OPEN ELECTIVES

B.Tech.

ELECTRONICS AND COMMUNICATION ENGINEERING

22EC855	-	ADHOC Sensor Networks
22EC856	-	Design Principles of IoT
22EC857	-	Introduction to Internet of Things
22EC858	-	Introduction to IoT Architecture
22EC859	-	Machine Learning for IoT Systems
22EC860	-	Security Aspects of IoT

COURSE CONTENTS

I SEM & II SEM

22EC855 ADHOC SENSOR NETWORKS

Source; https://www.elprocus.com/what-is-android-introduction-features-applications/

Hours Per Week :

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Any programming language with Oops concepts.

COURSE DESCRIPTION AND OBJECTIVES:

The aim of the course is that the student will be able to develop android application using Java in Android Studio.

MODULE–1

UNIT-1 **8L+0T+8P=16 Hours**

UNDERSTANDING ANDROID OS:

Android App Creation in Android Studio, Overview of the Android Architecture, The Anatomy of an Android Application, Overview of Android View Binding, Understanding Android Application and Activity Lifecycles

UNIT-2 **8L+0T+8P=16 Hours**

UNDERSTANDING STATES AND ACTIVITIES:

Handling Android Activity State Changes, Saving and Restoring the State of an Android Activity, Understanding Android Views, View Groups and Layouts, Android Constraint Layout, Android Touch and Multi-touch Event Handling

Implementing gestures and touch: Detecting Common Gestures Using the Android Gesture Detector Class, Implementing Custom Gesture and Pinch Recognition on Android,

PRACTICES:

- Develop an application that uses GUI components, Font and Colors.
- Develop an application that uses Layout Managers and event listeners.
- Develop a native calculator application.
- Write an application that draws basic graphical primitives on the screen.
- Develop an application that makes use of database.
- Develop an application that makes use of RSS Feed.

MODULE-2

UNIT-1 **8L+0T+8P=16 Hours**

DESIGNING ANDROID COMPONENTS:

Modern Android App Architecture with Jetpack, An Android Jetpack View Model Tutorial, Working with the Floating Action Button and Snackbar, Creating a Tabbed Interface using the TabLayout Component

WORKING WITH INTENTS AND NOTIFICATIONS:

Adding Sample Data to a Project, Working with the AppBar and Collapsing Toolbar Layouts, Overview of Android Intents, Android Explicit & Implicit Intents, Android Broadcast Intents and Broadcast Receivers, Overview of Android Services & Notifications, Foldable Devices and Multi-Window Support

UNIT-2 **8L+0T+8P=16 Hours**

ACCESSING STORAGE AND MULTIMEDIA

An Android Storage Access Framework, Video Playback on Android using the VideoView and MediaController Classes, Making Runtime Permission Requests in Android, Android Audio Recording and Playback using MediaPlayer and MediaRecorder

Creating and testing android app:

Android App Links, Creating, Testing and Uploading an Android App Bundle

PRACTICES:

- Implement an application that implements Multi threading.
- Develop a native application that uses GPS location information.
- Implement an application that writes data to the SD card.
- Implement an application that creates an alert upon receiving a message.
- Write a mobile application that creates alarm clock.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Implement basic states, gestures for Android applications	Create	1	1, 2, 4, 9, 10, 11, 12
2	Identify and design the components for developing android applications	Apply	1	1, 2, 3, 4, 9, 10, 11, 12
3	Display notifications and access storage and multimedia in android OS	Apply	2	1, 2, 3, 4, 9, 10, 11, 12
4	Design an android app for a given need	Create	2	1, 2, 3, 4, 9, 10, 11, 12

TEXTBOOKS:

1. Smyth, Neil, “Android Studio 4. 2 Development Essentials - Java Edition”, 2021
2. Wei-Meng Lee, “Beginning Android 4 Application Development”, 1st edition, Wiely Publishers, 2011.

REFERENCEBOOKS:

1. Prasanna Kumar Dixit, “Android”, 1st edition, Vikas Publishers, 2014.
2. Jerome (J.F.) DiMarzio, “Android - A programmers Guide”, 1st edition, Tata Mc Graw Hill, 2010.
3. Reto Meier, “Professional Android 4 Application Development”, 1st edition, Wiley Publishers, 2008
4. John Horton, “Android Programming for Beginners”, 1st edition, Pact Publishing, 2015.

