, including AM, DSB-SC, SSB, and the operation of transmitters and superheterodyne receivers.
2. Apply the concepts of frequency and phase modulation, including their generation, detection, and bandwidth analysis.
3. Understand and explain the principles of sampling and pulse analog modulation techniques such as PAM, PWM, and PPM.
4. Analyze pulse digital modulation schemes such as PCM, DPCM, DM, and ADM, and compare their performance.
5. Evaluate the effects of noise in communication systems and analyze signal-to-noise ratio and noise performance in AM systems.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO 11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 2 | | - | 1 | 2 | 1 | - | - | - | - |
| **CO2** | 3 | 2 | 2 | | - | - | - | 1 | - | - | - | - |
| **CO3** | 3 | 2 | - | | - | - | - | 1 | - | - | - | - |
| **CO4** | 3 | 3 | 2 | | - | - | - | 1 | - | - | - | - |
| **CO5** | 3 | 2 | - | 2 | - | 1 | 1 | - | - | - | - | - |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| **UNIT I** |
|---|
| **Amplitude Modulation:** Need for modulation, Amplitude modulation, Time and Frequency Domain representation of AM (DSB-FC), AM modulation (Square law modulator), AM demodulation (Envelope Detector), Power relations in AM, Generation of DSB-SC signal |

| |
|---|
| (Balanced Modulator), DSB-SC demodulation (Coherent Detector, COSTAS loop).<br>**SSB Modulation:** Definition of SSB, Time and Frequency domain representation, Vestigial Sideband (VSB) concept, AM Transmitter and receiver (basic block diagram), Superheterodyne Receiver, Receiver Characteristics (definitions of Sensitivity, Selectivity, Fidelity). |
| **UNIT II** |
| **Angle Modulation:** Definition and basic concepts of Frequency Modulation (FM) and Phase Modulation (PM), Single tone FM, Spectrum analysis of sinusoidal FM wave, Narrowband Frequency Modulation (NBFM), Wideband Frequency Modulation (WBFM), Constant average power in FM systems, Transmission bandwidth of FM wave (Carson's Rule), Generation of FM – Direct method (Reactance Modulator) and Indirect method (Armstrong Modulator), Detection of FM Phase Locked Loop (PLL) Detector, Pre-Emphasis and De Emphasis |
| **UNIT III** |
| **Pulse Analog Modulation schemes:** Sampling theorem, types of sampling, Types of Pulse Analog Modulation Schemes- Pulse Amplitude Modulation (PAM), Pulse Width Modulation (PWM), Pulse Position Modulation (PPM) -Qualitative treatment only. |
| **UNIT IV** |
| **Pulse Digital Modulation Schemes:** Quantization, Analog to Digital Conversion, PCM, Companding in PCM – mu law, A law. DPCM, DM and ADM. Comparison of PCM, DPCM, DM and ADM. |
| **UNIT V** |
| **Noise:** Definition, Sources of noise, Atmospheric noise, thermal noise, shot noise, Noise in two-port network: noise figure, equivalent noise temperature of Single and cascade stages, noise equivalent bandwidth.<br>**Noise in Analog Communication Systems:** Signal to Noise Ratio (SNR) and Figure of merit calculations in AM. |

| |
|---|
| ***Suggested Text Books:*** |
| 1. Simon Haykin, "Communication Systems," 2nd edition, Wiley India, 2011. |
| 2. H. Taub, D.L. Schilling, "Principles of communication systems",Tata McGraw Hill, 2001. |
| 3. B.P. Lathi, Zhi Ding, "Modern Digital and Analog Communication Systems", 4th edition, Oxford University Press, 2016. |
| ***Reference Text Books:*** |
| 1. Leon W Couch II., "Digital and Analog Communication Systems",6th edition, Pearson Education Inc., 2001. |
| 2. P. Ramakrishna Rao, "Analog Communication," 1st edition, TMH, 2011. |

| Course Code | Course Title | Core/PE/OE |
|---|---|---|

| **PC503EC** | **DIGITAL SYSTEM DESIGN USING VERILOG** | | | | | | **Core** |
|---|---|---|---|---|---|---|---|
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC302EC** | **3** | **0** | **0** | **0** | **30** | **70** | **3** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Understand the basic concepts of digital design and fundamentals of Verilog HDL.
2. Apply dataflow and switch level modeling techniques to design combinational circuits.
3. Develop behavioral models using procedural constructs in Verilog.
4. To understand Register Transfer Level (RTL) design and FSM implementation.
5. Implement digital systems using FPGA/ASIC design flow and real-time examples.

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

1. Describe basic concepts of Verilog HDL and digital design.
2. Design combinational circuits using dataflow and switch level modeling.
3. Develop behavioral models using procedural constructs.
4. Design and simulate sequential circuits using RTL.
5. Implement digital systems using FPGA/ASIC design examples.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | 2 | 1 | - | - | - | - | 1 | - | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | - |
| **CO3** | 3 | 3 | 2 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |
| **CO4** | 3 | 3 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| **CO5** | 3 | 3 | 2 | 3 | 2 | 1 | 1 | 2 | 1 | 1 | 2 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Introduction :** Overview of Digital Design with Verilog HDL, Structure of a Verilog module, Basic concepts, lexical conventions, data types (nets and registers), operators and expressions, system tasks and compiler directives, Gate Level Modelling, Gate types and delays.

**UNIT II**

**Dataflow Modelling:** Continuous assignment and delays, Adders and Subtractors, comparators, multiplexers and Demultiplexers, encoders and decoders, parity generator and checker circuits, Hierarchical modeling of arithmetic circuits, switch level modelling with examples, Design of stimulus block.

**UNIT III**

**Behavioral Modeling**: Behavioral modeling basics, procedural blocks (initial and always), blocking and non-blocking assignments, conditional statements, multi-way branching using case, loops, tasks and functions.

**UNIT IV**

**Register Transfer Level (RTL) design:** Modeling of latches and flip flops, shift registers and synchronous counters, Pattern detection, Mealy and Moore Verilog implementation of FSM based designs and simulation of sequential circuits.

| **UNIT V** |
|---|
| **FPGA/ASIC Design examples:** Introduction to ASIC/FPGA Design flow, design examples:4-bit adder-cum-Subtractor, Booth's multiplications, Restoring/Non-restoring division algorithms and 8 bit ALU. |

| **Suggested Text Books:** |
|---|
| 1. **Samir Palnitkar**, *Verilog HDL: A Guide to Digital Design and Synthesis*, Pearson Education, 2nd Edition, 2003. |
| 2. **Charles H. Roth Jr. and Larry L. Kinney**, *Digital Design Using Verilog HDL*, Cengage Learning, 2nd Edition, 2013. |
| 3. **Stephen Brown and Zvonko Vranesic**, *Fundamentals of Digital Logic with Verilog Design*, McGraw-Hill Education, 3rd Edition, 2014. |
| **Reference Text Books:** |
| 1. Michael John Sebastian Smith, *Application Specific Integrated Circuits*, 3/e, Pearson Education Asia, 2001. |
| 2. Ming-Bo Lin., Digital System Designs and Practices Using Verilog HDL and FPGAs. Wiley, 2008. |

| **Course Code** | **Course Title** | **Core/Elective** |
|---|---|---|

| PC 504 EC | AUTOMATIC CONTROL SYSTEMS | | | | | | Core |
|---|---|---|---|---|---|---|---|
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| PC 301 EC | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives :** The course is taught with the objectives of enabling the student to:

1. Introduce basic concepts of control systems, feedback effects, mathematical modeling, and system representation techniques.
2. Study time domain analysis of control systems using standard test signals and performance specifications.
3. Understand stability analysis of control systems using Routh–Hurwitz criterion and Root Locus techniques.
4. Introduce frequency response methods for analyzing control system performance and stability.
5. Understand state space representation and analysis of linear time invariant systems..

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

1. Develop the mathematical model of the physical systems and find the transfer function using different approaches.
2. Analyse time response of first and second order systems and evaluate steady state errors and controller effects.
3. Determine system stability and construct root locus plots for system analysis.
4. Analyse systems using Bode plots and Nyquist stability criterion.
5. Develop state models, obtain transfer functions, and analyze controllability and observability.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 2 | 1 | 2 | – | – | – | – | – | – | 1 |
| CO2 | 3 | 3 | 2 | 2 | 2 | – | – | – | – | – | – | 1 |
| CO3 | 3 | 3 | 2 | 2 | 2 | – | – | – | – | – | – | 1 |
| CO4 | 3 | 3 | 2 | 2 | 2 | – | – | – | – | – | – | 1 |
| CO5 | 3 | 2 | 2 | 2 | 2 | – | – | – | – | – | – | 1 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Control System Fundamentals**: Classification of control systems including Open and Closed loop systems, Effect of feedback on Control systems, Mathematical modeling of Mechanical systems and their conversion into electrical systems, Transfer function representation, Block diagram representation, Block diagram algebra and reduction and Signal flow graphs and Mason’s gain formula.

**UNIT II**

**Time Domain Analysis**: Standard test signals, Time response of first and second order systems with standard input signals, Time domain specifications, steady state error and error

| |
|---|
| constants. Effects of P, PI, PD and PID Controllers |
| **UNIT III** |
| Stability Analysis in S-Domain: Concept of stability, Routh Hurwitz criterion. Construction of Root locus. Effects of adding poles and zeros to open loop transfer function on the root loci. |
| **UNIT IV** |
| **Frequency Response Analysis**: Introduction to Frequency response of the system. Correlation between time and frequency responses, Frequency domain Specifications, Bode plots, Stability analysis, Nyquist plot and Nyquist criterion for stability**,** Compensation Networks: Lead, Lag, Lead-Lag |
| **UNIT V** |
| **State Space Representation**: Concept of state and state variables. State models of linear time invariant systems, Derivation of Transfer Function from State Model, State transition matrix, Solution of state equations. Controllability and Observability. |

| **Suggested Text Books:** |
|---|
| 1. Nagrath, I.J, and Gopal, M., "Control System Engineering", 5th edition, New Age Publishers, 2009 |
| 2. NagoorKani, "Control systems engineering", Oxford & IBH Publishing Company Private Limited, 2021. |
| 3. Ogata, "Modern Control Engineering", 2/e, Prentice Hall of India.,2011. |
| **Reference Text Books:** |
| 1. R.C. Sukla, "Control Systems", 3/e, Dhanpatrai and Sons,1998 |
| 2. Control Systems Engg. , Nise– John wiley , 3rd Edition 2000 |

| **Course Code** | **Course Title** | **Core/Elective** |
|---|---|---|

**R24**

| PE505EC | IOT SYSTEM DESIGN AND APPLICATIONS | | | | | | Core |
|---|---|---|---|---|---|---|---|
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| ES302CS | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives:**

1. To understand the fundamentals, architecture, and real-world applications of the Internet of Things (IoT).
2. To study IoT architecture, Machine-to-Machine (M2M) communication, and networking concepts used in IoT systems.
3. To learn IoT hardware platforms, embedded devices, sensors, actuators, and their interfacing techniques...
4. To understand IoT communication protocols such as MQTT, AMQP, CoAP, and mDNS for efficient device communication.
5. To develop the ability to design, implement, and evaluate basic IoT-based solutions for real-world applications.

