**Hall Ticket Number:**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |

**IV/IV B.Tech (Regular/Supplementary) DEGREE EXAMINATION**

**November, 2022** **Information Technology**

**Seventh Semester** **Internet Of Things**

**Time:** Three Hours **Maximum:** 50 Marks

*Answer Question No.1 compulsorily.* (10X1 = 10 Marks)

*Answer ONE question from each unit.* (4X10=40 Marks)

| | | | | | |
|---|---|---|---|---|---|
| 1. | a) | Define IOT. | CO1 | L1 | |
| | b) | Write any three advantages of IOT technology | CO1 | L2 | |
| | c) | List out the functional blocks in IOT system | CO1 | L3 | |
| | d) | What is a sensor? Give examples | CO2 | L1 | |
| | e) | What is an actuator? Give examples | CO2 | L2 | |
| | f) | List out the communication protocols in IOT | CO2 | L1 | |
| | g) | What does VNF stand for? | CO3 | L1 | |
| | h) | Expand GPIO. | CO3 | L3 | |
| | i) | Draw Djanho Architecture. | CO4 | L1 | |
| | j) | Mention the sensors used in home automation system. | CO4 | L2 | |

**Unit –I**

| | | | | | |
|---|---|---|---|---|---|
| 2. | a) | Briefly explain about various IOT enabling technologies. | CO1 | L1 | 5M |
| | b) | With neat diagram explain about Physical Design of IOT | CO1 | L2 | 5M |
| | | **(OR)** | | | |
| 3. | a) | List different IOT levels and write any two deployment templates with diagram | CO1 | L1 | 5M |
| | b) | Explain about IOT Protocols Layers with neat diagram | CO1 | L3 | 5M |

**Unit –II**

| | | | | | |
|---|---|---|---|---|---|
| 4. | a) | Demonstrate the sensing and working of LDR Sensor using Arduino | CO2 | L3 | 5M |
| | b) | Using Arduino with PIR sensor, how to detect motion of objects? Explain | CO2 | L3 | 5M |
| | | **(OR)** | | | |
| 5. | a) | Explain the features of Raspberry Pi | CO2 | L2 | 5M |
| | b) | Explain IOT Communication Models and APIs with diagram | CO2 | L4 | 5M |

**Unit –III**

| | | | | | |
|---|---|---|---|---|---|
| 6. | a) | Define M2M? Differentiate between M2M and IOT | CO3 | L2 | 5M |
| | b) | For Home automation case study, explain the steps for IOT design Methodology in detail | CO3 | L1 | 5M |
| | | **(OR)** | | | |
| 7. | a) | Define SDN. Differentiate between SDN and NFV | CO3 | L3 | 5M |
| | b) | What do you mean by NVF? Explain with an Example | CO3 | L1 | 5M |

**Unit –IV**

| | | | | | |
|---|---|---|---|---|---|
| 8. | a) | Explain about significance of cloud service in IOT. | CO4 | L3 | 5M |
| | b) | Discuss about WAMP protocol. | CO4 | L3 | 5M |
| | | **(OR)** | | | |
| 9. | a) | Explain an IOT application using Amazon Web Services. | CO4 | L1 | 5M |
| | b) | Write short notes on (i) Forest Fire Detection (ii) Smart Irrigation. | CO4 | L3 | 5M |

**a) Define IOT.**

**Ans**: The Internet of Things (IoT) describes the network of physical objects—"things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet.

**b) Write any three advantages of IOT technology**

Ans: 1.Minimize human effort 2. Save time 3. Enhanced data collection 4. Improved security

**c) List out the functional blocks in IOT system**

Ans: 1.Device 2. Communication 3. Services 4. Management 4. Security 5. Application

**d) What is a sensor? Give examples**

Ans: A sensor is a device that detects and responds to some type of input from the physical environment. The input can be light, heat, motion, moisture or pressure. Example PIR, Ultrasonic, DHT11, LDR etc..

**e) What is an actuator? Give examples**

Ans: In simple terms, an actuator operates in the reverse direction of a sensor. It takes an electrical input and turns it into physical action. Example Mortor,LED,Buzzer etc..

**f) List out the communication protocols in IOT**

Ans: 1.MQTT 2.XMPP 3.COAP Etc…

**g) What does VNF stand for?**

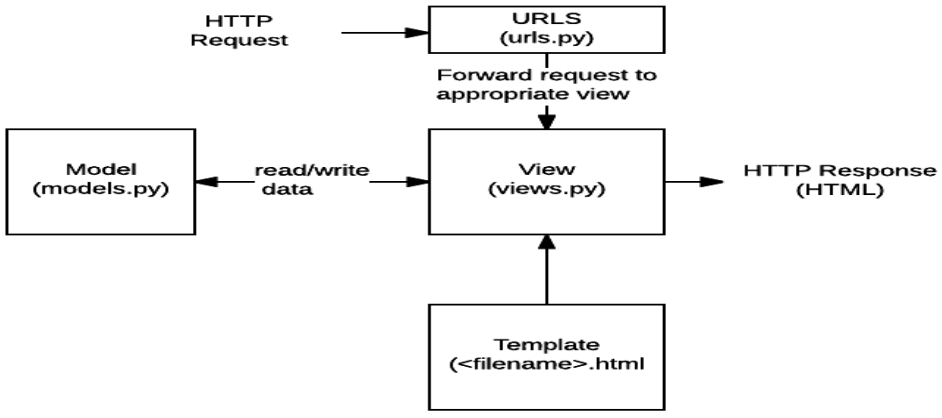
**Ans:** Network function Virtualization.