22EC856 INTERNET OF THINGS

Source: https://www.rtsrl.eu/blog/what-is-internet-of-things-iot/

Hours Per Week :

L	T	P	C
2	0	2	3

PREREQUISITE KNOWLEDGE: Basics of Internet.

COURSE DESCRIPTION AND OBJECTIVES:

This course introduces the introduction to Internet of Things and the basic concepts. The course emphasizes design issues and utilization of IoT devices, including various sensors and hardware boards.

MODULE –1

UNIT-1 **8L+0T+8P=16 Hours**

INTRODUCTION & CONCEPTS:

Introduction to Internet of Things, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, IoT Levels

UNIT-2 **8L+0T+8P=16 Hours**

INTERNET PRINCIPLES:

Internet Communications: An Overview, IP Addresses, MAC Addresses, TCP and UDP Ports, Application Layer Protocols, Python packages of interest for IoT.

DOMAIN SPECIFIC APPLICATIONS OF IOT

Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health & Life Style.

PRACTICES:

- Familiarization of various hardware boards.
- To analyze the IP address of Personal Computer
- Working principles of various IoT protocols.
- Implementation of MQTT protocol using Arduino board
- Python packages for implementing IoT applications

MODULE-2

UNIT-1 **8L+0T+8P=16 Hours**

INTERNET PRINCIPLES & M2M:

M2M: Introduction to M2M, M2M architecture, Difference between IoT and M2M, SDN and NFV for IoT.

IEEE 802.15.4: Physical layer, MAC layer, Uses and future of 802.15.4.

Zigbee: Architecture, Association, Network layer, APS layer and security.

Z-Wave: Z-wave Protocol.

UNIT-2 **8L+0T+8P=16 Hours**

CASE STUDY & ADVANCED IOT APPLICATIONS:

IoT applications in home, infrastructures, buildings, security, Industries, Home appliances, other IoT electronic equipments.

SKILLS:

- ✓ *Understand the specifications and how well different components work together using IoT.*
- ✓ *Learn above different sensors for collecting data.*
- ✓ *Design a prototype for various IoT applications.*

Case study illustrating IoT design: Home Automation, Smart Cities, Environment, Agriculture, Productivity Applications.

PRACTICES:

- Interfacing of LED and switch with Arduino boards and ESP8266.
- Traffic Light control using Arduino board and ESP8266.
- Interfacing DHT11 sensor with Arduino board and ESP8266.
- Interfacing of ultrasonic sensor with Arduino board and ESP8266.
- Interfacing of PIR sensor with Arduino board and ESP8266.
- DC motor control using L293D motor driver and Arduino board.
- Familiarization with Raspberry pi.
- Interfacing of LED and switch with Raspberry pi.
- Interfacing PIR sensorwith Raspberry pi.
- Interfacing DHT11 sensor with Raspberry pi.
- Interfacing of ultrasonic sensor with Raspberry pi.
- Interfacing of Picam with Raspberry pi.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Able to programs for IoT applications.	Apply	1	1, 2, 12
2	Able to design the framework necessary for IoT applications	Apply	1	1, 2, 5, 12
3	Able to develop prototypes for IoT devices	Apply	1	1, 2, 3, 5, 12
4	Categorize various IoT applications	Apply	2	1, 2, 12
5	Classify various advanced IoT applications and case studies.	Apply	2	1, 2

TEXT BOOKS:

1. Vijay Madisetti, Arshdeep Bahga," Internet of Things A Hands-On- Approach", 2014,
2. Adrian McEwen, "Designing the Internet of Things", Wiley Publishers, 2013,

REFERENCES BOOKS:

1. Francis daCosta, "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything", 1 st Edition, Apress Publications, 2013.
2. Cuno Pfister, Getting Started with the Internet of Things, O"Reilly Media, 2011, ISBN: 978-1-4493- 9357-1.
3. Kamal R. Internet of Things, McGraw Hill, 2017.

22EC857 INTRODUCTION TO EMBEDDED SYSTEMS

Source: https://cprimestudios.com/blog/5-myths-about-embedded-systems-development

Hours Per Week :

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Computer Architecture and Organization, Microprocessors and Microcontrollers.

COURSE DESCRIPTION AND OBJECTIVES:

The course objective is to study the applications, categories, hardware and software architectures, memory, testing tools in embedded systems, Firmware, Embedded C, operating system functions and various kernel objects and RTOS.

