IoT

Target Audience – Undergraduate / Graduate Students (B.Sc, M.Sc, B.Tech, BCA, MCA)

Duration – 120 Hrs

Theory Class – Initially with trainer lead classes, further will go with video content

Practical Class – a. With Online/Offline Simulator

b. With IoT KIT

Pre Prerequisites–Basic knowledge of computer science, Mathematics, and programming languages (e.g., Python, C/C++)

Syllabus

1. **Introduction to IoT**
   - IoT Fundamentals
   - IoT Ecosystem and Applications
   - IoT Architecture and Protocols
   - Characteristics and Building Blocks of IoT
   - Levels of IoT

2. **Embedded System and Electronics Basic**
   - Introduction to Embedded System
   - Basics of Electronics (Voltage, Current, Resistance)
   - Digital Circuit and Logic Design
   - Data Communications and Computer Networks (Basics relevant to IoT)

3. **Introduction to Arduino**
   - Introduction to the Arduino Platform
   - Arduino IDE and Basics of Programming
   - Types of Arduino Boards and their Applications

4. **Arduino Programming**
   - Arduino Langugae (C/C++/Python)
   - Programming Structures and Syntax
   - Variables, Datatypes and Operators
   - Conditional Statements, Control Statements
   - Arrays and Functions

5. **Sensors and Actuators with Arduino**
   - Microprocessors and their applications
   - Microcontrollers and their applications
   - Basics of Sensors - IR, DHT11, HC-SR04, LDR, PIR, MQ135
   - Basics of Actuator - Servo motor and DC motor

- Seven segments LED, RGB LED, Relay, Keypads
- Display the data on Liquid Crystal Display(LCD), interfacing keypad

6. IoT Communication and Connectivity

- Bluetooth Module – HC-05, Control/handle 220V AC supply – interfacing relay module
- NODEMCU wifi Module – ESP8266
- Introduction to MQTT Protocol
- Ethernet shield

7. Project Work

- Hands-on IoT Project integrating various sensors, actuators, and communication modules.

Detailed Syllabus of IoT

1. **Introduction to IoT**

- IoT Fundamentals: Understanding the basic concepts and principles of the Internet of Things, including its definition, scope, and significance in various domains.

- **IoT Ecosystem and Applications:** Exploring the diverse applications and domains where IoT technologies are being implemented, such as smart cities, healthcare, agriculture, and industry.

- **IoT Architecture and Protocols:** Delving into the architectural components of IoT systems, including devices, connectivity, data processing, and applications. Exploring communication protocols like MQTT, CoAP, HTTP, etc.

- **Characteristics and Building Blocks of IoT:** Identifying the key characteristics and components that define IoT systems, including sensors, actuators, connectivity, and data analytics.

- Levels of IoT: Understanding the different levels of IoT deployment, from edge devices to cloud-based platforms, and their respective roles in IoT ecosystems.

2. **Embedded System and Electronics Basics**

- **Introduction to Embedded Systems:** Understanding the fundamentals of embedded systems, including their definition, characteristics, and applications in IoT.

- **Basics of Electronics (Voltage, Current, Resistance):** Reviewing fundamental electronics concepts such as voltage, current, resistance, and Ohm's law, essential for understanding sensor and actuator interfacing.

- **Digital Circuit and Logic Design:** Exploring digital logic gates, Boolean algebra, and basic digital circuit design principles relevant to embedded systems and microcontrollers.

- Data Communications and Computer Networks (Basics relevant to IoT): Introducing fundamental concepts of data communications and computer networks, including protocols, network topologies, and communication mediums relevant to IoT applications.

3. **Introduction to Arduino**

- Introduction to the Arduino Platform: Familiarizing students with the Arduino ecosystem, including hardware platforms, development environments, and community support.

- **Arduino IDE and Basics of Programming:** Introducing the Arduino Integrated Development Environment (IDE) and basic programming concepts for microcontroller-based systems.

- **Types of Arduino Boards and their Applications:** Exploring different Arduino board variants and their specific features, capabilities, and applications in IoT projects.

4. **Arduino Programming**

- Arduino Language (C/C++/Python): Learning the basics of the C/C++/Python programming language and its application in Arduino development.

- **Programming Structures and Syntax:** Understanding programming structures such as loops, conditional statements, and functions in the context of Arduino programming.

- **Variables, Datatypes, and Operators:** Exploring data types, variables, and operators supported by Arduino and their usage in programming.

- **Conditional Statements, Control Statements:** Implementing conditional and control statements to create decision-making algorithms and control structures in Arduino programs.

- **Arrays and Functions:** Understanding the use of arrays and functions for managing data and code organization in Arduino sketches.

5. **Sensors and Actuators with Arduino**

- **Microprocessors and their applications:** Understanding the role and functionality of microprocessors in embedded systems and IoT devices.

- **Microcontrollers and their applications:** Exploring the capabilities and applications of microcontroller-based systems, with a focus on Arduino-compatible microcontrollers.

- **Basics of Sensors and Actuators**: Introducing various sensors (e.g., IR, DHT11, HC-SR04) and actuators (e.g., servo motor, DC motor) commonly used in IoT projects.

- **Interfacing Sensors and Actuators with Arduino:** Practical exercises on connecting and interfacing sensors and actuators with Arduino boards, including input/output methods and signal processing.

