IT 705 – Internet of Things

CAT - 2

01.04.2025

Answer all Questions

PART A – (5×2 = 10 Marks)

1. How does **WoT** differ from **IoT**? Give an example.

IoT (Internet of Things):

- Refers to the network of physical devices connected to the internet that collect and exchange data.
- Focuses on connectivity and communication between devices using sensors, actuators, and embedded systems.

WoT (Web of Things):

- Builds on top of IoT by integrating web standards and technologies (like HTTP, REST, JSON) to interact with IoT devices.
- Makes IoT devices accessible and manageable through the web, enabling better interoperability and ease of use.

Example:

- **IoT:** A smart thermostat sends temperature data to the cloud using MQTT.
- **WoT:** The same thermostat provides a RESTful API to control or monitor it via a web browser or mobile app.

2. Compare **SOA** and **SODA**?

Feature	**SOA**	**SODA**
Focus	Software Services	Physical Devices as Services

Feature	SOA	SODA
Application	Enterprise systems, web services	IoT systems, sensor networks
Components	Software modules	Devices and their interfaces
Communication	Web protocols (SOAP, REST)	Lightweight protocols + Web standards

Example:

- **SOA:** A banking web app uses a payment service via SOAP.
- **SODA:** A smart sensor offers its readings as a web service for other systems to access.

3. State the reasons for using **Modbus** in the industrial environment?

Reasons for using Modbus in the industrial environment:

- **Simple and easy** to implement.
- **Open protocol** – no license needed.
- Supports **communication between different devices** (interoperability).
- Works with both **serial (RTU/ASCII)** and **network (TCP/IP)** modes.
- **Reliable** in harsh industrial settings.
- **Widely used** in industries like manufacturing, energy, etc.

4. What is meant by **software framework**?

A **software framework** is a set of **predefined tools, libraries, and rules** that helps developers build applications faster and more efficiently.

- It provides **structure** and **reusable components**.

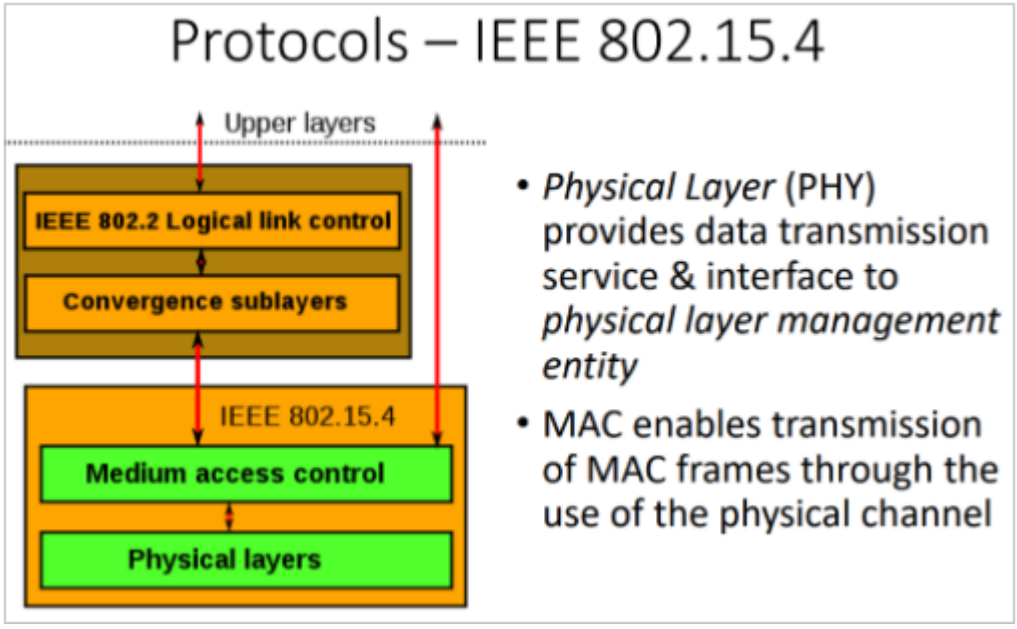
- Promotes **standardization** and **code organization**.
- Example: **Django** for Python web development, **React** for building user interfaces.

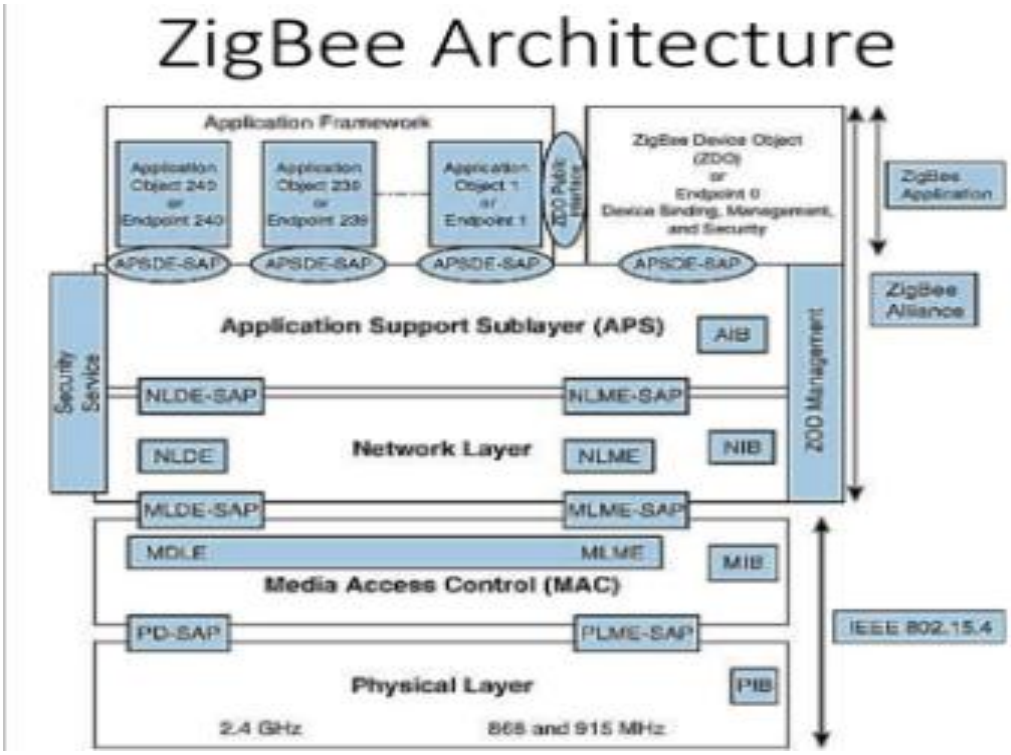
5. Compare **SOAP** and **REST** protocols?