**h) Expand GPIO.**

Ans: General Purpose Input and Output.

**i)Draw Django Architecture.**

**Ans:**




**j) Mention the sensors used in home automation system.**

- Temperature Sensors & Smart Thermostats.
- Light Sensors
- Motion Sensors
- Water Leak/Freeze Sensors
- Window and Door Sensors

- Video Doorbell.
- Weather Sensors
- Smart Smoke and CO Sensors.

**2. a) Briefly explain about various IOT enabling technologies. CO1 L1 5M**

IoT is enabled by several technologies including Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Embedded Systems, Security Protocols and architectures, Communication Protocols, Web Services, Mobile internet and semantic search engines.

**1) Wireless Sensor Network (WSN):** Comprises of distributed devices with sensors which are used to monitor the environmental and physical conditions. Zig Bee is one of the most popular wireless technologies used by WSNs. WSNs used in IoT systems are described as follows:

- Weather Monitoring System: in which nodes collect temp, humidity and other data, which is aggregated and analysed.
- Indoor air quality monitoring systems: to collect data on the indoor air quality and concentration of various gases.
- Soil Moisture Monitoring Systems: to monitor soil moisture at various locations.
- Surveillance Systems: use WSNs for collecting surveillance data(motion data detection).
- Smart Grids : use WSNs for monitoring grids at various points.
- Structural Health Monitoring Systems: Use WSNs to monitor the health of structures(building, bridges) by collecting vibrations from sensor nodes deployed at various points in the structure.

**2) Cloud Computing:** Services are offered to users in different forms.

- Infrastructure-as-a-service(IaaS):provides users the ability to provision computing and storage resources. These resources are provided to the users as a virtual machine instances and virtual storage.
- Platform-as-a-Service(PaaS): provides users the ability to develop and deploy application in cloud using the development tools, APIs, software libraries and services provided by the cloud service provider.
- Software-as-a-Service(SaaS): provides the user a complete software application or the user interface to the application itself.

**3) Big Data Analytics:** Some examples of big data generated by IoT are Sensor data generated by IoT systems.

- Machine sensor data collected from sensors established in industrial and energy systems.
- Health and fitness data generated IoT devices.
- Data generated by IoT systems for location and tracking vehicles.
- Data generated by retail inventory monitoring systems.

**4) Communication Protocols:** form the back-bone of IoT systems and enable network connectivity and coupling to applications.

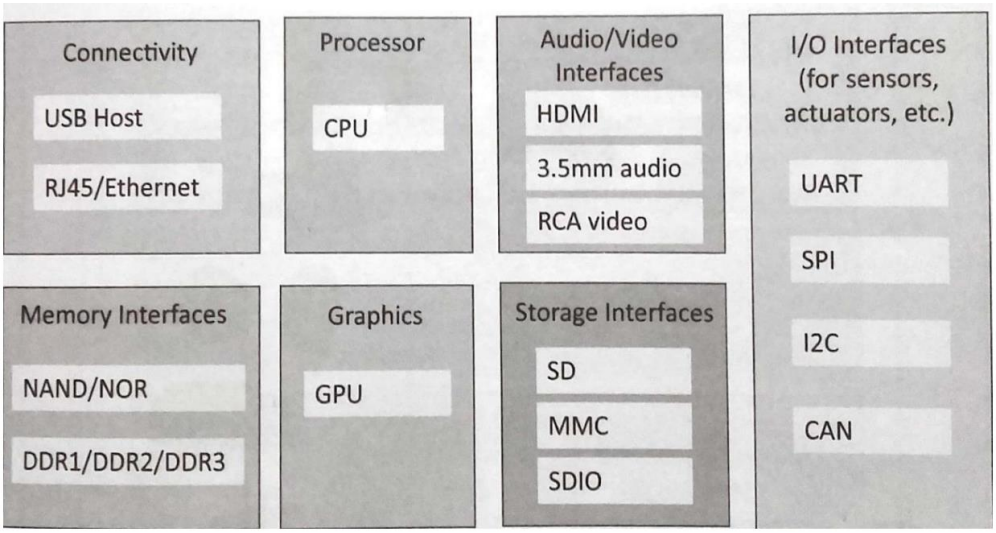
- Allow devices to exchange data over network.
- Define the exchange formats, data encoding addressing schemes for device and routing of packets from source to destination.
- It includes sequence control, flow control and retransmission of lost packets.