- **Seven segments LED, RGB LED, Relay, and Keypads:** Exploring additional electronic components commonly used in IoT projects, such as LEDs, relays, and keypads.

- **Displaying Data on Liquid Crystal Display (LCD) and interfacing keypad:** Demonstrating how to display sensor data and interact with users using LCD displays and keypads connected to Arduino boards.

6. **IoT Communication and Connectivity**

- **Bluetooth Module (HC-05) and Relay Module Interfacing:** Learning how to interface Bluetooth modules for wireless communication and relay modules for controlling high-voltage AC devices.

- **NODEMCU Wi-Fi Module (ESP8266):** Exploring Wi-Fi connectivity options for IoT devices using the ESP8266-based NODEMCU module.

- **Introduction to MQTT Protocol:** Understanding the Message Queuing Telemetry Transport (MQTT) protocol for lightweight messaging communication between IoT devices and platforms.

- **Ethernet Shield:** Exploring wired Ethernet connectivity options for IoT devices using Ethernet shields compatible with Arduino boards.

7. **Project Work (Minimum 5 Projects with description)**

- **Hands-on IoT Project:** Applying the knowledge and skills gained throughout the course to design and implement a real-world IoT project integrating various sensors, actuators, communication modules, and Arduino-based microcontrollers.

- **Integration of Online/Offline Simulators:** Utilizing online/offline simulators such as Tinkercad, SmarDen.tech, Blynk, etc., to simulate and test IoT projects in a virtual environment before deployment.

Reference Books/Study Material

1. "The C Programming Language" by Brian Kernighan and Dennis Ritchie.
2. "Digital Design: Principles and Practices" by John F. Wakerly.
3. "Computer Networking: A Top-Down Approach" by James F. Kurose and Keith W. Ross
4. "Internet of Things: Principles and Paradigms" by Rajkumar Buyya, Amir Vahid Dastjerdi, Elsevier, 2016.
5. Macro Schwartz, “Internet of Things with Arduino- Cookbook”, Packt 2016
6. "Sensor Technology Handbook" by Jon S. Wilson.
7. "Wireless Sensor Networks: Principles, Design and Applications" by Shakil Akhtar
8. Massimo Banzi, “Getting started with Arduino”, 2nd Edition, Oreilly, 2011[Make:Makezine.com]
9. Macro Schwartz, “Internet of Things with Arduino”, Open Home Automation
10. Michael Margolis, “Arduino Cookbook”, Oreilly, 2011

Online/ Offline Simulator –

1. **Tinkercad**
   https://www.tinkercad.com/
2. **SmarDen.tech**
   https://www.smarden.tech/
3. **Blynk**
   https://blynk.io/

**** And any new suggestions are highly appreciated.**

LAB EXPERIMENTS
Lab 1: Interfacing Light Emitting Diode (LED) - Blinking LED - Objective: Familiarize students with the GPIO peripheral of the ATmega microcontroller and basic digital output. - Task: Write a program to blink an LED connected to an Arduino board at a specific frequency.
Lab 2: Reading Potentiometer Values and Controlling LED Brightness (Night Lamp) - Objective: Introduce students to analog input handling and PWM output for controlling LED brightness. - Task: Interface a potentiometer with Arduino to control the brightness of an LED, simulating a night lamp.
Lab 3: Interfacing Temperature and Humidity Sensors - Objective: Learn to connect environmental monitoring sensors to Arduino and use appropriate libraries. - Task: Interface a temperature sensor (e.g., LM35) and/or humidity sensor (e.g., DHT11) with Arduino to monitor environmental conditions.
Lab 4: Controlling Servo Motors - Objective: Understand servo motor control using PWM signals. - Task: Interface a servo motor with Arduino and write a program to control its position.
Lab 5: Interfacing with LCD Displays - Objective: Display sensor data on an LCD screen for local monitoring. - Task: Interface an LCD display with Arduino and display data generated by a sensor.
Lab 6: Bluetooth-based Home Automation - Objective: Enable IoT capabilities by integrating actuators (relays) with Arduino and remote connectivity via Bluetooth. - Task: Interface a relay module with Arduino to control home appliances wirelessly using a mobile phone application.
Lab 7: Advanced Sensor Integration - Accelerometer and Gyroscope - Objective: Explore advanced sensor integration for motion sensing applications. - Task: Interface an accelerometer and gyroscope with Arduino to detect and measure motion.
Lab 8: Reading Analog Sensor Values Objective: Learn to read analog sensor values and convert them to digital format. Task: Interface analog sensors (e.g., light sensor, temperature sensor) with Arduino and read their values.
Lab 9: Interfacing Air Quality Sensor Objective: Learn to use pollution sensors with Arduino for smart city applications. Task: Interface an air quality sensor (e.g., MQ135) with Arduino to measure pollution levels and trigger an LED indicator.
Lab 10: Interfacing Button and LED - Objective: Understand GPIO in both input and output modes. - Task: Interface a button and an LED with Arduino and write a program to blink or glow the LED when the button is pressed.
Lab 11: Project Showcase - Mini Smart Home System - Objective: Apply knowledge and skills acquired in previous labs to design and implement a mini smart home system. - Task: Design and implement a project integrating multiple sensors, actuators, and communication modules for home automation.