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

1. **Understand** the fundamental concepts, architecture, enabling technologies, and applications of the Internet of Things..
2. **Explain** IoT communication models, M2M communication, and network management concepts used in IoT systems.
3. **Interface and program** IoT hardware platforms such as Arduino and Raspberry Pi with various sensors and actuators.
4. **Analyze and implement** IoT communication protocols including MQTT, AMQP, CoAP, and mDNS for IoT applications.
5. **Design and develop** simple IoT-based solutions by integrating hardware, communication protocols, and cloud-enabled technologies.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | - | - | 3 | - | - | - | 1 | 1 | - | 1 |
| CO2 | 3 | 3 | - | 1 | 3 | - | - | - | 2 | 1 | - | 1 |
| CO3 | 3 | 3 | 2 | 2 | 3 | - | - | - | 2 | 1 | - | 1 |
| CO4 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 1 |
| CO5 | 1 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Introduction to Internet of Things:** Introduction to Signals and Systems (Brief), Definition and Characteristics of Internet of Things (IoT), Sensors and Actuators, Physical Design of IoT, IoT Communication Models, IoT Communication APIs, IoT-enabled Technologies, Wireless Sensor Networks (WSN), Cloud Computing, Embedded Systems, IoT Levels and Deployment Templates, Domain-Specific IoT Applications including Smart Homes, Smart Cities, Smart Environment, Smart Energy, Smart Agriculture, and Smart Industry.

**UNIT II**

**IoT Architecture:** Introduction to IoT Architecture, IoT Reference Model, IoT Reference Architecture, Functional Blocks of IoT, Information Flow in IoT Systems, IoT Layers,

| |
|---|
| Deployment Models, Security and Privacy Considerations, and Real-World IoT Architecture**.** |
| **UNIT III** |
| **IoT Communication and M2M:** Introduction to Machine-to-Machine (M2M) Communication, M2M Architecture, IoT Communication Technologies, Software Defined Networks (SDN) – Overview, Network Function Virtualization (NFV) – Overview, Comparison of SDN and NFV, Simple Network Management Protocol (SNMP), IoT Device Management, Network Monitoring, and Applications of M2M in IoT. |
| **UNIT IV** |
| **IoT Physical Devices:** Introduction to Arduino, Raspberry Pi, Installation and Configuration, Serial Communication, SPI, $I^2C$ Interface, LED Interfacing, Buzzer, Relay Control, Servo Motor, DC Motor Speed Control, Stepper Motors, Sensor and Actuator Interfacing, and Development of IoT-Based Embedded Applications**.** |
| **UNIT V** |
| **IoT Protocols:**<br>Introduction to IoT Communication Protocols, MQTT, Quality of Service (QoS), MQTT Security, Advanced Message Queuing Protocol (AMQP), Constrained Application Protocol (CoAP), Multicast DNS (mDNS), Secure IoT Communication, and Applications of IoT Protocols. |

| |
|---|
| **Suggested Text Books:** |
| 1. **Arshdeep Bahga& Vijay Madisetti**, *Internet of Things – A Hands-on Approach*, Universities Press, 2015. |
| 2. **Raj Kamal**, *Internet of Things: Architecture and Design Principles*, McGraw Hill Education, First Edition, 2017. |
| 3. **Olivier Hersent**, **David Boswarthick**, and **Omar Elloumi**, *The Internet of Things: Key Applications and Protocols*, Wiley, Second Edition, 2012. |
| **Reference Text Books:** |
| 1. **Samuel Greengard**, *The Internet of Things*, MIT Press, 2015. |
| 2. **Daniel Minoli**, *Building the Internet of Things with IPv6 and MIPv6*, Wiley, 2013. |

| **Course Code** | **Course Title** | | | | | | **Core/PE/OE** |
|---|---|---|---|---|---|---|---|
| **PC 551 EC** | **MICROCONTROLLER & INTERFACING LAB** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC 501 EC** | **-** | **-** | **-** | **2** | **25** | **50** | **1** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Apply assembly language programs on 8086 trainer kit in stand-alone/serial mode.
2. Classify interface modules into input/output and memory interfaces with 8086.
3. Develop and execute the assembly & embedded C programming concepts of 8051 MC.
4. Design and develop 8051embedded C programs for various interface modules.
5. Develop Interface with serial and I2C bus.

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

1. Apply different addressing modes and model programs using 8086/8051 Instruction set.
2. Explain the usage of string instructions of 8086 for string manipulation and comparison.
3. Program Timers/Counters using 8051 microcontroller
4. Develop different programs using C-cross compilers for 8051 microcontroller.
5. Design & Develop different interfacing applications using 8051 microcontroller.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 2 | 3 | 3 | - | - | - | - | - | - | 2 |
| **CO2** | 3 | 3 | 3 | 2 | 3 | - | - | - | - | - | - | 2 |
| **CO3** | 3 | 3 | 2 | 3 | 2 | 3 | 3 | - | 2 | - | - | 2 |
| **CO4** | 2 | 3 | 2 | 3 | 3 | 3 | 3 | - | 2 | - | - | 2 |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 3 | 3 | - | 3 | - | - | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**PART A: Experiments for 8086 using any Compiler & appropriate hardware**

1. Use of 8086 trainer kit and execution of programs. (Instruction set for simple Programs using 4 to 5 lines of instruction code under different addressing modes for data transfer, manipulation & arithmetic operations)
2. Branching operations and logical operations in a given data.
3. Multiplication and division using memory locations.
4. Single byte, multi byte Binary and BCD addition & subtraction.
5. Code conversions.
6. String Searching and Sorting.

**PART B: Experiments for 8051 using any Cross Compiler & appropriate hardware**

1. Familiarity and use of 8051/8031 Microcontroller trainer and execution of

| | |
|---|---|
| | programs. |
| 2. | Instruction set for simple programs (using 4 to 5 lines of instruction code). |
| 3. | Interfacing 8051 with DAC to generate waveforms. |
| 4. | Program to control stepper motor using 8051. |
| 5. | Execution of programs in simulation software & generation of hex file. |
| 6. | ADC interfacing with 8051. |
| 7. | Timer and counter operations & programming using 8051. |
| 8. | Serial communications using UART. |

**Note:**

1. At least ten experiments to be conducted in the semester.
2. Minimum of 5 from Part A and Part B is compulsory.
3. In Part-A & B, perform the experiments using appropriate hardware kits & assembler simulators like 8086 Emulator, MASM/TASM, edsim51/Keil software etc.

| **Suggested Text Books:** |
|---|
| 1. DouglasV. Hall,―Microprocessors and Interfacing Programming and Hardware ‖, 2nd Edition, Tata McGraw – Hill publishing company Limited, New Delhi, 2008. |
| 2. MykePredko – *Programming and Customizing the 8051 Microcontroller,* TMH, 2005. |
| 3. Mazidi M.A, Mazidi J.G &Rolin D. Mckinlay, "The 8051 Microcontroller & Embedded Systems using Assembly and C," 2/e, Pearson Education, 2007. |

| **Course** | **Course Title** | **Core/PE/OE** |
|---|---|---|

| **Code** | | | | | | | |
|---|---|---|---|---|---|---|---|
| **PC552EC** | **DIGITAL SYSTEM DEISGN UISNG VERILOG HDL LAB** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC 504 EC** | **-** | **-** | **-** | **2** | **25** | **50** | **1** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. To provide hands-on experience in modeling complex digital systems using Verilog HDL at various levels of abstraction (Behavioral, Structural, and Dataflow).
2. To impart knowledge on designing efficient finite state machines (FSMs) for control-path applications, including Mealy and Moore architectures.
3. To facilitate understanding of the separation between Data path and Control units in digital system design for implementing arithmetic and logic operations.
4. To demonstrate the verification process of digital designs through the creation of test benches and the analysis of simulation waveforms.
5. To encourage independent design thinking through a mini-project that integrates multiple sub-modules into a functional hardware system.

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

1. Model and simulate combinational and sequential circuits such as encoders, parity generators, and universal shift registers using Verilog HDL.
2. Design and implement synchronous and asynchronous counters to meet specific timing and sequence requirements in digital systems.
3. Develop FSM-based Controllers for real-world applications, such as traffic light systems and vending machines, by deriving state diagrams and transition tables.
4. Synthesize Data path Components and integrate them with controllers to perform complex operations like serial addition and ALU functions.
5. Evaluate and document a complete digital system (e.g., Simple CPU or UART) by managing the full design cycle from architectural planning to simulation-based testing.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | - | 3 | - | - | - | - | - | - | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | - | - | 1 |
| **CO3** | 3 | 3 | 3 | 2 | 2 | 1 | - | - | - | - | - | 2 |
| **CO4** | 3 | 3 | 3 | 2 | 3 | - | - | - | - | - | - | 2 |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 2 | 3 | 2 | 3 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| **Part-A: List of Experiments** |
|---|

| |
|---|
| 1. Basic Arithmetic Circuits: Write structural and dataflow Verilog HDL models for:<br>• 4-bit Ripple Carry Adder<br>• 4-bit Adder–Subtractor |
| 2. Advanced Adders: Write Verilog HDL models for:<br>• 4-bit Carry Look-Ahead Adder<br>• 2-digit BCD Adder |
| 3. Comparator Circuits: Write Verilog HDL programs for:<br>• 4-bit Comparator<br>• 8-bit comparator using two 4-bit comparators and additional logic (Hierarchical structural model) |
| 4. Multiplexers and Decoders: Write a Verilog HDL program in behavioral model for:<br>• 8:1 Multiplexer<br>Write a Verilog HDL program in hierarchical structural model for:<br>• 16:1 Multiplexer using 4:1 multiplexer<br>• 3:8 Decoder using 2:4 decoder |
| 5. Encoder and Parity Circuits: Write a Verilog HDL program in behavioral model for:<br>• 8:3 Encoder<br>• 8-bit Parity Generator and Checker |
| 6. Counters: Write Verilog HDL programs in structural/behavioral models for:<br>• 8-bit Asynchronous Up–Down Counter<br>• 8-bit Synchronous Up–Down Counter |
| 7. Shift Registers and Multipliers: Write a Verilog HDL program for:<br>• 8-bit Universal Shift Register (Structural Model)<br>• 8-bit Shift-and-Add Multiplier (Behavioral Model) |
| 8. Finite State Machines: Write a Verilog HDL program for:<br>• 4-bit Sequence Detector using Mealy and Moore State Machines |
| 9. Controller Design using FSM: Write Verilog HDL programs for:<br>• Traffic Light Controller using FSM<br>• Vending Machine Controller using FSM |
| 10. Datapath and Controller Design: Write Verilog HDL programs for implementation of Datapath and controller units:<br>• Serial Adder<br>• Arithmetic Logic Unit (ALU) |

| **Part B – Mini Project / Design Exercise: The student must design, develop Verilog code, simulate and test ONE of the following systems:** |
|---|
| 1. 8-bit Simple CPU |
| 2. Booth's Algorithm for signed binary number multiplication |
| 3. MAC Unit or DSP module (FIR/IIR Filter) |
| 4. Synchronous and Asynchronous Data Transfer: UART with Baud Rate Generator |

| |
|---|
| 5. Design of 4-bit Thermometer to Binary Code Converter |

| **Suggested Text Books:** |
|---|
| 1. Samir Palnitkar, Verilog HDL: A Guide to Digital Design and Synthesis, 2nd Edition, Pearson Education. |
| 2. Michael D. Ciletti, Advanced Digital Design with the Verilog HDL, PHI. |

| **Course Code** | **Course Title** | **Core/PE/OE** |
|---|---|---|
| **SD553EC** | **SKILL DEVELOPMENT COURSE-I** | **Core** |

| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
|---|---|---|---|---|---|---|---|
| | L | T | D | P | | | |
| - | - | - | - | 2 | 50 | - | 1 |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Design electronic schematics and PCB layouts using appropriate EDA software tools.
2. Develop skills in PCB layout design, routing, and Gerber file generation using EDA tools.
3. Understand the basics of IoT systems and Interface sensors and actuators with microcontroller-based IoT platforms.
4. Develop IoT applications using wireless communication and cloud platforms.
5. Apply PCB design and IoT concepts to solve real-world engineering problems.