MODULE-1

UNIT-1 **6L+0T+6P=12 Hours**

INTRODUCTION:

Basic concepts, Applications and Categories of embedded systems, Hardware architecture, Software architecture of Embedded Systems, Process of generating executable images, Development/testing tools.

UNIT-2 **10L+0T+10P=20 Hours**

PROGRAMMING:

Comparison of Assembly and C languages, C and Embedded C. Programming in C: Arrays, Structures, Loops and Decisions, Pointers, Functions, Embedded C: Header files for Project and Header files for Port.

PRACTICES:

- Programming with Embedded C using any compiler.
- Demonstration/Practical session for creation of header files.
- Program to create loops in Embedded C
- Program to implement decisions in Embedded C
- Develop program to implement interrupt function

MODULE-2

UNIT-1 **6L+0T+6P=12 Hours**

OPERATING SYSTEMS:

Introduction to Operating Systems, Process and threads, Scheduling, Non-preemptive and Preemptive scheduling, Real Time Scheduling.

UNIT-2 **10L+0T+10P=20 Hours**

Introduction to Real Time Operating Systems, Shared Data Problem, Semaphores, Priority inversion problem, Inter process/task communication techniques.

PRACTICES:

- Create and schedule a process/task
- Demonstrate shared data problem
- Create and use semaphores

SKILLS:

- ✓ *Choose component for Embedded System*
- ✓ *Understand operating system concepts*
- ✓ *Understand*

- Find schedulability using Gantt charts
- Implement IPC techniques

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Identify the components of embedded systems and differentiate various embedded systems	Analyze	1	1, 2, 3, 4, 9, 10, 11, 12
2	Design embedded systems using standard procedure	Create	1	1, 2, 3, 4, 9, 10, 11, 12
3	Choose necessary component and buses for the embedded system	Apply	2	1, 2, 3, 4, 9, 10, 11, 12
4	Apply the knowledge of operating system functions and various kernel objects	Apply	2	1, 2, 3, 4, 9, 10, 11, 12

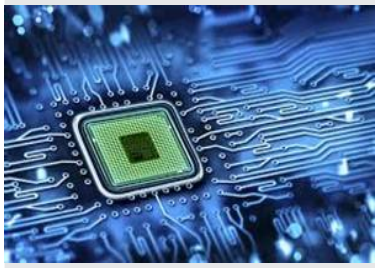
TEXT BOOKS:

1. Raj Kamal, “Embedded Systems Architecture, Programming and Design”, 3rd edition, Mc Graw Hill, 2017.
2. Lyla B. Das, “Embedded Systems An Integrated Approach,” Pearson Education, 2013

REFERENCE BOOKS:

1. Marilyn wolf, “Computers as Components: Principles of Embedded Computer systems design”, 4th edition, Morgan Kaufmann Publishers, 2017.
2. Dr. K.V.K.K. Prasad, “Embedded Real time Systems”, Black book, Dreamtech Press, 2003.
3. Daniel W. Lewis, “Fundamentals of Embedded Software: Where C and Assembly Meet”, 1st edition, Pearson, 2001.
4. John Catsoulis, “Designing Embedded Hardware”, 2nd Edition, O’Reilly Media, Inc., 2005.
5. “Getting Started with Arduino: The Open Source Electronics Prototyping Platform”, 3rd edition, Maker Media Inc., 2015.

22EC858 MICROPROCESSORS AND MICROCONTROLLERS



Source: https://www.google.com /search?q=Microcontroller&sxsrf=AliCzsbnrzqS2pUyMf6IF_

Hours Per Week :

L	T	P	C
2	0	2	3

PREREQUISITE KNOWLEDGE: Digital Electronics.

COURSE DESCRIPTION AND OBJECTIVES:

This course introduces the principles of microprocessors and microcontrollers. It also deals with the programming concepts of microprocessors and microcontrollers.

MODULE - 1

UNIT-1 **6L+0T+6P=12 Hours**

8086 MICROPROCESSOR:

Architecture, Flag register, Signals, Memory segmentation, Physical address generation, Minimum mode, Maximum mode, Interrupts, Memory organization.

UNIT-2 **10L+0T+10P=20 Hours**

8086 INSTRUCTIONS:

Data transfer instructions, Arithmetic instructions, Logical instructions, Branch instructions, String instructions, machine control instructions; Addressing Modes, Assembler directives, Procedures and macros.