**5) Embedded Systems:** is a computer system that has computer hardware and software embedded to perform specific tasks. Embedded System range from low cost miniaturized devices such as digital watches to devices such as digital cameras, POS terminals, vending machines, appliances etc.,

**b) With neat diagram explain about Physical Design of IOT CO1 L2 5M**

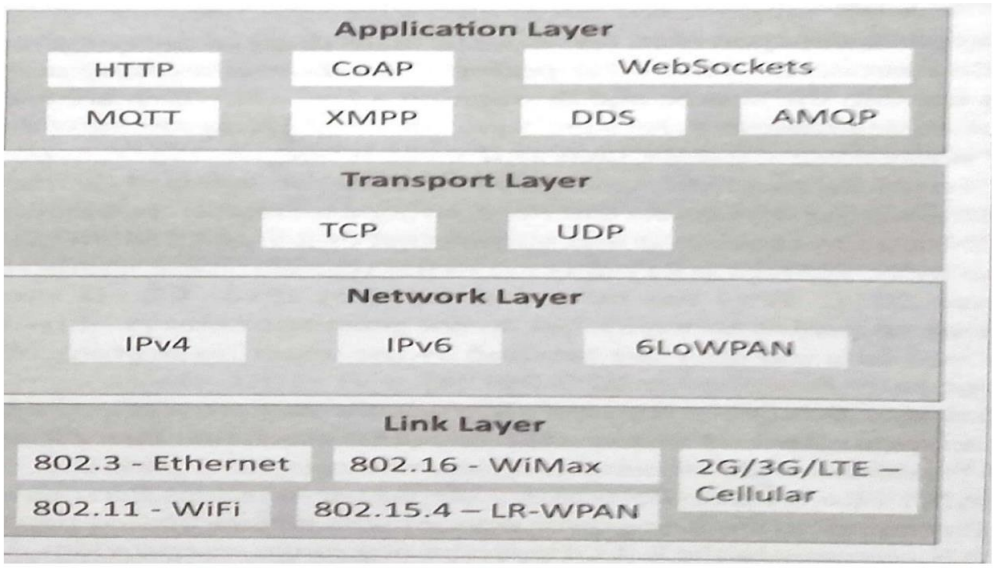
Physical Design Of IoT

1) Things in IoT:




The things in IoT refers to IoT devices which have unique identities and perform remote sensing, actuating and monitoring capabilities. IoT devices can exchange dat with other connected devices applications. It collects data from other devices and process data either locally or remotely. An IoT device may consist of several interfaces for communication to other devices both wired and wireless. These includes (i) I/O interfaces for sensors, (ii) Interfaces for internet connectivity (iii) memory and storage interfaces and (iv) audio/videointerfaces.

2) IoT Protocols:




a) **Link Layer:** Protocols determine how data is physically sent over the network‘s physical layer or medium. Local network connect to which host is attached. Hosts on the same link exchange data packets over the link layer using link layer protocols. Link layer determines how packets are coded and signaled by the h/w device over the medium to which the host is attached.

**Protocols:**

- 802.3-Ethernet: IEEE802.3 is collection of wired Ethernet standards for the link layer. Eg: 802.3 uses co-axial cable; 802.3i uses copper twisted pair connection; 802.3j uses fiber optic connection; 802.3ae uses Ethernet overfiber.
- 802.11-WiFi: IEEE802.11 is a collection of wireless LAN(WLAN) communication standards including extensive description of link layer. Eg: 802.11a operates in 5GHz band, 802.11b and 802.11g operates in 2.4GHz band, 802.11n operates in 2.4/5GHz band, 802.11ac operates in 5GHz band, 802.11ad operates in 60Ghzband.
- 802.16 - WiMax: IEEE802.16 is a collection of wireless broadband standards including exclusive description of link layer. WiMax provide data rates from 1.5 Mb/s to 1Gb/s.
- 802.15.4-LR-WPAN: IEEE802.15.4 is a collection of standards for low-rate wireless personal area network (LR-WPAN). Basis for high level communication protocols such as ZigBee. Provides data rate from 40kb/s to250kb/s.
- 2G/3G/4G-Mobile Communication: Data rates from 9.6kb/s(2G) to up to100Mb/s(4G).

b) **Network/Internet Layer:** Responsible for sending IP datagrams from source n/w to destination n/w. Performs the host addressing and packet routing. Datagrams contains source and destination address.

**Protocols:**

- IPv4: Internet Protocol version4 is used to identify the devices on a n/w using a hierarchical addressing scheme. 32-bit address. Allows total of $2^{32}$ addresses.
- IPv6: Internet Protocol version6 uses 128-bit address scheme and allows $2^{128}$ addresses.
- 6LOWPAN:(IPv6 over Low power Wireless Personal Area Network) operates in 2.4 GHz frequency range and data transfer 250 kb/s.

c) **Transport Layer:** Provides end-to-end message transfer capability independent of the underlying n/w. Set up on connection with ACK as in TCP and without ACK as in UDP. Provides functions such as error control, segmentation, flow control and congestion control.