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

1. Design electronic schematics using EDA tools.
2. Create PCB layouts and generate Gerber files.
3. Interface sensors and actuators with IoT development boards.
4. Develop basic IoT applications using wireless communication and cloud platforms.
5. Design projects integrating PCB and IoT technologies.

**Module I: PCB DESIGNING ( 6 weeks)**

This module provides hands-on training in Printed Circuit Board (PCB) design using industry-standard Electronic Design Automation (EDA) tools.

Students will learn the complete PCB development workflow, including schematic creation, component library utilization, PCB layout design, component placement, routing strategies, design rule verification, and generation of manufacturing files such as Gerber files.

The module emphasizes practical implementation of electronic circuit designs and equips students with the skills required to develop professional, fabrication-ready PCB layouts.

**Module II: IoT applications ( 6 weeks)**

This module offers practical exposure to the design and development of Internet of Things (IoT) applications.

Students will explore IoT architectures, microcontroller-based development platforms, sensor and actuator interfacing, wireless communication technologies, cloud-based data management, and remote monitoring and control systems. Through hands-on activities, students will develop and implement IoT solutions for real-world applications involving automation, environmental monitoring, smart devices, and connected systems.

The module focuses on integrating hardware, software, and communication technologies to build efficient and scalable IoT applications.

**Evaluation Criteria (50 Marks)**

| Criterion | Marks |
|---|---|
| Continuous Assessment (Active participation/Progress Reviews) | 10 |
| Practical Skills and Task Completion | 10 |
| Project / Application Development / Hands-on Activity | 10 |
| Report | 10 |
| Presentation | 10 |
| **Total** | 50 |

***Suggested Text Books:***

1. Simon Monk, **"Make Your Own PCBs with EAGLE",** McGraw-Hill Education.
2. Chris Robertson, **"PCB Designer's Reference",** Prentice Hall.
3. Arshdeep Bahga and Vijay Madisetti, **"Internet of Things: A Hands-On Approach"**, Universities Press Limited, 2010.
4. Raj Kamal, **"Internet of Things: Architecture and Design Principles",** McGraw-Hill Education.

***Open-Source software for PCB Design/ IoT Design:***

1. **KiCad** – Open-source PCB design suite for schematic capture and PCB layout.
2. **EasyEDA** – Free web-based PCB design platform.
3. **Arduino IDE** – Open-source development environment for Arduino and ESP boards.
4. **Tinkercad Circuits** – Free online platform for Arduino circuit simulation and prototyping.
5. **Node-RED** – Open-source flow-based IoT programming tool.
6. **ESP-IDF** – Official development framework for ESP32.
7. **ThingSpeak** – Cloud platform for IoT data visualization and analytics.
8. **Mosquitto MQTT Broker** – Open-source MQTT communication platform.

**R24**

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| **PE511EC** | **ARTIFICIAL NEURAL NETWORKS** | | | | | | **Elective** |
| **Prerequisite** | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| | **L** | **T** | **D** | **P** | | | |
| **-** | **3** | **-** | **-** | **-** | **30** | **70** | **3** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Understand the fundamental concepts and evolution of ANNs and their biological inspiration.
2. Learn various neural network models, architectures, and activation functions along with basic learning laws.
3. Analyze different learning algorithms including supervised, unsupervised, and reinforcement learning techniques.
4. Study advanced neural network models such as multilayer perceptrons, backpropagation networks, and associative memories.
5. Explore feedback networks and self-organizing systems like Hopfield networks, RBF networks, Kohonen maps, and Boltzmann machines.

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

1. Describe the basic concepts, architectures, and learning rules of artificial neural networks.
2. Apply various learning algorithms to design and implement simple neural network models.
3. Analyze and design associative memory models and pattern recognition systems.
4. Implement and evaluate multilayer neural networks and backpropagation algorithms for solving complex problems like XOR.
5. Analyze and apply advanced neural network models such as Hopfield networks, RBF networks, Kohonen maps, and Boltzmann machines in real-world applications.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | - | 1 | - | - | - | - | - | - | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | - | - | 2 |
| **CO3** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | - | - | 2 |
| **CO4** | 3 | 2 | 3 | 2 | 2 | - | - | - | 1 | - | - | 2 |
| **CO5** | 3 | 2 | 3 | 1 | 2 | - | - | - | 1 | - | 1 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT – I**

Fundamental concepts, Evolution of Neural Networks, Artificial Neural Networks: Description of biological neuron, Basic model, Classification, Feed forward and Recurrent topologies, Neural network activation functions, Basic learning laws: Hebb's law, Perceptron law, delta rule, Widrow& Hoff rule, LMS rule, Correlation rule, Winner takes-all rule and In-star &out-star learning law.

**UNIT – II**

Neural Network learning algorithms: Supervised, Un-supervised & Reinforcement learning,Fundamentals of connectionist modeling: McCulloach–Pits model, Perceptron model, Adaptive Linear model &Multiple AdaLine model, Implementation of basic logic gates with different neuron models.

| |
|---|
| **UNIT – III** |
| Different Neural Network Models, Memories & their applications: Pattern association, Pattern storage (LTM & STM), Pattern clustering and feature map, Neural Network memories: Hetro associative, Auto associative, Bidirectional associative memory: Topology, learning algorithm & Applications and Interpolative Neural Network. |
| **UNIT – IV** |
| Feed forward Neural Networks, multi-layer Neural Network with linear and non-linear Processing units. Multi-layer perceptron:Topology,learning algorithm, Applications and limitations, Perceptron Neural Networks solution of Xor problem, Perceptron convergence theorem, Back propagation Neural Networks: Learning rule, Features of Back propagation, limitations of Back propagation Neural Network and extensions of Back Propagation model. |
| **UNIT – V** |
| Feedback Neural Networks, Hopfield network, capacity and energy analysis of Hopfield Neural Network,Radial Basis Function networks: Topology, learning algorithm &Applications, Kohenen's self-organising network: Topology, learning algorithm & Applications, Stochastic neuron, Boltzman's machine, Boltzman's learning law, Issues in Implementation of Boltzman's learning law. |

| | |
|---|---|
| **Suggested Text Books:** | |
| 1. | B. Yeganaranarana, Artificial Neural Networks, Prentice Hall, New Delhi, 2007. |
| 2. | J.A.Freeman and D.M.Skapura, Neural Networks Algorithms, Applications and Programming Techniques, Addison Wesley, New York, 1999. |
| 3. | Simon Haykin, Neural Networks (A Comprehensive Foundation), McMillan College Publishing Company, New York, 1994. |
| | **Reference Text Books:** |
| 1. | **S. N. Sivanandam and S. N. Deepa**, *Principles of Soft Computing*, Wiley India, 3rd Edition, 2018. |
| 2. | **Laurene Fausett**, *Fundamentals of Neural Networks: Architectures, Algorithms and Applications*, Pearson Education, 1st Edition, 1994. |

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| PE512EC | DATA SCIENCE WITH APPLICATIONS IN ECE | | | | | | Elective |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| - | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Introduce the fundamentals of data science and its applications in Electronics and Communication Engineering.
2. Understand data collection, storage, preprocessing, and visualization techniques for engineering datasets.
3. Apply statistical methods and exploratory data analysis techniques for interpreting engineering data.
4. Familiarize students with hypothesis testing, machine learning concepts, and predictive analytics.
5. Develop the ability to apply supervised and unsupervised learning techniques to solve real-world ECE problems involving IoT, communication systems, sensor networks, and smart systems.

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

1. Understand the fundamentals of data science, data collection, storage, and applications in Electronics and Communication Engineering.
2. Apply statistical methods, data analytics, and visualization techniques for engineering datasets.
3. Perform data preprocessing, exploratory data analysis, and correlation analysis on real-world datasets.
4. Analyze engineering data using hypothesis testing, ANOVA, and basic machine learning techniques.
5. Apply supervised and unsupervised learning algorithms for solving problems in Electronics and Communication Engineering applications.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO 11 | PO 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | – | – | 2 | – | – | – | – | – | – | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 3 | – | – | – | – | 1 | – | 2 |
| **CO3** | 3 | 2 | 3 | 3 | – | – | – | 1 | 1 | – | 2 | 3 |
| **CO4** | 3 | 3 | 3 | 3 | – | – | – | – | – | – | 2 | 3 |
| **CO5** | 3 | 3 | 3 | 2 | 3 | 1 | 1 | – | 1 | 1 | – | 3 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| UNIT I |
|---|
| Introduction to Data Science: Introduction, Core Concepts and Technologies, Data Science Process and Toolkits, Applications of Data Science in Electronics and Communication Engineering. Data: Data and Information, Importance of Data, Types of Data, Data Sources,Data Collection Techniques, Data Storage. |
| **UNIT II** |
| **Data Analytics:** Importance of Data Analytics, Types of Data Analytics—Descriptive, Diagnostic, Predictive and Prescriptive. Statistics: Understanding Attributes, Types of Attributes—Discrete and Continuous Attributes. Measures of Central Tendency: Mean, Median, Mode. Measures of Dispersion: Range, Variance, Standard Deviation, Skewness and Kurtosis.<br>**Data Visualization:** Importance of Data Visualization, Principles of Effective Visualization,Line Charts, Bar Charts, Scatter Plots, Histograms, Box Plots, Pie Charts, Heat Maps,Correlation Matrix. Introduction to Visualization Tools: Matplotlib, Seaborn and Plotly. |
| **UNIT III** |
| **Data Cleaning and Preparation:** Exploring Data, Data Cleaning, Handling Missing Data, Data Integration, Data Transformation, Data Normalization and Standardization, Discretization and Binning. Outliers: Introduction, Detecting and Filtering Outliers.<br>**Exploratory Data Analysis:** Univariate, Bivariate and Multivariate Analysis. Correlation Analysis: Correlation Coefficients,Covariance, Feature Selection.<br>**Sampling:** Random and Non-Random Sampling. Case Studies: Exploratory Analysis using Engineering Datasets (IoT Sensor Data, Human Activity Recognition Data, Smart Home Energy<br>**Data and Wireless Sensor Network Data). Data Sources**: Kaggle, UCI Machine Learning Repository and IEEE DataPort. |
| **UNIT IV** |
| **Hypothesis Testing:** Introduction, Developing Hypothesis, Null and Alternate Hypothesis,Type-I Error, Type-II Error, p-value, Confidence Interval, Approaches to Hypothesis Testing.<br>**Analysis of Variance (ANOVA):** Introduction to ANOVA, One-Way ANOVA, Two-WayANOVA.<br>**Introduction to Machine Learning:** Artificial Intelligence, Machine Learning, DeepLearning, Supervised Learning and Unsupervised Learning. Case Studies: Statistical Inference and Hypothesis Testing using Engineering Datasets (Sensor Measurements, Communication System Performance, Network Traffic and IoT Device Data).<br>**Data Sources:** Kaggle, UCI Machine Learning Repository, IEEE DataPort and Open Power System Data. |
| **UNIT V** |

**Supervised Learning:** Regression—Simple Linear Regression and Multiple Linear Regression.
**Case Study:** Signal Strength Prediction, Sensor Calibration or Energy Consumption Prediction.
**Classification:** Logistic Regression, CART.Case Study: Fault Detection in Electronic Devices,Communication Network Traffic Classification or Predictive Maintenance.
**Unsupervised Learning:** Clustering—K-Means Clustering. Case Study: IoT Device Clustering, Wireless Sensor Network Node Clustering or Smart Meter Consumer Segmentation.
**Data Sources:** Kaggle, UCI Machine Learning Repository, IEEE DataPort, Open Power System Data and NASA Prognostics Data Repository.

| **Suggested Text Books:** |
|---|
| 1. Chirag Shah, A Hands-on Introduction to Data Science, Cambridge University Press. |
| 2. Avinash Navlani, Armando Fandango, Ivan Idris, Python Data Analysis: Perform Data Collection, Data Processing, Wrangling, Visualization and Model Building Using Python, 3rd Edition. |
| 3. Suresh Kumar Mukhiya, Usman Ahmed, Hands-On Exploratory Data Analysis with Python, Packt Publishing. |
| **Reference Text Books:** |
| 1. Wes McKinney, Python for Data Analysis, 3rd Edition, O'Reilly Media. |
| 2. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, O'Reilly Media. |

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| PE513EC | PYTHON PRO GRAMMING AND APPLICATIONS | | | | | | Elective |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| ES302CS | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives :**

1. Introduce the fundamentals of Python programming, including syntax, data types, control structures, functions, and modules.
2. Develop programming skills using Python data structures and standard libraries for solving engineering problems.
3. Apply Python packages such as NumPy, SciPy, Matplotlib, and Lcapy for circuit analysis, data visualization, and signal processing.
4. Understand advanced Python concepts including file handling, exception handling, object-oriented programming, database connectivity, GUI programming, and multithreading.
5. Enable students to develop Python-based applications for real-world Electronics and Communication Engineering problems.