PRACTICES:

- Introduction to TASM software.
- 8-bit, 16-bit Addition, Subtraction.
- 8-bit, 16-bits Multiplication and Division.
- Find square, cube and square root.
- Find number of zeros and ones, Separate the 10s and 1s of a given 8-bit number.
- Searching a number, Find and replace the number in a given array.
- Convert Hexa to BCD and Hexa to ASCII.
- Factorial, sum of n numbers, average of n numbers.
- Find the smallest/largest number, Arrange the given numbers in sorting order.
- Block transfer using string instructions.
- Display of character/ string on console using DOS INT 21H function calls.
- File management using DOS INT 21H function calls.

MODULE-2

UNIT-1 **6L+0T+6P=12 Hours**

8051 MICROCONTROLLERS:

Comparing Microprocessors and Microcontrollers; Selection of Microcontrollers, Architecture, PSW, Signals, Memory organization, Instruction set, Addressing modes of 8051.

UNIT-2 **10L+0T+10P=20 Hours**

8051 COMPONENTS:

On-chip Components: Parallel Ports, Timers/Counters, Serial port, Interrupts. Interfacing with 8051: LCD, Keyboard, Stepper Motor.

SKILLS:

- ✓ *Understand the specifications and how well different components work together for IoT Boards.*
- ✓ *Learn different data and number representations.*
- ✓ *Design ALU and Control unit.*
- ✓ *Identify the types of IoT application protocols and their uses.*
- ✓ *To enable the students to take up the real-time industry as well as interdisciplinary projects.*

PRACTICES:

- Introduction to Keil vision 4 software,
- Addition, Subtraction, Multiplication and Division.
- Find number of zeros and ones, Separate the 10s and 1s of a given 8bit number.
- Factorial, sum of n numbers, average of n numbers.
- Find the smallest/largest number, Arrange the given numbers in sorting order.
- Addition, Subtraction, Reverse subtraction.
- Find number of zeros and ones, Separate the 10s and 1s of a given 8bit number.
- Factorial, sum of n numbers, average of n numbers.
- Find the smallest/largest number, Arrange the given numbers in sorting order.
- Blinking of LED's, Reading Switches and Glowing LED's using Assembly and C.
- 7 segment LED with 8051.
- LCD module with 8051.
- Stepper motor speed and rotation control using 8051.
- Waveform generation using DAC with 8051.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Architect a microprocessor or microcontroller system and estimate the required hardware and software resources.	Apply	1	1, 2, 3, 4, 9, 10, 11, 12
2	Select a microprocessor or microcontroller suitable to the application.	Analyse	1	1, 2, 3, 4, 9, 10, 11, 12
3	Write assembly language program in 8086 for various applications.	Apply	1	1, 2, 3, 4, 9, 10, 11, 12
4	Create the memory and IO interfacing techniques 8051.	Create	2	1, 2, 3, 4, 9, 10, 11, 12
5	Write assembly language program in microcontroller 8051 for various applications	Apply	2	1, 2, 3, 4, 9, 10, 11, 12

TEXT BOOKS:

1. Kenneth J. Ayala, "The 8086 Microprocessor: Programming and Interfacing the PC", 3rd edition, Cengage Learning, 2007.
2. Muhammad Ali Mazidi, Janice GillispieMazidi, RolinD.Mckinlay, "The 8051 Microcontroller and Embedded Systems", 2nd edition, Pearson Education, 2012.

REFERENCE BOOKS:

1. Barry B. Brey, "The Intel microprocessors 8086/8088, 80186/80188, 80286, 80386,80486, Pentium, Pentium Pro processor, Pentium II, Pentium III, Pentium 4, and Core2 with 64-bit extensions: architecture, programming, and interfacing", 8th edition, Pearson Prentice Hall, 2014.
2. Yu Cheng Liu and Glenn A Gibson, "Microcomputer Systems: The 8086/8088 Family Architecture Programming and Design", 2nd edition, Prentice Hall, 2015.
3. Kenneth J. Ayala, "The 8051 Microcontroller: Architecture Programming and Applications", 3rd edition, Cengage Learning, 2008.
4. K. M. Bhurchandi and A. K. Ray, Advance Microprocessor and Peripherals, 3rd edition, Tata McGraw Hill, 2017.1 st Edition, Apress Publications, 2013.