**Protocols:**

- TCP: Transmission Control Protocol used by web browsers (along with HTTP and HTTPS), email (along with SMTP, FTP). Connection oriented and stateless protocol. IP Protocol deals with sending packets, TCP ensures reliable transmission of protocols in order. Avoids n/w congestion and congestion collapse.
- UDP: User Datagram Protocol is connectionless protocol. Useful in time sensitive applications, very small data units to exchange. Transaction oriented and stateless protocol. Does not provide guaranteed delivery.

d) **Application Layer:** Defines how the applications interface with lower layer protocols to send data over the n/w. Enables process-to-process communication using ports.

**Protocols:**

- HTTP: Hyper Text Transfer Protocol that forms foundation of WWW. Follow request-response model Stateless protocol.
- CoAP: Constrained Application Protocol for machine-to-machine(M2M) applications with constrained devices, constrained environment and constrained n/w. Uses client- server architecture.
- WebSocket: allows full duplex communication over a single socket connection.
- MQTT: Message Queue Telemetry Transport is light weight messaging protocol based on publish-subscribe model. Uses client server architecture. Well suited for constrained environment.
- XMPP: Extensible Message and Presence Protocol for real time communication and streaming XML data between network entities. Support client-server and server-server communication.
- DDS: Data Distribution Service is data centric middleware standards for device-to-device or machine-to-machine communication. Uses publish-subscribe model.
- AMQP: Advanced Message Queuing Protocol is open application layer protocol for business messaging. Supports both point-to-point and publish-subscribe model.

**(OR)**

**3. a) List different IOT levels and write any two deployment templates with diagram. CO1 L1 5M**

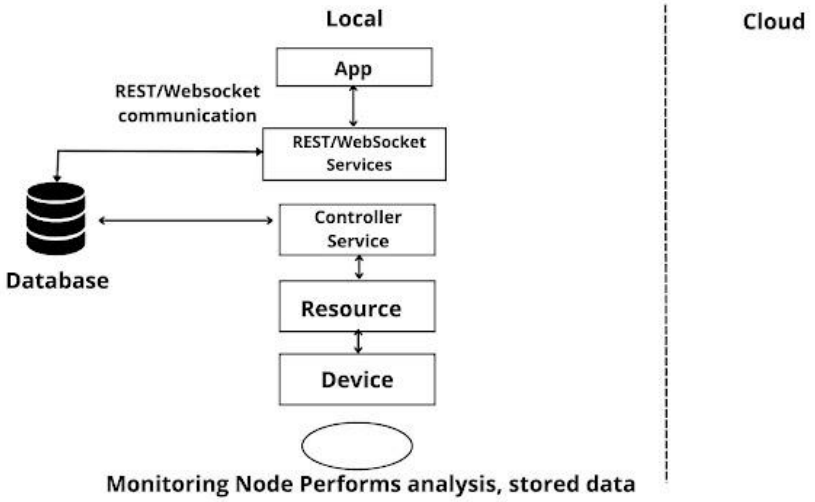
**Different IOT levels:**

There are totally 6 different types of IOT levels.

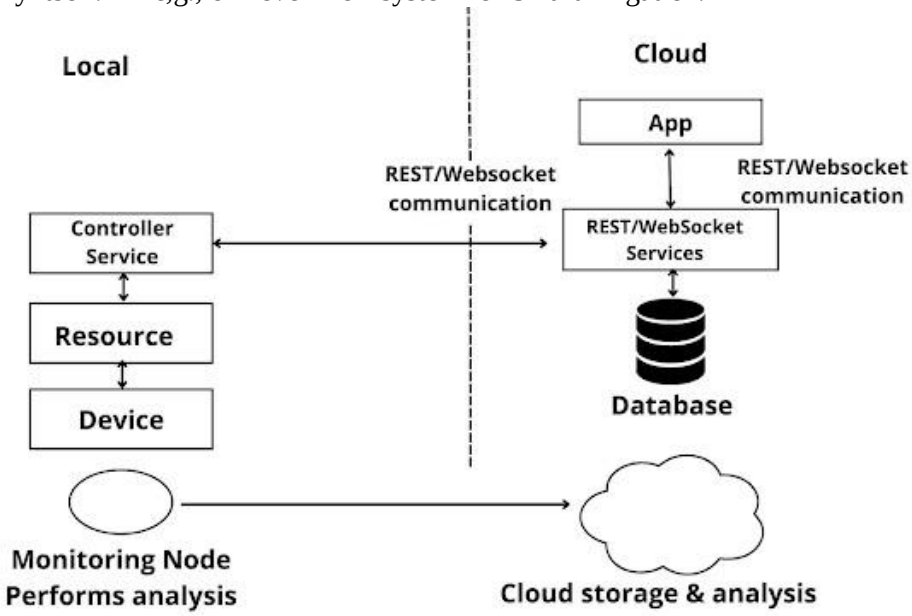
- IOT Level 1
- IOT Level 2
- IOT Level 3
- IOT Level 4
- IOT Level 5
- IOT Level 6

**[Explain any 2]**

**1) IoT Level1:** System has a single node that performs sensing and/or actuation, stores data, performs analysis and host the application as shown in fig. Suitable for modeling low cost and low complexity solutions where the data involved is not big and analysis requirement are not computationally intensive. An e.g., of IoT Level1 is Home automation.