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

1. Build basic programs using fundamental programming constructs like variables, conditional logic, looping, and functions
2. Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
3. Create, run and manipulate Python Programs using core data structures like Lists, dictionaries and use Regular Expressions
4. Develop programs in Python for implementation of non-linear circuits and analyze filters.
5. Use file handling, exception handling, OOP, databases, and Tkinter to develop simple Python applications like a Student Management System.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | - | - | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO2** | 3 | 3 | - | 1 | 3 | - | - | - | 2 | 1 | - | 1 |
| **CO3** | 3 | 3 | 2 | 2 | 3 | - | - | - | 2 | 1 | - | 1 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 1 |
| **CO5** | 1 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Introduction to Python:** History of Python, Need of Python programming, Features of

| |
|---|
| python. **Python basics:** Tokens, working with data types and variables, working with numeric data, working with string data, Boolean expressions, selection structure, iteration structure. **Python Functions:** default values of arguments, named arguments, local and global variables. |
| **UNIT II**<br>**Modules:** creating, documenting and Importing modules, Use of standard modules.**Data structures:** Lists: basic lists, creating and processing list of lists, Tuples, Dictionaries,Implementation of stacks and sets,working onSequences- reversing, permuting, sorting. |
| **UNIT III**<br>**Python Installation and Packages:** Introduction to PIP, installing and uninstalling packages via PIP. Python Packages: **Numpy:** Creating arrays from lists Array Operations, Linear Algebra. **Matplotlib:** Introduction Basic Plotting Types of Graphs Advanced Plotting. **Scipy:** Introduction,SciPy Modules, Numerical Integration, Optimization, Statistical Functions. **Circuit analysis:** Circuit representation, processing of components, Introduction to Nodes, Branches and Loops, Loop and Nodal analysis.<br>**Case study:** Model circuits and perform nodal analysis and loop analysis using Lcapy (opensource) Python package for solving linear circuits using matrix operations |
| **UNIT IV**<br>**Signal Analysis:** Representation of Continuous time signals, Discrete time signals, PythonImplementation of sampling, Fourier Transform, Laplace transform, Z-transform, DiscreteFourier Transform, Fast Fourier transform, Design of LTI filters, FIR filters and IIR filters usingPython.<br>**Case study:** Cleaning Up Data Noise with Fourier Transform using Python |
| **UNIT V**<br>**Advanced Python Programming**: **File Handling:** Reading and writing files, file modes, and basic file operations. **Exception Handling:** Introduction, try, except, else, and finally blocks. **Object-Oriented Programming (OOP):** Introduction, classes and objects, self argument, __init__() method, class methods, class and object variables, public and private members, inheritance, and operator overloading. **Python Database Programming:** Introduction, database connectivity, creating tables, inserting, updating, deleting, and retrieving records. **GUI Programming using Tkinter:** Introduction, basic widgets, and event handling. Multithreading: Introduction, thread creation, and basic applications.<br>**Case Study:** Development of a Student Management System using Python for file handling, database operations, and basic GUI to store, update, search, and manage student records. |

| ***Suggested Text Books:*** |
|---|
| 1. Michael Urban and Joel Murach, ―Python Programming‖, Mike Murach& Associates, Incorporated, 2016 |
| 2. KalilurRahman, ―Python Data Visualization Essentials Guide‖, BPB publications,2021. |
| 3. Shivkumar V. Iyer , ― Simulating Nonlinear Circuits with Python Power Electronics-An Open-Source Simulator, Based on Python, Springer International Publishing, 2018. |
| **Reference Text Books:** |
| 1. Thomas Haslwanter, ―Hands-on Signal Analysis with Python: An |

Introduction‖,Springer International Publishing, 2021.

2. Vijay Madisetti and ArshdeepBahga, 'Internet of Things (A Hands-on-Approach)', 1 st Edition, VPT, 2014.

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| PE514EC | INDUSTRY 4.0 AND SMART ELECTRONIC SYSTEMS | | | | | | Elective |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| PC 402 EC | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. **Understand** the concepts, technologies, and evolution of Industry 4.0 and Smart Manufacturing.
2. **Explain** the role of Embedded Systems, Industrial IoT, sensors, and communication technologies in industrial automation.
3. **Analyze** industrial automation systems, communication protocols, and smart manufacturing architectures.
4. **Explore** Artificial Intelligence, Machine Learning, Data Analytics, Cloud Computing, and Cybersecurity for Industry 4.0 applications.
5. **Identify** emerging Industry 4.0 applications and evaluate future technologies and career opportunities in smart industries.

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

1. Explain the concepts, architecture, enabling technologies, and applications of Industry 4.0 and Smart Manufacturing.
2. Describe the role of Embedded Systems, Industrial IoT (IIoT), sensors, actuators, and edge/cloud computing in industrial environments.
3. Analyze industrial automation systems, PLC/SCADA architectures, industrial communication protocols, and smart manufacturing networks.
4. Apply AI, Machine Learning, Data Analytics, Cloud Computing, and Cybersecurity concepts to Industry 4.0 applications.
5. Evaluate emerging Industry 4.0 technologies and propose suitable smart solutions for real-world industrial applications.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO 11 | PO 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | 1 | 2 | – | 2 | – | – | 1 | – | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 3 | – | 1 | – | 1 | 1 | – | 2 |
| **CO3** | 3 | 3 | 3 | 2 | 3 | – | 2 | – | 1 | 1 | 1 | 2 |
| **CO4** | 3 | 3 | 2 | 3 | 3 | 1 | 2 | 2 | – | 1 | – | 3 |
| **CO5** | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 2 | 2 | 2 | 2 | 3 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Introduction to Industry 4.0 and Smart Manufacturing:** Evolution of Industrial

| |
|---|
| Revolutions (Industry 1.0 to Industry 4.0), Smart Manufacturing, Digital Transformation, Cyber Physical Systems (CPS), Smart Factories, Industrial Internet of Things (IIoT), Industry 4.0 ecosystem, Opportunities and challenges in Industry 4.0. |
| **UNIT II** |
| **Embedded Systems and Industrial IoT:** Role of Embedded Systems in Industry 4.0, Sensors and Actuators, IoT Architecture, Industrial IoT (IIoT), Wireless Sensor Networks, Arduino and Raspberry Pi Platforms, Edge Computing, Cloud-based Smart Monitoring Systems. |
| **UNIT III** |
| **Industrial Automation and Communication:** Industrial Automation Fundamentals, PLC and SCADA Systems, Machine-to-Machine (M2M) Communication, Industrial Communication Protocols, MQTT, Zigbee, Wi-Fi and Bluetooth, Robotics in Smart Manufacturing, Predictive Maintenance. |
| **UNIT IV** |
| **Artificial Intelligence and Data Analytics:** Introduction to Artificial Intelligence (AI) and Machine Learning (ML), AI Applications in Smart Manufacturing, Data Analytics for Industrial Systems, Big Data Concepts, Digital Twin Technology, Industrial Cloud Computing, Intelligent Decision Support Systems. |
| **UNIT V** |
| **Cybersecurity and Emerging Industrial Technologies:** Fog Computing in IIoT, Cybersecurity in Industry 4.0, Data Privacy and Secure Industrial Communication, Applications of Industry 4.0 in Smart Grids, Healthcare, Autonomous Vehicles, EV Electronics, Semiconductor Industry, 5G-enabled Industrial Systems. |

| |
|---|
| **Suggested Text Books:** |
| 1. Alasdair Gilchrist, Industry 4.0: The Industrial Internet of Things, Apress, 2016 |
| 2. Vijay Madisetti and Arshdeep Bahga, Internet of Things: A Hands-On Approach, Universities Press, 2014. |
| 3. Raj Kamal, Internet of Things: Architecture and Design Principles, McGraw Hill Education, 2017. |
| **Reference Text Books:** |
| 1. Frank Lamb, Industrial Automation: Hands-On, McGraw Hill Education, 2013. |
| 2. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson, 2022. |

## SCHEME OF INSTRUCTION & EXAMINATION
## B.E. VI- Semester
## (ELECTRONICS AND COMMUNICATION ENGINEERING)

| S. No | Course Code | Course Title | Scheme of Instruction | | | | Scheme of Examination | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P/D | Contact Hrs/Wk | CIE | SEE | Duration in Hrs | |
| **Theory Course** | | | | | | | | | | |
| 1 | PC601EC | Digital Communication | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 2 | PC602EC | VLSI Design | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 3 | PC603EC | Antenna and Wave Propagation | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 4 | PC 604EC | Embedded systems | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 5 | PC 605EC | Digital Image and Video Processing | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| 6 | | Professional Elective-II | 3 | - | - | 3 | 30 | 70 | 3 | 3 |
| **Practical / Laboratory Course** | | | | | | | | | | |
| 7 | PC651EC | Communication Systems Lab | - | - | 2 | 2 | 25 | 50 | 3 | 1 |
| 8 | PC652EC | VLSI Design Lab | - | - | 2 | 2 | 25 | 50 | 3 | 1 |
| 9 | PC 653EC | Embedded systems& IOT Applications Lab | - | - | 2 | 2 | 25 | 50 | 3 | 1 |
| 10 | PW654EC | Mini Project | - | - | 4 | 4 | 25 | 50 | 3 | 2 |
| 11 | *PW701EC | *Summer Internship | Minimum 4 weeks of internship during summer vacation | | | | | | | |
| **Total** | | | **18** | **0** | **10** | **28** | **280** | **620** | **30** | **23** |

**Note:*** Students have to undergo summer internship of minimum 4- Weeks duration at the end of semester VI and Evaluation will be done in VII - Semester

| Professional Elective-II | | |
|---|---|---|
| **S. No.** | **Course Code** | **Course Title** |
| 1 | PE621EC | Real Time Operating Systems |

| | | |
|---|---|---|
| 2 | PE622EC | FPGA Design and Applications |
| 3 | PE623EC | Satellite Communication |
| 4 | PE624EC | Linux and scripting languages |

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| **PC601EC** | **DIGITAL COMMUNICATION** | | | | | | **Core** |
| **Prerequisite** | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| | **L** | **T** | **D** | **P** | | | |
| **PC502EC** | **3** | **-** | **-** | **-** | **30** | **70** | **3** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. To introduce the concepts of optimum receiver, baseband digital data transmission and
2. To analyze the error performance of different digital carrier modulation schemes like ASK, FSK, PSK etc.
3. To familiarize with the concepts of information theory and basic source coding techniques.
4. To understand the principles and techniques of channel coding.
5. To familiarize the students with the concepts of spread spectrum communication with emphasis on DSSS and FHSS.