22EC859 SMART & VIRTUAL INSTRUMENTATION

Source: https://new.siemens.com/uk/en/products/automation/process-instrumentation/smart-instruments.html

Hours Per Week :

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Basic understanding of Sensors, any programming language concepts.

COURSE DESCRIPTION AND OBJECTIVES:

To familiarize students with the smart and intelligent sensors with VI software. Acquire knowledge on Data Acquisition Systems and network interface concepts. Understand various analysis tools and develop programs for Industrial Applications

MODULE–1

UNIT-1 **8L+0T+8P=16 Hours**

INTRODUCTION TO VIRTUAL INSTRUMENTATION:

Computers in Instrumentation, Virtual Instrumentation (VI), LabVIEW and VI, Conventional and Graphical Programming, Components of LabVIEW, Owned and Free Labels, Tools and Other Palettes, Arranging Objects, Pop-Up Menus, Color Coding, Code Debugging.

UNIT-2 **8L+0T+8P=16 Hours**

VI PROGRAMMING TECHNIQUES:

VIs and sub-VIs, Loops and Charts, Arrays, Clusters and graphs, Case and sequence structures, Formula nodes, Local and global variables, Strings and file I/O.

Data Acquisition System: Measurement and Automation Explorer, The Waveform Data Type, Working in DAQmx, Working in NI-DAQ(Legacy DAQ), Use of Simple VIs, Intermediate VIs.

PRACTICES:

- Introduction to LabVIEW
- Use of NI Elvis
- Use of SubVI
- Formula node
- Shift registers
- Array, Strings
- Function Generator
- DC voltage measurement using DAQ

MODULE-2

UNIT-1 **8L+0T+8P=16 Hours**

INTERFACING INSTRUMENTS:

GPIB and RS232: RS232C versus GPIB, handshaking, GPIB interfacing, RS232C/RS485 interfacing, Standard commands for programmable instruments, VISA, Instrument interfacing and LabVIEW.

UNIT-2 **8L+0T+8P=16 Hours**

INTERFACING SMART SENSORS:

Introduction, Classification, Smart Sensors, Cogent Sensors, Soft or Virtual Sensors, Self Adaptive Sensors, Self-Validating Sensors, VLSI Sensors, Temperature Compensating Intelligent Sensors. Film sensors (Thick film sensors, this film sensor), MEMS and Nano-Sensors.

PRACTICES:

- Analog Input and Output Interface
- Frequency Measurement
- Network Interface
- Thermocouple Interface and Celsius to Fahrenheit conversion
- Stepper Motor
- Simulation of Tank Process
- Clusters
- PID controller for DC motor

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Analyze the basic concept of smart sensors, virtual instrument.	Analyze	1	1, 2, 4, 9, 10, 11, 12
2	Create a Virtual Instrument using graphical programming	Create	1	1, 2, 3, 4, 9, 10, 11, 12
3	Develop systems for real-time signal acquisition and analysis.	Apply	2	1, 2, 3, 4, 9, 10, 11, 12
4	Apply concepts of network interface for data communication.	Create	2	1, 2, 3, 4, 9, 10, 11, 12
5	Interface physical parameters with computer through data acquisition systems for practical applications	Apply	2	1, 2, 3, 4, 9, 10, 11, 12

TEXT BOOKS:

1. Dr. Sumathi. S and Prof. Surekha. P, “LabVIEW Based Advanced Instrumentation Systems”, 2nd edition, 2007.
2. Jovitha Jerome, “Virtual Instrumentation using LabVIEW”, PHI Learning Pvt. Ltd, New Delhi, 2010.

REFERENCE BOOKS:

1. Lisa .K, Wells and Jeffrey Travis, “LABVIEW for Everyone”, Prentice Hall, 2009.
2. Skolkoff, “Basic concepts of LABVIEW 4”, PHI, 1998.
3. Gupta. S, Gupta. J.P, “PC Interfacing for Data Acquisition and Process Control”
4. Gary Johnson, “LabVIEW Graphical Programming”, McGraw Hill, 2006.

22EC860 WIRELESS SENSOR NETWORKS

Source: https://www.smart-energy.com/magazine-article/wireless-sensor-network-tech-iiot/

Hours Per Week :

L	T	P	C
2	0	2	3

PREREQUISITE KNOWLEDGE: Basics of computer networks.