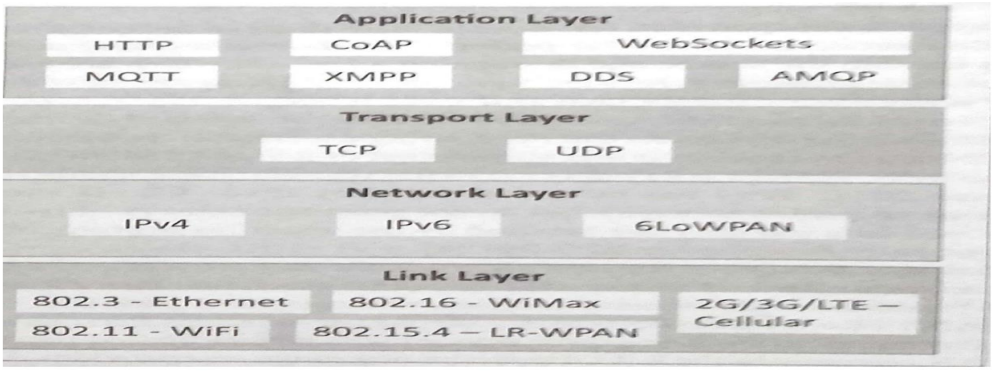

**2) IoT Level2:** has a single node that performs sensing and/or actuating and local analysis as shown in fig. Data is stored in cloud and application is usually cloud based. Level2 IoT systems are suitable for solutions where data are involved is big, however, the primary analysis requirement is not computationally intensive and can be done locally itself. An e,g., of Level2 IoT system for Smart Irrigation.




**CO1 L3 5M**

**b) Explain about IOT Protocols Layers with neat diagram**

3) IoT Protocols:



a) **Link Layer:** Protocols determine how data is physically sent over the network's physical layer or medium. Local network connect to which host is attached. Hosts on the same link exchange data packets over the link layer using link layer protocols. Link layer determines how packets are coded and signaled by the h/w device over the medium to which the host is attached.

**Protocols:**

- 802.3-Ethernet: IEEE802.3 is collection of wired Ethernet standards for the link layer. Eg: 802.3 uses co-axial cable; 802.3i uses copper twisted pair connection; 802.3j uses fiber optic connection; 802.3ae uses Ethernet overfiber.
- 802.11-WiFi: IEEE802.11 is a collection of wireless LAN(WLAN) communication standards including extensive description of link layer. Eg: 802.11a operates in 5GHz band, 802.11b and 802.11g operates in 2.4GHz band, 802.11n operates in 2.4/5GHz band, 802.11ac operates in 5GHz band, 802.11ad operates in 60Ghzband.
- 802.16 - WiMax: IEEE802.16 is a collection of wireless broadband standards including exclusive description of link layer. WiMax provide data rates from 1.5 Mb/s to 1Gb/s.
- 802.15.4-LR-WPAN: IEEE802.15.4 is a collection of standards for low-rate wireless personal area network (LR-WPAN). Basis for high level communication protocols such as ZigBee. Provides data rate from 40kb/s to250kb/s.
- 2G/3G/4G-Mobile Communication: Data rates from 9.6kb/s(2G) to up to100Mb/s(4G).

b) **Network/Internet Layer:** Responsible for sending IP datagrams from source n/w to destination n/w. Performs the host addressing and packet routing. Datagrams contains source and destination address.

**Protocols:**

- IPv4: Internet Protocol version4 is used to identify the devices on a n/w using a hierarchical addressing scheme. 32-bit address. Allows total of $2^{32}$ addresses.
- IPv6: Internet Protocol version6 uses 128-bit address scheme and allows $2^{128}$ addresses.
- 6LOWPAN:(IPv6 over Low power Wireless Personal Area Network) operates in 2.4 GHz frequency range and data transfer 250 kb/s.

c) **Transport Layer:** Provides end-to-end message transfer capability independent of the underlying n/w. Set up on connection with ACK as in TCP and without ACK as in UDP. Provides functions such as error control, segmentation, flow control and congestion control.

**Protocols:**

- TCP: Transmission Control Protocol used by web browsers (along with HTTP and HTTPS), email (along with SMTP, FTP). Connection oriented and stateless protocol. IP Protocol deals with sending packets, TCP ensures reliable transmission of protocols in order. Avoids n/w congestion and congestion collapse.
- UDP: User Datagram Protocol is connectionless protocol. Useful in time sensitive applications, very small data units to exchange. Transaction oriented and stateless protocol. Does not provide guaranteed delivery.

d) **Application Layer:** Defines how the applications interface with lower layer protocols to send data over the n/w. Enables process-to-process communication using ports.