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

1. Understand the elements of digital communication and analyse baseband transmission and optimum receiver.
2. Describe digital carrier modulation schemes and M-ary signalling concepts.
3. Explain Shannon–Fano and Huffman coding and assess the capacity of communication channels.
4. Distinguish different types of Error control codes along with their encoding/decoding algorithms.
5. Understand generation of PN sequence and analyze the performance of Spread Spectrum communication systems.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 3 | 1 | 2 | 2 | - | - | - | - | - | - | 1 |
| **CO2** | 3 | 2 | 1 | 1 | 2 | - | - | - | - | - | - | 1 |
| **CO3** | 3 | 3 | 1 | 2 | 1 | - | - | - | - | - | - | 1 |
| **CO4** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | - | - | 1 |
| **CO5** | 3 | 2 | 1 | 1 | 1 | - | - | - | - | - | - | 1 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Introduction to Digital Communication:** Elements of Digital Communication System**,** Comparison of Digital and Analog Communication Systems.

**Base band digital data transmission –** Block diagram, Inter Symbol Interference, Nyquist

| |
|---|
| criterion for Zero ISI, Eye pattern.<br>**Optimum Receiver**: Optimum filter and matched filter, and evaluation of probability of error evaluations. |
| **UNIT II** |
| **Digital Carrier Modulation Schemes** — Description and generation of ASK, FSK, PSK. Signal Constellation, Coherent detection of Binary ASK, FSK, PSK. DPSK. Comparison of digital carrier modulation schemes.<br>**M-ary signaling schemes:** Introduction, QPSK- generation and detection, Signal Constellation, Synchronization methods. |
| **UNIT III** |
| Information Theory and Source Coding: Uncertainty, Information, entropy, information rate. Discrete memory less channel – Probability relations in a channel, priori & posteriori entropies, Joint entropy, conditional entropy, mutual information, Channel capacity - Binary Symmetric Channel, Binary Erasure Channel, cascaded channels, Shannon-Hartley Theorem – Shannon Bound.<br>Source coding: Shannon – Fano and Huffman coding. |
| **UNIT IV** |
| **Channel Coding:** Introduction to error correcting codes, types of transmission errors, need for error control coding.<br>**Linear Block Codes (LBC):** Matrix description of LBC, generation, Syndrome calculation and error detection, Minimum distance of Linear block code, error detection and error correction capabilities.<br>**Binary cyclic codes (BCC):** Polynomials, Algebraic description of cyclic codes, systematic encoding using generator polynomial and parity check polynomial, syndrome calculation, decoding and error correction using shift registers.<br>**Convolution codes:** Encoding, Decoding using code tree, state diagram. |
| **UNIT V** |
| **Spread Spectrum Communication:** Advantages of Spread Spectrum, generation and characteristics of PN sequences. Direct sequence spread spectrum and Frequency hopping spread spectrum systems. CDMA, ranging using DSSS. Acquisition and Tracking of DSSS and FHSS signals. |

| **Suggested Text Books:** |
|---|
| 1. B. P. Lathi and Zhi Ding, *Modern Digital and Analog Communication Systems*, 4th Edition, Oxford University Press, 2009. |
| 2. Leon W. Couch II, *Digital and Analog Communication Systems*, 6th Edition, Pearson, 2001. |
| 3. K. Sam Shanmugam, *Digital and Analog Communication Systems*, Wiley, 1979. |
| **Reference Text Books:** |
| 3. Simon Haykin, *Digital Communication*, 4th Edition, Wiley India, 2011. |

4. Herbert Taub and Donald L. Schilling, *Principles of Communication Systems*, 2nd Edition, Tata McGraw-Hill, 2001.

| **Course Code** | **Course Title** | | | | | | **Core/PE/OE** |
|---|---|---|---|---|---|---|---|
| **PC602EC** | **VLSI DESIGN** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC302EC** | **3** | **1** | **0** | **0** | **30** | **70** | **4** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. To introduce MOS/CMOS device physics and fabrication technologies for understanding modern IC design fundamentals
2. To develop skills in VLSI physical design including stick diagrams, layout rules, and delay analysis.
3. To analyze CMOS inverter performance across static, dynamic, power, and scaling perspectives.
4. To design high-performance VLSI subsystems including adders, multipliers, memories, and sequential circuits.
5. To impart knowledge of VLSI testing methodologies for fault detection and testability enhancement.

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

1. Describe MOS transistor characteristics, CMOS fabrication processes, and inverter design principles (Understanding L2).
2. Apply VLSI design rules to create stick diagrams and layouts for NMOS/CMOS circuits with accurate RC delay calculations (Applying L3).
3. Analyze static/dynamic behavior, power consumption, and technology scaling effects on CMOS inverter performance (Analyzing L4).
4. Design advanced VLSI subsystems including optimized adders, multipliers, SRAM/DRAM cells, and sequential logic (Creating L6).
5. Evaluate fault models and generate test patterns for combinational circuits using path sensitization and Boolean difference methods (Evaluating L5)

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | 2 | 1 | | | | | 1 | | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | |
| **CO3** | 3 | 3 | 2 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |
| **CO4** | 3 | 3 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| **CO5** | 3 | 3 | 2 | 3 | 2 | 1 | 1 | 2 | 1 | 1 | 2 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| **UNIT I** |
|---|
| **Introduction:** Introduction to IC Technology – MOS, PMOS, NMOS, CMOS Fabrication Process. **Basic Electrical Properties**: Basic Electrical Properties of MOS: Ids- Vds relationships, MOS transistor threshold Voltage, gm, gds, figure of merit; Pass transistor, NMOS Inverter, Various pull ups. Introduction to FinFET and its Challenges |
| **UNIT II** |
| **VLSI Circuit Design Processes:** VLSI Design Flow, MOS Layers, Stick Diagrams, Design Rules and Layout, and Transistors Layout Diagrams for NMOS and CMOS Inverters and Gates. Basic circuit concepts, Sheet Resistance RS and its concept to MOS, Area Capacitance Units, Calculations – RC Delays. |
| **UNIT III** |
| **The CMOS Inverter:** Introduction, the static CMOS inverter, Evaluating the robustness of the CMOS inverter – static behavior, Performance of CMOS inverter – dynamic behavior, Power, energy and energy-delay, Technology scaling and its impact on the inverter metrics. |
| **UNIT IV** |
| **Subsystem Design**: Shifters, Carry skip adder, carry select adder , Booth Multiplier, Memory Elements: 6T SRAM cell, 1T DRAM cell. Sequential Logic Design:CMOS D latch and Edge triggered Flip flops. |
| **UNIT V** |
| **VLSI Testing:** Role of testing, types of testing, functionality tests, manufacturing tests, stuck-at faults, short circuit and open circuit faults, controllability , observability, delay fault testing, level sensitive scan design, Boundary scan architecture. |

| **Suggested Text Books:** |
|---|
| 1. Kamran Eshraghian, Douglas A. Pucknell, Essentials of VLSI Circuits and Systems, PHI, 2005 Edition. |
| 2. Jean M. Rabey, Digital Integrated Circuits, Prentice Hall India, 2003. |
| 3. CMOS VLSI Design: A Circuits and Systems Perspective – Neil H. E. Weste, David Harris.. |
| **Reference Text Books:** |
| 1. Wayne Wolf, Modern VLSI Design, Pearson Education, 3rd Edition, 1997. |
| 2. P Weste Kamran Eshraghian, Principles of CMOS VLSI Design - A Systems Perspective, by NEILHE, Pearson Education Series, Asia, 2002. |

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| PC603EC | ANTENNAS AND WAVE PROPAGATION | | | | | | Core |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| PC 405 EC | 3 | - | - | - | 30 | 70 | 3 |

| |
|---|
| **Course Objectives :** The course is taught with the objectives of enabling the student to: |
| 1. Understand the basic principles of antennas and introduce the antenna terminologies. |
| 2. Analyzing the radiation characteristics of thin wire Dipole antenna, Monopole antenna and Loop antenna. |
| 3. Explore antenna array requirements and techniques for measurement of antenna parameters. |
| 4. Characterization of UHF & VHF antennas and Microstrip antenna |
| 5. To explain the various modes of radio wave propagation. |
| **Course Outcomes :** On completion of this course, the student will be able to : |
| 1. Illustrate the basic principles of antennas and learn the antenna terminology. |
| 2. Analyze the radiation characteristics of different wire antennas, and describe the introduction to CST Studio. |
| 3. Apply the principles of antenna arrays and measure standard antenna parameters. |
| 4. Evaluate the behaviour and performance of VHF, UHF and Microwave antennas for various ranges of frequencies and get updated with latest developments in the smart antennas. |
| 5. Understand and compare various modes of radio wave propagation used for different applications. |

| CO \ PO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | – | – | – | – | – | – | – | – | – | 1 |
| **CO2** | 3 | 3 | 1 | 1 | 2 | – | – | – | – | – | – | 1 |
| **CO3** | 3 | 2 | 2 | 2 | 2 | – | – | – | – | – | – | 1 |
| **CO4** | 3 | 3 | 2 | 1 | 2 | – | – | – | – | – | – | 2 |
| **CO5** | 3 | 2 | – | 1 | – | – | – | – | – | – | – | 1 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| UNIT I |
|---|
| **Antenna Fundamentals:** Introduction, principles of radiation-single wire, two wire, current distribution on a thin wire antenna, isotropic radiator, Antenna parameters: Radiation pattern, Beam area, beam efficiency, radiation intensity, directivity, gain, radiation efficiency, effective aperture, effective length, Antenna impedance, antenna field zones, antenna polarization , and antenna temperature, Friis transmission equation, Retarded potentials |
| **UNIT II** |
| Analysis of Linear Wire Antennas :Introduction, Radiation from Infinitesimal/Short dipole or an Alternating Current Element, Half-wave dipole and quarter wave monopole, Loop antennas-small loop, far field pattern of small Circular Loop with uniform current, Comparison of far fields of small loop and short dipole, , Introduction to Antenna Simulation using CST studio |
| **UNIT III** |
| **Antenna Arrays**: Array of point sources, two element arrays with equal and unequal amplitudes, different phases, linear n-element array with uniform distribution, Broadside and End fire arrays, Derivation of their characteristics and comparison, Principle of Pattern Multiplication. Binomial Array. Antenna Measurements: Introduction, Antenna Test Site and Measurement Setup, Gain Measurement Techniques (Comparison Method, Absolute Method, and Three-Antenna Method), Sources of Errors. |
| **UNIT IV** |
| **VHF,UHF and Microwave Antennas:** Helical antenna - Geometry, Helix modes, design considerations for Helical antenna , Yagi-Uda Antenna, Horn antenna, Parabolic Reflector Antennas, Micro Strip Antennas- Basic characteristics, feeding Methods, Smart Antennas-Basic concepts of smart antennas and types of smart antennas. |
| **UNIT V** |
| **Wave Propagation:** Ground, Space and Surface waves. Troposphere refraction and reflection, Duct propagation, Sky wave propagation, Regular and irregular variations in ionosphere Line of sight propagation. |

| Suggested Text Books: |
|---|
| 1. J. D. Kraus, R. J. Marhefka, and Ahmad S. Khan, “Antennas and Wave Propagation”, McGraw-Hill, 5th Edition, 2017. |
| 2. Constantine A. Balanis, “Antenna Theory: Analysis and Design”, John Wiley, 4th Edition ,2015. |
| 3. K.D. Prasad, “Antenna and wave Propagation “, Satya Prakashan Publishing Company, 2009 |
| **Reference Text Books:** |
| 1. Edward C. Jordan and Keith G. Balmain, “Electromagnetic Waves and Radiating |

| | |
|---|---|
| | Systems", 2nd Edition, PHI, 1968. |
| 2. | I.J. Bahl and P.Bhartia , "Microstrip Antennas" , Artech House, 1980 |

| Course Code | Course Title | | | | | | Core/PE/OE |
|---|---|---|---|---|---|---|---|
| PC604EC | EMBEDDED SYSTEMS | | | | | | Core |
| | Contact Hours per Week | | | | CIE | SEE | Credits |
| Prerequisite | L | T | D | P | | | |
| PC409EC | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Understand the basic concepts and characteristics of embedded systems.
2. Study the architecture and organization of embedded processors.
3. Learn embedded software development using Embedded C and assembly programming.
4. Understand communication interfaces and protocols used in embedded systems.
5. Study design methodologies and applications of embedded systems in real-world scenarios.