COURSE DESCRIPTION AND OBJECTIVES:

This course is targeted at understanding and obtaining hands-on experience with the state of the art in such wireless sensor networks which are often composed using relatively inexpensive sensor nodes that have low power consumption, low processing power and bandwidth. Explore the various MAC routing protocols evolved in wireless sensor networks.

MODULE–1

UNIT- 1 **8 L+0T+8P=16hours**

INTRODUCTION:

Introduction to Wireless Networks, Protocol Suites, and Standards, OSI Model and TCP/IP Protocol Suite, Ad-hoc Networks, Comparison of Ad-hoc and Sensor Networks, applications of WSNs, challenges for WSNs, hardware components of wireless sensor node, energy consumption of a sensor nodes, operating system and execution environments and examples of sensor nodes.

UNIT-2 **8L+0T+8P=16hours**

NETWORK ARCHITECTURE AND PHYSICAL LAYER:

Sensor network scenarios, optimization goals and figures of merit, design principles for wireless sensor networks, service interfaces for wireless sensor networks, gateway concepts, wireless channel and communication fundamentals, physical layer, and transceiver design considerations in wireless sensor networks.

MODULE-2

UNIT-1 **8L+0T+8P=16hours**

MAC LAYER PROTOCOLS FOR WIRELESS SENSOR NETWORKS:

Fundamentals of wireless MAC protocols, Low duty cycle protocols and wakeup concepts, contention-based protocols, schedule- based protocols, IEEE 802.15.4 MAC protocols, error control and link layer management.

UNIT-2 **8L+0T+8P=16hours**

ROUTING PROTOCOLS FOR WIRELESS SENSOR NETWORKS:

The forwarding and routing concept, Gossiping and agent-based unicast forwarding, energy efficient unicast methods, broadcast and multicast methods, geo-graphic routing methods and mobile nodes, TEEN, APTEEN and SPIN protocols.

PRACTICES:

Experiments to be carried out in any network simulator like NETSIM, NS2 and OMNET++ etc.

- Examine the various path loss models available for wireless networks
- Identify the various reasons for hidden node terminal problem in wireless ad-hoc networks.
- Create a scenario where both ad-hoc and wireless sensor network are available and examine he interference problem.

SKILLS:

- ✓ *Able to adapt the wireless sensor network with sensor nodes which have limitations in power consumption, processing power and bandwidth.*
- ✓ *Able to specify the requirements for the hardware and software solutions for energy-efficient sensor network for new applications.*
- ✓ *Able to apply appropriate algorithms to improve existing or to develop new wireless sensor network applications.*

- Simulate the MAC routing protocols for wireless sensor networks.
- Simulate the multiple wireless scenarios for throughput performance.
- Analyze performance of the hierarchical routing protocols in various parameters like end-to-end delay, bit error rate and throughput.
- Design a wireless sensor network and observe how the power consumption consumed for the network.
- Examine the importance of channel accessing mechanism helped for getting high throughput in wireless networks.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Analyze the various solutions involved for designing WSN	Analyze	1	1, 2, 4, 9, 10, 11, 12
2	To identify the Wireless Sensor Network node architecture and real time nodes.	Apply	1	1, 2, 3, 4, 9, 10, 11, 12
3	Analyze the performance of Medium Access Control protocols for power consumption, fairness, channel utilization and control packet overhead.	Analyze	2	1, 2, 3, 4, 9, 10, 11, 12
4	Evaluate the performance of routing protocols for wireless sensor network.	Evaluate	2	1, 2, 3, 4, 9, 10, 11, 12
5	Solve IoT security problems using light weight cryptography	Analyze	2	1, 2, 3, 5

TEXT BOOKS:

1. Holger Karl, Andreas Willig “Protocols and Architecture for Wireless Sensor Networks” John Wiley and Sons, Ltd, 2007
2. Feng Zhao and Leonides Guibas, “Wireless Sensor Networks”, Elsevier Publication–2002.

REFERENCE BOOKS:

1. C. Siva Ram Murthy and B. S. Manoj, “Ad Hoc Wireless Networks Architectures and Protocols”, Prentice Hall, PTR, 2004.
2. Kazem sohraby, Daniel Minoli and Taieb Znati, “Wireless Sensor Networks: Technology, Protocols and Application” John Wiley, 2007.
3. C.K Toh, “Ad-Hoc Mobile Wireless Networks: Protocols and Systems” 1st edition, Pearson, 2007.