**Protocols:**

- HTTP: Hyper Text Transfer Protocol that forms foundation of WWW. Follow request-response model Stateless protocol.
- CoAP: Constrained Application Protocol for machine-to-machine(M2M) applications with constrained devices, constrained environment and constrained n/w. Uses client- server architecture.
- WebSocket: allows full duplex communication over a single socket connection.
- MQTT: Message Queue Telemetry Transport is light weight messaging protocol based on publish-subscribe model. Uses client server architecture. Well suited for constrained environment.
- XMPP: Extensible Message and Presence Protocol for real time communication and streaming XML data between network entities. Support client-server and server-server communication.
- DDS: Data Distribution Service is data centric middleware standards for device-to-device or machine-to-machine communication. Uses publish-subscribe model.
- AMQP: Advanced Message Queuing Protocol is open application layer protocol for business messaging. Supports both point-to-point and publish-subscribe model.

**4. a) Demonstrate the sensing and working of LDR Sensor using Arduino CO2 L3 5M**

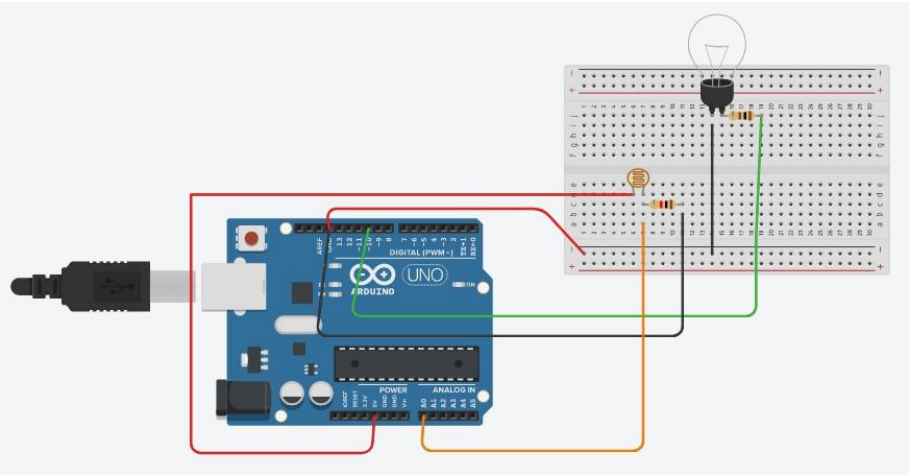
**Components:**

1. Arduino Uno-R3 board
2. Bread board small
3. 2 Resistors
4. Light Bulb
5. Photoresistor (LDR)
6. Connecting Wires

**Description:**

1. **Photoresistor:** Photoresistors, also known as light dependent resistors (LDR), are light sensitive devices most often used to indicate the presence or absence of light, or to measure the light intensity.

**Hardware Design:**



**Source Code:**

```
int BULB = 10;
int ldr = A0;
void setup()
{
        Serial.begin(9600);
        pinMode(BULB, OUTPUT);
        pinMode(ldr, INPUT);
}
void loop()
{
        int ldrStatus = analogRead(ldr);
        Serial.println(ldrStatus);
        if (ldrStatus >300) {
           digitalWrite(BULB, HIGH);
        }
        else {
           digitalWrite(BULB, LOW);
        }
}
```

**Output:**

**b) Using Arduino with PIR sensor, how to detect motion of objects? Explain CO2 L3 5M**

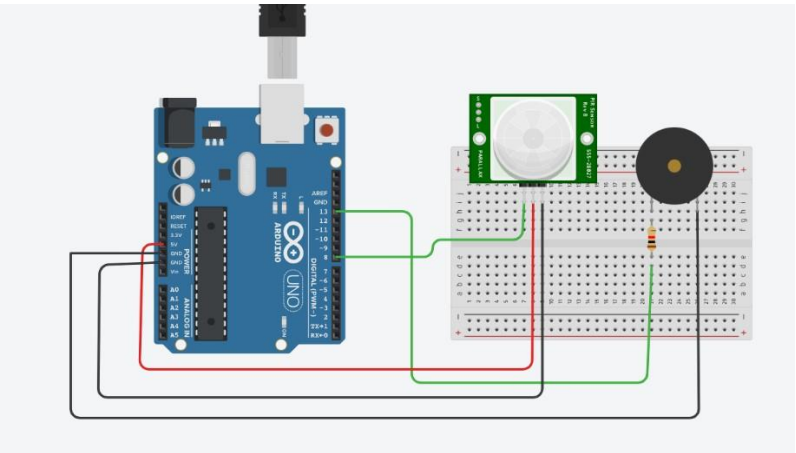
**Components:**

1) Arduino Uno-R3 board
2)Breadboard small
3)Buzzer
4)Resistor
5)Wires
6)PIR sensor

**Description:**

PIR sensor: A passive infrared sensor (PIR sensor) is an electronic sensor that measures infrared (IR) light radiating from objects in its field of view. They are most often used in PIR-based motion detectors. PIR sensors are commonly used in security alarms and automatic lighting applications.
PIR sensors detect general movement, but do not give information on who or what moved. For that purpose, an imaging IR sensor is required.