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

1. Understand the architecture and characteristics of embedded systems
2. Analyze the working of embedded processors and microcontrollers
3. Develop embedded software using Embedded C and development tools.
4. Implement communication protocols used in embedded and IoT systems.
5. Study design methodologies and applications of embedded systems in real-world scenarios.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | - | - | 1 | - | - | - | - | 1 | - | 2 |
| **CO2** | 3 | 3 | 2 | - | 2 | - | - | - | - | 1 | - | 2 |
| **CO3** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | 1 | - | 2 |
| **CO4** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | 1 | - | 2 |
| **CO5** | 3 | 2 | - | 2 | 2 | - | - | - | - | 1 | - | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| UNIT I |
|---|
| **Introduction to Embedded Systems** Characteristics and classification of embedded systems, Embedded systems vs general purpose systems, Hardware and software components, Processor embedded into a system, Embedded system design process. System-on-Chip (SoC) architecture, Hardware–software co-design, Design metrics and challenges. Embedded platforms: Arduino, Raspberry Pi, ESP32, STM32. |
| **UNIT II** |
| RISC and CISC architecture concepts, ARM architecture fundamentals, ARM register organization, Current Program Status Register (CPSR), pipeline architecture. Exception handling, interrupts and vector tables. ARM processor families: ARM7, ARM9, Cortex-M, Cortex-A, Cortex-R processors. Introduction to instruction set architecture. |
| **UNIT III** |

| |
|---|
| Embedded Software and RTOS: Embedded software development process, Embedded C programming, Cross compiler, assembler, linker and loader. Debugging techniques and development tools. Real Time Operating Systems (RTOS): task management, scheduling, inter-process communication, semaphores and mutex. Examples: Free RTOS, VxWorks. |
| **UNIT IV** |
| **Serial communication concepts:** UART, SPI, I2C, CAN protocol, USB communication. Wireless communication technologies: Bluetooth, Wi-Fi, Zigbee. Introduction to IoT communication protocols such as MQTT and HTTP. |
| **UNIT V** |
| Embedded System Design and Applications: Hardware–software co-design, low power design techniques, embedded system testing and debugging, security concepts. Applications of embedded systems in automotive systems, consumer electronics, industrial automation, medical devices, smart home and IoT systems. |

| **Suggested Text Books:** |
|---|
| 1.Raj Kamal – Embedded Systems: Architecture, Programming and Design, McGraw Hill. |
| 2. Frank Vahid and Tony Givargis – Embedded System Design, Wiley |
| 3. **Jonathan W. Valvano**, *Embedded Systems: Introduction to ARM Cortex-M Microcontrollers*, CreateSpace Independent Publishing Platform, 5th Edition, 2019. |
| **Reference Text Books:** |
| 1. Shibu K V – Introduction to Embedded Systems, McGraw Hill. |
| 2. David E. Simon – An Embedded Software Primer, Pearson. |

**R24**

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| **PC605EC** | **Digital Image and Video Processing** | | | | | | **Elective** |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| **PC405EC** | **3** | **-** | **-** | **-** | **30** | **70** | **3** |

**Course Objectives**

1. Study fundamental concepts of Image Processing and various Image Transforms
2. Learn Image Enhancement Techniques in Spatial and Frequency domain, Image Segmentation methods
3. Familiarize with fundamentals of Image compression, Lossy & Lossless Compression methods.
4. Define concepts of Video Processing, Image Formation models, and processing of Video signals.
5. Understand general methodologies of 2 D Motion Estimation and Video coding methods.

**Course Outcomes**

After completing this course, the student will be able to:

1. Use different transforms for various applications like Image Enhancement ,Compression etc..
2. Use Spatial and Transform techniques to Enhance the given image and to extract the features of the image.
3. Use Lossless and Lossy compression techniques for real time applications.
4. Understand the fundamental concepts of Video capturing and Three Dimensional Motion Models.
5. Understand and analyze various Motion estimation techniques.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | P012 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | 2 | 2 | – | – | – | – | – | – | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 3 | – | – | – | – | – | – | 2 |
| **CO3** | 3 | 3 | 2 | 3 | 3 | – | – | – | – | – | – | 2 |
| **CO4** | 3 | 2 | 1 | 2 | 2 | – | – | – | – | – | – | 2 |
| **CO5** | 3 | 3 | 3 | 3 | 3 | – | – | – | – | – | – | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT I**

**Fundamentals of Image Processing and Image Transforms:** Basic steps of Image Processing System, Sampling and Quantization of an image, relationship between pixels
**Image Transforms:** 2 D- Discrete Fourier Transform, Discrete Cosine Transform (DCT), Wavelet **Transforms:** Continuous Wavelet Transform, Discrete Wavelet Transforms.

**UNIT II**

**Image Processing Techniques: Image Enhancement:** Spatial domain methods: Histogram processing, Fundamentals of Spatial filtering, Smoothing spatial filters, Sharpening spatial filters. **Frequency domain methods:** Basics of filtering in frequency domain, image smoothing, image sharpening, Selective filtering. Laplacian of Gaussian (LOG) filters.
**Image Segmentation:** Segmentation concepts, Point, Line and Edge Detection, Thresholding, Region Based segmentation, Hough Transform, Boundary detection, chain coding.

| |
|---|
| **UNIT III** |
| **Image Compression:** Image compression fundamentals - Coding Redundancy, Spatial and Temporal redundancy, Compression models: Lossy & Lossless, Huffman coding, Arithmetic coding, LZW coding, Run length coding, Bit plane coding, Transform coding, Predictive coding, Wavelet coding, JPEG Standards. |
| **UNIT IV** |
| **Basic concepts of Video Processing:** Analog Video, Digital Video. Time-Varying Image Formation models: Three-Dimensional Motion Models, Geometric Image Formation, Photometric Image Formation, Sampling of Video signals, Filtering operations. |
| **UNIT V** |
| **2-D Motion Estimation:** Optical flow, General Methodologies, Pixel Based Motion Estimation, Block- Matching Algorithm, Mesh based Motion Estimation, Global Motion Estimation, Region based Motion Estimation, Multi resolution motion estimation, Waveform based coding, Block based transform coding, Predictive coding, Application of motion estimation in Video coding, constant dependent video coding and joint shape and texture coding .MPEG and H.26X standards. |


| |
|---|
| **Suggested Text Books:** |
| 1. Gonzaleze and Woods, *Digital Image Processing*, 3rd edition, Pearson. |
| 2. Yao Wang, Joem Ostermann , Ya–quin Zhang, *Video processing and communication*, 1st Edition, PH Int |
| 3. S.Jayaraman, S.Esakkirajan, T.Veera Kumar *Digital Image Processing*, TMH, 2009. |
| **Reference Text Books:** |
| 1. M. Tekalp, *Digital Video Processing*, Prentice Hall International |
| 2. John Woods, *Multi-dimensional Signal, Image and Video Processing and Coding* 2nd Edition, Elsevier. |

| Course Code | Course Title | | | | | | Core/PE/OE |
|---|---|---|---|---|---|---|---|
| PC 651EC | COMMUNICATION SYSTEMS LAB | | | | | | Core |
| | Contact Hours per Week | | | | CIE | SEE | Credits |
| Prerequisite | L | T | D | P | | | |
| PC502EC,PC601EC | - | - | - | 2 | 25 | 50 | 1 |

**Course Objectives :** The course is taught with the objectives of enabling the student to:

1. To demonstrate AM, FM, mixer, and multiplexing techniques.
2. To understand pulse modulation techniques such as PAM, PWM, and PPM
3. To understand and simulate digital modulation (i.e., ASK, FSK, BPSK, QPSK) generation.
4. To model analog and pulse modulation techniques using CAD tools.
5. To model PCM, Delta modulation, and digital modulation techniques using CAD tools.

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

1. Understand and simulate modulation and demodulation of AM and FM.
2. Construct and understand the need for pre-emphasis and de-emphasis at the transmitter and receiver respectively
3. Simulate the PAM, PWM & PPM circuits.
4. Understand generation and detection of baseband transmission (i.e., PCM, DM, and ADM) and bandpass transmission (i.e., ASK,FSK,PSK, MSK and QPSK)
5. Understand the error control coding.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 2 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO2** | 2 | 2 | 3 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO3** | 2 | 2 | 3 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO4** | 3 | 2 | 2 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO5** | 2 | 2 | 1 | 1 | 2 | - | - | - | - | - | - | 1 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| List of Experiments |
|---|
| **PART-A** |
| 1. Amplitude Modulation and Demodulation. |
| 2. Frequency Modulation and Demodulation.. |
| 3. Pre-emphasis and De-emphasis and plot the frequency response |
| 4. Multiplexing Techniques (FDM and TDM). |
| 5. Mixer Characteristics and plot the frequency response |
| 6. Verification of Sampling Theorem. |
| 7. PWM, PPM generation and detection. |
| 8. Generation and Detection of AM, FM, PAM, PWM, PPM modulation techniques using |

| | |
|---|---|
| | MATLAB/Simulink/Lab-view |
| **PART – B** | |
| 1. | Digital Communication |
| 2. | PCM modulation and demodulation. |
| 3. | Channel encoding and decoding. |
| 4. | Linear and Adaptive Delta Modulation and Demodulation. |
| 5. | ASK generation and Detection. |
| 6. | FSK and Minimum Shift Keying generation and Detection. |
| 7. | ASK generation and Detection. |
| 8. | Generation and Detection of PCM, Delta modulation and Digital modulation schemes (ASK, FSK, |

| **Suggested Text Books:** |
|---|
| 1. Communication systems by V. Chandra Sekar, SASTRA University, Oxford University Press, 2013, ISBN: 9780198078050 |
| 2. Digital Communication Systems Using MATLAB and Simulink, Second Edition by Dennis Silage |

| Course Code | Course Title | | | | | | Core/PE/OE |
|---|---|---|---|---|---|---|---|
| **PC 652EC** | **VLSI DESIGN LAB** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC602EC** | **-** | **-** | **-** | **2** | **25** | **50** | **1** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Understand the fundamentals of digital logic circuits and basic logic gates.
2. Design and analyze combinational circuits such as adders, multiplexers, encoders, and decoders.
3. Develop skills in designing arithmetic and logic units (ALU).
4. Analyze the characteristics and behavior of MOSFET and CMOS circuits.
5. Gain practical knowledge in analog circuit design and VLSI layout techniques.