Feature	**SOAP**	**REST**
Protocol	Strict protocol	Architectural style
Format	XML only	JSON, XML, HTML, etc.
Speed	Slower	Faster
Complexity	More complex	Simple and lightweight
Flexibility	Less flexible	Highly flexible

PART B – (3×5 = 15 Marks)

6. Draw the protocol stack of **IEEE 802.15.4** and **Zigbee** architecture.





7. Explain the **unified multitiered WoT architecture** in detail.

The **Unified Multitiered Web of Things (WoT) Architecture** is a layered model that organizes how IoT devices interact using web technologies. It ensures **scalability**, **interoperability**, and **integration** of heterogeneous devices through the web.

Layers of the Architecture:

1. **Perception Tier (Device Tier):**
    - Contains **physical devices** like sensors, actuators, RFID, and embedded systems.
    - Responsible for **data collection** from the environment.
    - Devices generate raw data (temperature, motion, humidity, etc.).
2. **Network Tier (Communication Tier):**
    - Transmits data from perception devices to the service or middleware layer.

- Uses communication technologies like **Wi-Fi, Bluetooth, Zigbee, 4G/5G, Ethernet**, etc.
- Handles **data routing, switching, and transmission**.

3. **Service Tier (Middleware Layer):**
   - Processes and stores data received from the network layer.
   - Offers **APIs** and **web services** to interact with devices.
   - Ensures **security, authentication, and data filtering**.
   - Converts device-specific data into **standard web formats** (like JSON, XML).
4. **Application Tier (User Interface Layer):**
   - Provides user-friendly **interfaces** (web/mobile apps) for accessing device functionalities.
   - Interacts with the service layer through **RESTful APIs** or **WebSockets**.
   - Enables users to **monitor, control, and visualize** device data.
5. **Business Tier (Optional – Decision Making):**
   - Analyzes data for **business intelligence**, decision-making, and automation.
   - Can integrate **AI, analytics, and cloud computing** for insights.
   - Example: Sending alerts if temperature exceeds threshold.

Advantages:

- **Interoperability:** Supports various devices using common web standards.
- **Scalability:** Each tier can scale independently.
- **Security:** Centralized security in the service tier.
- **Flexibility:** Allows plug-and-play device integration.

Example Use Case:

In a **smart home system**:

- **Perception Layer:** Temperature sensors and motion detectors collect data.
- **Network Layer:** Wi-Fi sends data to the cloud.
- **Service Layer:** Web services process and store data.
- **Application Layer:** Mobile app shows current room temperature and allows control.
- **Business Layer:** Suggests turning on AC if temperature is high.

8. **Analyse** the significance and architecture of **Cloud of Things**. Compare it with **WoT** and **IoT**.

Cloud of Things (CoT) is the integration of **Cloud Computing** with the **Internet of Things (IoT)**. It enables **storage, processing, and analysis** of data generated by IoT devices using cloud services.

🧱 **Architecture of Cloud of Things:**

1. **Perception Layer (Device Layer):**
    - Consists of physical **sensors, actuators, smart devices** that collect data.
    - Devices may be embedded with RFID, GPS, cameras, etc.
2. **Network Layer:**
    - Transfers data from IoT devices to the cloud via **gateways and internet**.
    - Uses technologies like Wi-Fi, 4G/5G, Zigbee, etc.
3. **Cloud Middleware Layer:**
    - Provides **storage, computing power, and platforms**.
    - Examples: AWS IoT Core, Microsoft Azure IoT Hub, Google Cloud IoT.

- Manages **data processing, filtering, and analysis**.

4. **Application Layer:**
   - Offers **services to end-users** via web/mobile apps.
   - Examples: Smart city dashboards, home automation apps, fleet tracking systems.

🌟 **Significance of Cloud of Things:**

- **Scalability:** Handles large-scale IoT deployments.
- **Remote Access:** Data can be accessed from anywhere via cloud.
- **Big Data Processing:** Handles huge volumes of data efficiently.
- **Data Analytics:** Enables AI/ML-based insights.
- **Cost Efficiency:** Reduces need for on-premise infrastructure.

🔁 **Comparison: IoT vs WoT vs CoT**

Feature	IoT	WoT	CoT
Focus	Device connectivity	Web integration of devices	Cloud integration with IoT
Technologies	Sensors, MQTT, CoAP	HTTP, REST, JSON	Cloud storage, analytics, APIs
Accessibility	Local/limited	Web-accessible	Remote/cloud-based
Intelligence	Device-level	Web-level	Cloud-level (AI, ML)
Example	Smart meter sending data	Accessing sensor via REST API	Analyzing sensor data on cloud