**Hardware Design:**



**Source Code:**

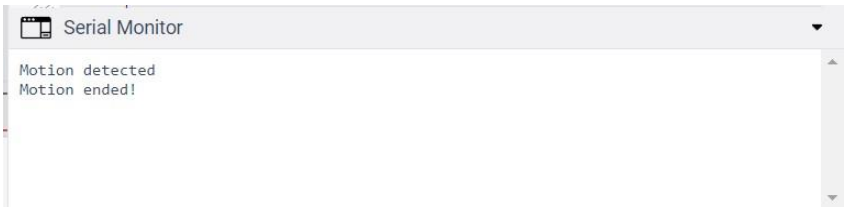
```
int buzzer=13;
int inputPin=8;
int pirState=LOW;
int val=0;
void setup()
{
        pinMode(buzzer, OUTPUT);
        pinMode(inputPin,INPUT);
        Serial.begin(9600);
}

void loop()
{
        val=digitalRead(inputPin);
        if(val==HIGH)
        {
           digitalWrite(buzzer,HIGH);
           if(pirState==LOW)
           {
                Serial.println("Motion detected");
                pirState=HIGH;
           }
        }
        else
         {
```

```
            digitalWrite(buzzer,LOW);
            if(pirState==HIGH)
            {
                Serial.println("Motion ended!");
                pirState=LOW;

            }
          }
}
```

**Output:**




**(OR)**

**5. a) Explain the features of Raspberry Pi** **CO2 L2 5M**

The features of Raspberry Pi computers that make all these possible include:

**1. Central Processing Unit (CPU)**

Every computer has a Central Processing Unit, and so does the Raspberry Pi. It is the computer's brain and carries out instructions using logical and mathematical operations. Raspberry Pi makes use of the ARM11 series processor on its boards.

**2. HDMI port**

Raspberry Pi board has an HDMI or High Definition Multimedia Interface port that allows the device to have video options of the output from the computer displayed. An HDMI cable connects the Raspberry Pi to an HDTV. The supported versions include 1.3 and 1.3. It also comes with an RCA port for other display options.

**3. Graphic Processing Unit (GPU)**

This unit, GPU or Graphic Processing Unit, is another part of the Raspberry pi board. Its primary purpose is to hasten the speed of image calculations.

**4. Memory (RAM)**

Random Access Memory is a core part of a computer's processing system. It is where real-time information is stored for easy access. The initial Raspberry Pi had 256MB RAM. Over the years, developers gradually and significantly improved the size. Different Raspberry Pi models come with varying capacities. The model with the maximum capacity presently is the Raspberry Pi 4 with 8GB RAM space.

**5. Ethernet port**

The Ethernet port is a connectivity hardware feature available on B models of Raspberry Pi. The Ethernet port enables wired internet access to the minicomputer. Without it, software updates, web surfing, etc., would not be possible using the Raspberry Pi. The Ethernet port found on Raspberry computers uses the RJ45 Ethernet jack. With this component, Raspberry Pi can connect to routers and other devices.

**6. SD card slot**

Like most other regular computers, Raspberry Pi must have some sort of storage device. However, unlike conventional PCs, it does not come with a hard drive, nor does it come with a memory card. The Raspberry Pi board has a Secure Digital card or SD card slot where users must insert SD cards for the computer to function. The SD card functions like a hard drive as it contains the operating system necessary for turning the system on. It also serves to store data.

**7. General Purpose Input and Output (GPIO) pins**

These are upward projecting pins in a cluster on one side of the board. The oldest models of the Raspberry Pi had 26 pins, but most have 40 GPIO pins. These pins are pretty sensitive and should be handled carefully. They are essential parts of the Raspberry Pi device as they add to its diverse applications. GPIO pins are used to interact with other electronic circuits. They can read and control the electric signals from other boards or devices based on how the user programs them.

**8. LEDs**

These are a group of five light-emitting diodes. They signal the user on the present status of the Raspberry Pi unit. Their function covers:

PWR (Red): This functions solely to indicate power status. When the unit is on, it emits a red light and only goes off when the unit is switched off, or disconnected from the power source.

ACT (Green): This flashes to indicate any form of SD card activity.

LNK (Orange): LNK LED gives off an orange light to signify that active Ethernet connectivity has been established.

100 (Orange): This light comes on during Ethernet connection when the data speed reaches 100Mbps.

FDX (Orange): FDX light also comes during Ethernet connection. It shows that the connection is a full-duplex.