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

1. Implement and verify basic digital logic circuits and combinational designs.
2. Design and analyze adders, multiplexers, encoders, decoders, and ALU circuits.
3. Evaluate MOSFET characteristics and CMOS inverter performance parameters.
4. Design and analyze analog circuits such as amplifiers and operational amplifiers.
5. Develop VLSI design skills, including ring oscillators and layout design of basic circuits.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 2 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO2** | 2 | 2 | 3 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO3** | 2 | 2 | 3 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO4** | 3 | 2 | 2 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO5** | 2 | 2 | 1 | 1 | 2 | - | - | - | - | - | - | 1 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**List of Experiments**

**PART-A**

1. Experiment on Digital Circuits All basic gates Not, AND, NOR NAND OR Ex-OR Ex-NOR
2. Experiment on Digital Circuits Half Adder and Full Adder.
3. Experiment on Digital Circuits Multiplexer and De-Mux.
4. Experiment on Digital Circuits Encoder and Decoder.

| |
|---|
| 5. Experiment on Digital Circuits ALU Design |
| **PART – B** |
| 1. Characteristics of MOSFET. |
| 2. Calculation of rise time and fall time for CMOS inverter. |
| 3. To build a three stage and five stage ring oscillator circuit in 0.18um and 0.13um technology and compare its frequencies and time period. |
| 4. NMOS Common Source Amplifier. |
| 5. Design of Differential Amplifier. |
| 6. Design of Operational Amplifier. |
| 7. Draw the layout of Inverter Circuit. |

| **Suggested Text Books:** |
|---|
| 1. Kamran Eshraghian, Douglas A. Pucknell, Essentials of VLSI Circuits and Systems, PHI, 2005 Edition. |
| 2. Jean M. Rabey, Digital Integrated Circuits, Prentice Hall India, 2003. |

| **Course Code** | **Course Title** | | | | | | **Core/PE/OE** |
|---|---|---|---|---|---|---|---|
| **PC 653 EC** | **EMBEDDED SYSTEMS AND IOT APPLICATIONS LAB** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **PC604EC** | **-** | **-** | **-** | **2** | **25** | **50** | **1** |
| **Course Objectives:** The course is taught with the objectives of enabling the student to: | | | | | | | |
| 1. Familiarize with the usage of IDE tools and execution of programs using ARM processor | | | | | | | |
| 2. Understand the usage of various devices like LCD, Temperature sensor, Buzzer, Stepper Motor by interfacing them to LPC2148. | | | | | | | |
| 3. Study the designing and implementation of IoT applications using Arduino/RPi | | | | | | | |
| **Course Outcomes:** On completion of this course, the student will be able to: | | | | | | | |
| 1. Understand and use IDE toolsfor embedded system development. | | | | | | | |
| 2. Develop interfacing applicationsfor input/output devices using ARM processors. | | | | | | | |
| 3. Design embedded programs to read sensor data and display results | | | | | | | |
| 4. Develop IoT-based applicationsusing Arduino/Raspberry Pi platforms. | | | | | | | |
| 5. Apply RTOS concepts such as multitasking, scheduling, queues, and semaphores in embedded systems. | | | | | | | |

| **CO** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 3 | 2 | 2 | - | - | - | - | - | - | - |
| CO2 | 3 | 2 | 3 | 2 | 1 | - | - | - | - | - | - | - |
| CO3 | 3 | 2 | 3 | 2 | 2 | - | - | - | - | - | - | - |
| CO4 | 3 | 3 | 3 | 2 | 2 | - | - | - | - | - | - | - |
| CO5 | 2 | 3 | 2 | 3 | 3 | - | - | - | - | - | - | - |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| **PART-A** |
|---|
| **Interfacing Programs using embedded C on ARM Micro controller Kit** |
| 1. Program to interface 8-Bit LED and switch interface |
| 2. Program to implement Buzzer interface on IDE environment |
| 3. Program to display message in a 2 line x 16 characters LCD display |
| 4. Program to interface stepper motor Stepper motor and rotate in clockwise and anti lock wise direction |
| 5. Program to interface a Temperature sensor LM35 and read the values and display it |

| |
|---|
| 6. Program to demonstrate serial communication i.e to transmit from kit and receive from PC using serial port. |
| 7. Implementation of multitasking using two tasks (LED blinking with different delays) (RTOS) |
| 8. Study of RTOS scheduler and task switching |
| 9. Implementation of priority-based task scheduling |
| 10. Multitasking with queue and semaphore |
| **PART-B** |
| **Interfacing Programs using C / Python Programming Arduino / Raspberry Pi Kit for IoT Applications** |
| 1. Interface a Push button with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed |
| 2. Interface Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED at sensor detection. |
| 3. Interface LCD with Arduino/Raspberry Pi and write a program to display a message on it. |
| 4. Interface DHT11 sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings |
| 5. Interface motor using relay with Arduino/Raspberry Pi and write a program to turn ON motor when push button is pressed. |
| 6. Interface Bluetooth unit with Arduino/Raspberry Pi and write a program to turn LED ON/OFF when "1"/"0" is received from smart phone using Bluetooth. |
| 7. Program to upload temperature and humidity data to thing speak cloud using Arduino/Raspberry Pi |

| **Suggested Text Books:** |
|---|
| 1. Embedded Systems: Real-Time Interfacing to ARM Cortex-M Microcontrollers – Cengage Learning |
| 2. The 8051 Microcontroller and Embedded Systems using Assembly and C – Pearson |
| 3. Getting Started with Arduino – Maker Media |
| 4. Internet of Things: A Hands-On Approach – Universities Press |

| Course Code | Course Title | | | | | | Core/PE/OE |
|---|---|---|---|---|---|---|---|
| **PW654EC** | **MINI PROJECT** | | | | | | **Core** |
| | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| **Prerequisite** | **L** | **T** | **D** | **P** | | | |
| **-** | **-** | **-** | **-** | **4** | **25** | **50** | **2** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. To design a small electronic system for real-life applications.
2. To apply knowledge of components, ICs, and microcontrollers.
3. To develop basic hardware and limited software skills.
4. To learn PCB design and system implementation.
5. To improve project planning, testing, and documentation skills.

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

1. get practical experience of software design and development, and coding practices within Industrial/R&D Environments.
2. gain working practices within Industrial/R&D Environments
3. prepare reports and deliver effective presentation.
4. demonstrate effective written and oral communication skills
5. innovate in various engineering disciplines and nurture their entrepreneurial ideas.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 2 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO2** | 2 | 2 | 3 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO3** | 2 | 2 | 3 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO4** | 3 | 2 | 2 | 2 | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO5** | 2 | 2 | 1 | 1 | 2 | - | - | - | - | - | - | 1 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**Guidelines for Mini Project**

1. The mini-project is a team activity having maximum of 3 students in a team. This is electronic product design work with a focus on electronic circuit design.
2. The mini project may be a complete hardware or a combination of hardware and software. The software part in mini project should be less than 50% of the total work.
3. Mini Project should cater to a small system required in laboratory or real life.

| | |
|---|---|
| 4. | It should encompass components, devices, analog or digital ICs, micro controller with which functional familiarity is introduced. |
| 5. | After interactions with course coordinator and based on comprehensive literature survey/ need analysis, the student shall identify the title and define the aim and objectives of mini-project. |
| 6. | Student is expected to detail out specifications, methodology, resources required, critical issues involved in design and implementation and submit the proposal within first week of the semester. |
| 7. | The student is expected to exert on design, development and testing of the proposed work as per the schedule. |
| 8. | Art work and Layout should be made using CAD based PCB simulation software. Due considerations should be given for power requirement of the system, mechanical aspects for enclosure and control panel design. |
| 9. | Completed mini project and documentation in the form of mini project report is to be submitted at the end of semester. |
| 10. | The tutorial sessions should be used for discussion on standard practices used for electronic circuits/product design, converting the circuit design into a complete electronic product, PCB design using suitable simulation software, estimation of power budget analysis of the product, front panel design and mechanical aspects of the product, and guidelines for documentation /report writing. |

# Professional Elective -II

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| PE621EC | **Real Time Operating Systems** | | | | | | Elective |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| PC604EC | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives :** The course is taught with the objectives of enabling the student to:

1. Understand the fundamentals of RTOS.
2. Study the task management and Synchronization.
3. Uniprocessor scheduling algorithms.
4. To get acquainted with memory allocation strategies.
5. Explore the Unix kernel structure and its process management.

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

1. Differentiate OS, RTOS and classify Real-Time kernels.
2. Propose different inter process communication models opted in RTOS.
3. Demonstrate the use of different scheduling algorithms to estimate the deadline.
4. Allocate the memory for blocking and non-blocking functions.
5. Explore the features of UNIX operating system and differentiate between UNIX and POSIX.

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | - | - | 3 | - | - | - | 1 | 1 | - | 1 |
| **CO2** | 3 | 3 | - | 1 | 3 | - | - | - | 2 | 1 | - | 1 |
| **CO3** | 3 | 3 | 2 | 2 | 3 | - | - | - | 2 | 1 | - | 1 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 1 |
| **CO5** | 1 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**Unit-1**

**Real Time Operating Systems:** Brief History of OS, Defining RTOS, Differences between Traditional OS and RTOS; Architecture of RTOS, Kernels, Pre-emptive Vs Non-pre emptive kernels, Jobs and Process, release time, deadlines, and timing constraints, hard and soft timing

| |
|---|
| constraints, hard versus soft real time systems. |
| **Unit-II** |
| **RTOS Task and Task state**, Task Priorities, inter task Communication & Synchronization – Definition of Context Switching, Foreground ISRs and Background Tasks. Critical Section: Re-entrant Functions. |
| **Process Synchronisation**- Message queues, Mail boxes, pipes, Critical section, Semaphores, Classical synchronisation problem, Deadlocks. |
| **Unit-III** |
| **Scheduling**: Types of scheduling, scheduling algorithms: FCFS, SJF, Priority, Round Robin, UNIX Multi-level feedback queue scheduling, Thread scheduling, Multiprocessor scheduling concept, Real Time scheduling concept. Classical Uni-processor Scheduling Algorithms – RMS, Pre-emptive EDF scheduling – principle, comparisons. |
| **Unit-IV:** |
| **Memory Management:** Introduction, dynamic memory allocation in embedded systems, fixed-size memory management in embedded systems, blocking vs. non-blocking memory functions, hardware memory management units**.** Comparison of various RTOS like QNX, VX works, FreeRTOS. |
| **Unit-V:** |
| **UNIX Kernel**:File System, Concepts of –Process, Concurrent Execution & Interrupts, Process Management – forks & execution. Basic level Programming with System calls, Overview of Commands File I/O (open, create, close, lseek, read, write), Process Control (fork, vfork, exit, wait, waitpid, exec). POSIX Standards. |



| |
|---|
| **Suggested Text Books:** |
| 1. Raj Kamal, “Embedded Systems- Architecture, Programming and Design” Tata McGraw Hill, 2006. |
| 2. Singh, Rajib Mall. Real-Time Systems: Theory and Practice. Pearson Education India, 2006. |
| 3. Qing Li, and Caroline Yao. Real-Time Concepts for Embedded Systems. CMP Books, 2003. |
| **Reference Text Books:** |
| 1. Prasad, K. V. K. K. Embedded Real-Time Systems: Concepts, Design & Programming. Dream Tech Press, 2005 2. Herma K., |
| 2. Krishna.C.M, Kang, Shin.G, “Real Time Systems”, McGraw Hill, 1997. |

| **Course Code** | **Course Title** | | | | | | **Core/Elective** |
|---|---|---|---|---|---|---|---|
| **PE622EC** | **FPGA DESIGN AND APPLICATIONS** | | | | | | **Elective** |
| **Prerequisite** | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| | **L** | **T** | **D** | **P** | | | |
| **PC602EC** | **3** | **-** | **-** | **-** | **30** | **70** | **3** |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. Understand the evolution of programmable logic devices and FPGA architecture.
2. Learn the complete FPGA design flow using Verilog HDL and industry-standard EDA tools.
3. Design, simulate, synthesize and implement digital systems on FPGA platforms
4. Interface FPGA with memories and communication peripherals
5. Develop FPGA-based solutions for DSP, embedded, AI and communication applications.

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

1. Explain FPGA architecture and programmable logic technologies.
2. Develop synthesizable Verilog HDL designs and verify functionality.
3. Implement digital systems on FPGA considering timing and resource utilization.
4. Design FPGA-based embedded systems with standard interfaces.
5. Apply FPGA technology to real-world applications including DSP, AI and communication.

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 2 | 1 | 2 | - | - | - | - | - | - | 1 |
| **CO2** | 3 | 3 | 2 | 2 | 3 | - | - | - | - | 1 | 1 | 2 |
| **CO3** | 3 | 3 | 3 | 2 | 3 | 1 | - | - | - | 1 | 1 | 2 |
| **CO4** | 3 | 3 | 2 | 2 | 3 | 1 | 1 | - | - | 1 | 1 | 2 |
| **CO5** | 3 | 3 | 2 | 3 | 3 | 1 | 1 | - | - | 1 | 1 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

**UNIT – I**

**Introduction to Programmable Logic Devices:** Evolution of PLDs, PAL, GAL, CPLD and FPGA.

| |
|---|
| FPGA architecture: CLBs, LUTs, flip-flops, programmable interconnects, I/O blocks, clock management, Block RAM, DSP slices, FPGA families (AMD/Xilinx, Intel, Lattice, Microchip). |
| **UNIT – II** |
| FPGA Design Flow using Verilog HDL: RTL design, simulation, testbench development, synthesis, technology mapping, constraints, place and route, bitstream generation, FPGA programming, timing analysis. |
| **UNIT – III** |
| Advanced FPGA Design: FSM design, memory design (ROM, RAM, FIFO), pipelining, parallel processing, clock domain crossing, resource and power optimization, introduction to High-Level Synthesis. |
| **UNIT – IV** |
| FPGA Interfacing and System Integration: UART, SPI, I2C, ADC/DAC interfacing, VGA basics, DDR memory concepts, soft-core processors, SoC FPGA concepts, on-chip debugging. |
| **UNIT – V** |
| FPGA Applications: DSP, image/video processing, AI/ML acceleration, CNN basics, software-defined radio, robotics, IoT edge devices, ASIC prototyping. Reconfigurable computing, partial reconfiguration, and emerging trends in FPGA technology. |


| |
|---|
| **Suggested Text Books:** |
| 1. Pong P. Chu, "FPGA Prototyping by Verilog Examples", Wiley. |
| 2. Stephen Brown and Zvonko Vranesic, "Fundamentals of Digital Logic with Verilog Design", McGraw-Hill. |
| 3. Wayne Wolf, "FPGA-Based System Design", Prentice Hall. |
| **Reference Text Books:** |
| 1. Uwe Meyer-Baese, "Digital Signal Processing with Field Programmable Gate Arrays", Springer. |
| 2. Clive Maxfield, "The Design Warrior's Guide to FPGAs", Newnes. |

| Course Code | Course Title | | | | | | Core/Elective |
|---|---|---|---|---|---|---|---|
| PE623EC | SATELLITE COMMUNICATION | | | | | | Elective |
| Prerequisite | Contact Hours per Week | | | | CIE | SEE | Credits |
| | L | T | D | P | | | |
| PC502EC | 3 | - | - | - | 30 | 70 | 3 |

**Course Objectives:** The course is taught with the objectives of enabling the student to:

1. To understand basics of satellite communications
2. To study various effects on satellite communications and to understand types of antennas used.
3. To study various components in satellite and satellite TV systems.
4. To analyse and design satellite communication link and study various access techniques.
5. To study various applications of satellite communications in practical world.

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

1. Explain principle, working and operation of satellite.
2. Illustrate various effects on satellite communications and its antennas.
3. Explain various components in satellite and satellite TV systems.
4. Analyse and design satellite communication link
5. Illustrate role of satellite in various applications

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 2 | 1 | 2 | - | - | - | - | - | - | 1 |
| **CO2** | 3 | 3 | 2 | 2 | 3 | - | - | - | - | 1 | 1 | 2 |
| **CO3** | 3 | 3 | 3 | 2 | 3 | 1 | - | - | - | 1 | 1 | 2 |
| **CO4** | 3 | 3 | 2 | 2 | 3 | 1 | 1 | - | - | 1 | 1 | 2 |
| **CO5** | 3 | 3 | 2 | 3 | 3 | 1 | 1 | - | - | 1 | 1 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| **UNIT – I** |
|---|
| A Brief History of Satellite Communications, Overview and Indian Scenario of Satellite Communications, Kepler's Laws, Definitions of Terms for Earth-Orbiting Satellites, Orbital Elements, Apogee and Perigee Heights, Orbit Perturbations - Effects of a non-spherical earth, Atmospheric drag |
| **UNIT – II** |
| Antenna Look Angles, The Polar Mount Antenna, Limits of Visibility, Near Geostationary Orbits, Earth Eclipse of Satellite, Sun Transit Outage, Launching Orbits, Atmospheric Losses, Ionospheric Effects, Rain Attenuation, Other Propagation Impairments, Antenna Polarization, Polarization of Satellite Signals, Cross-Polarization Discrimination, Ionospheric Depolarization, Rain Depolarization, Ice Depolarization, Horn Antennas, The Parabolic reflector, Offset feed, Double-reflector antennas |
| **UNIT – III** |
| Power Supply, Attitude Control - Spinning satellite stabilization, Momentum wheel stabilization, Station Keeping, Thermal Control, TT&C Subsystem, Transponders - The wideband receiver, the input demultiplexer, the power amplifier, The Antenna Subsystem.<br>Receive-Only Home TV Systems - The outdoor unit, the indoor unit for analog (FM) TV, Master Antenna TV System, Community Antenna TV System, Transmit-Receive Earth Stations |
| **UNIT – IV** |
| Equivalent Isotropic Radiated Power, Transmission Losses, The Link-Power Budget Equation, System Noise, Carrier-to-Noise Ratio, The Uplink - Saturation flux density, Input backoff, Downlink – Output back-off, Effects of rain – Uplink & Downlink rain-fade margin, Combined Uplink and Downlink C/N Ratio Single Access, Pre-assigned FDMA, Demand-Assigned FDMA, Spade System, TDMA, Pre-assigned TDMA, Demand-assigned TDMA, Satellite-Switched TDMA, CDMA |
| **UNIT – V** |
| C-Band and Ku-Band Home Satellite TV, Digital DBS TV, DBS-TV System Design, DBS-TV Link Budget, Error Control in Digital DBS-TV, Master Control Station and Uplink, Installation of DBS-TV, Antennas, Satellite Radio Broadcasting, Digital Video Broadcast(DVB) Standards, Digital Video Broadcast– Terrestrial (DVB-T) Satellite Mobile Services, VSATs, Radarsat, Global Positioning Satellite System (GPS), Orbcomm, Iridium |

| **Suggested Text Books:** |
|---|
| 1. Dennis Roddy,“ Satellite Communications”,4th Edition,TataMcGraw-Hill. |
| 2. Timothy Pratt, Charles Bostian, Jeremy Allnutt,“ Satellite Communications”,2nd Edition,John Wiley & Sons. |
| 3. WilburL.Pritchard, Henri G. Suyderhoud, Robert A. Nelson, “Satellite Communication Systems Engineering”, 2nd Edition, Pearson |
| **Reference Text Books:** |
| 1. TriT.Ha, Digital Satellite Communication, Tata McGraw-Hill, Special Indian Edition2009. |
| 2. N.Agarwal,“ Design of Geosynchronous Space Craft”,PrenticeHall,1986 |

| **Course Code** | **Course Title** | | | | | | **Core/Elective** |
|---|---|---|---|---|---|---|---|
| PE624EC | **Linux and scripting languages** | | | | | | **Elective** |
| **Prerequisite** | **Contact Hours per Week** | | | | **CIE** | **SEE** | **Credits** |
| | **L** | **T** | **D** | **P** | | | |
| **-** | **3** | **-** | **-** | **-** | **30** | **70** | **3** |

| **Course Objectives:** The course is taught with the objectives of enabling the student to: |
|---|
| 1. Introduce the fundamentals of the Linux operating system, file management, user administration, and command-line utilities. |
| 2. Develop an understanding of Linux networking concepts, network services, and file sharing mechanisms. |
| 3. Familiarize students with scripting language concepts and enable them to automate system administration tasks using Perl. |
| 4. Introduce Tcl/Tk scripting for developing graphical user interfaces and understanding event-driven programming. |
| 5. Enable students to develop Python scripts for solving engineering problems using modular programming, data structures, and object-oriented concepts. |
| **Course Outcomes:** On completion of this course, the student will be able to: |
| 1. Demonstrate proficiency in Linux operating system fundamentals |
| 2. Apply Linux networking concepts and configure basic network services |
| 3. Develop automation scripts using Perl to perform file handling |
| 4. Design GUI-based applications using Tcl/Tk by utilizing scripting constructs |
| 5. Develop Python programs using data structures |

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | - | 1 | - | - | - | - | - | - | 2 |
| **CO2** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | - | - | 2 |
| **CO3** | 3 | 3 | 2 | 2 | 2 | - | - | - | - | - | - | 2 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO4** | 3 | 2 | 3 | 2 | 2 | - | - | - | 1 | - | - | 2 |
| **CO5** | 3 | 2 | 3 | 1 | 2 | - | - | - | 1 | - | 1 | 2 |

Correlation rating: Low / Medium / High: 1 / 2 / 3 respectively.

| |
|---|
| **UNIT – I** |
| **Linux Basics:** Introduction to Linux, File System of the Linux, General usage of Linux kernel & basic commands, Linux users and group, Permissions for file, directory and users, Searching a file & directory, zipping and unzipping concepts. |
| **UNIT – II** |
| **Linux Networking:** Introduction to Networking in Linux, Network basics & tools, File transfer protocol in Linux, Network file system, Domain Naming Services, Dynamic hosting configuration Protocol & Network information Services. |
| **UNIT – III** |
| **Perl Scripting:** Introduction to Perl Scripting, working with Simple Values, Lists and Hashes, Loops and Decisions, Regular Expressions, Files and Data in Perl Scripting, References &Subroutines, Running and Debugging Perl, Modules, Object-Oriented Perl. |
| **UNIT – IV** |
| **TCL/TK Scripting:** TCL Fundamentals, String and Pattern Matching, TCL Data Structures, Control Flow Commands, Procedures and Scope, Evel, Working With UNIX, Reflection and Debugging, Script Libraries, TK Fundamentals, TK by Examples, The Pack Geometry Manager, Binding Commands to X Events, Buttons and Menus, Simple TK Widgets, Entry and List box Widgets Focus, Grabs and Dialogs. |
| **UNIT – V** |
| **Python Scripting:** Introduction to Python, Using the Python Interpreter, More Control Flow Tools, Data Structures, Modules, Input and Output, Errors and Exceptions, Classes, Brief Tour of the Standard Library. |

**Suggested Text Books:**

1. Daniel J Barrett "Linux Pocket Guide: Essential Commands" O'REILLY, 1st Edition, 2016.
2. Brent Welch, Ken Jones, and Jeff Hobbs, "Practical Programming in Tcl and Tk", Prentice Hall, 4th edition, 2003.
3. Anurag Gupta, G Biswas "Python Programming" McGraw Hill, 1st Edition, 2019.

**Reference Text Books:**

1. Red Hat Enterprise Linux 4: System Administration Guide Edition
2. Copyrights 2005 Red Hat, Inc. 2. Tom Christiansen, brian d foy, Larry Wall, Jon Orwant "Programming Perl" O'REILLY, 4th Edition,2012.