**9. USB ports**

Universal service bus (USB) ports are a principal part of Raspberry Pi. They allow the computer to connect to a keyboard, mouse, hard drives, etc. The first model of Raspberry Pi had only two USB 2.0 ports. Subsequent models increased this number to four. Raspberry Pi 4 and Pi 400, much newer models, come with a mix of USB 2.0 and USB 3.0 ports.

**10. Power source**

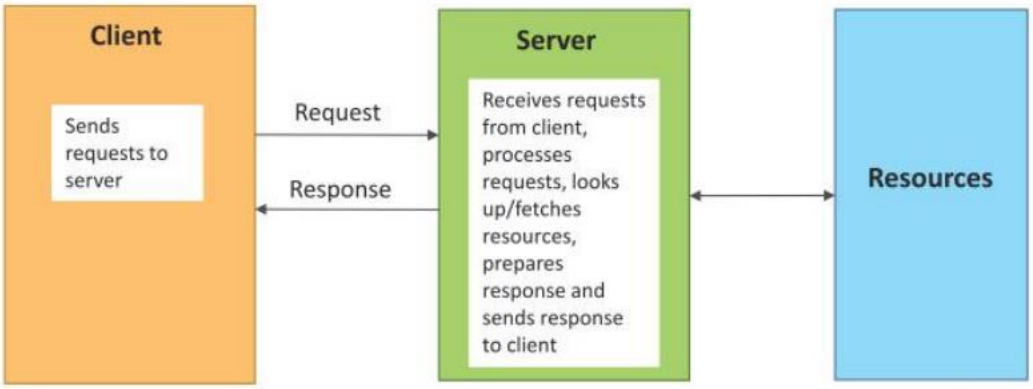
Raspberry Pi has a power source connector that typically uses a 5V micro USB power cable. The amount of electricity any Raspberry Pi consumes depends on what it's used for and the number of peripheral hardware devices connected.

**b) Explain IOT Communication Models and APIs with diagram CO2 L4 5M**

**<u>IoT Communication Models:</u>**

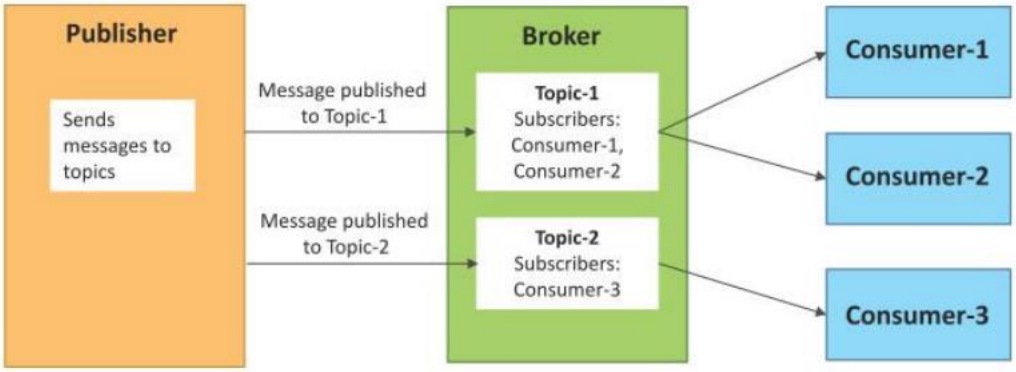
1) Request-Response 2) Publish-Subscribe 3) Push-Pull 4) Exclusive Pair

**1) Request-Response Model:**



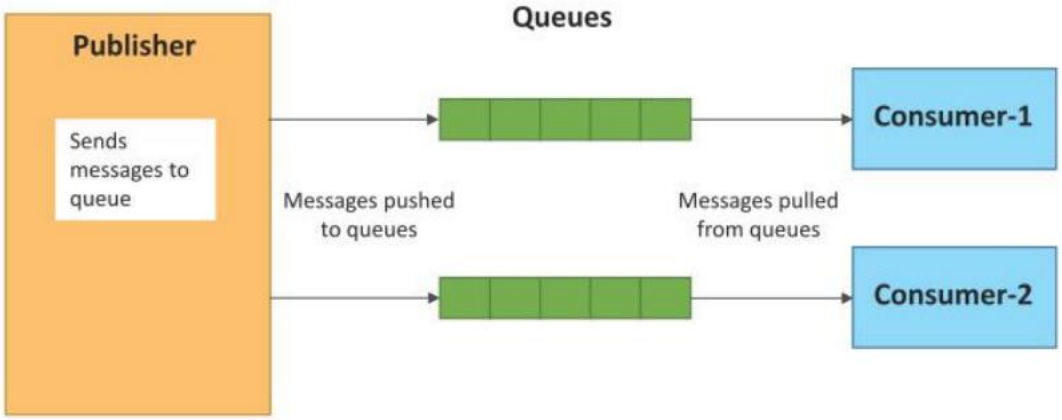

In which the client sends request to the server and the server replies to requests. Is a stateless communication model and each request-response pair is independent of others.

**2) Publish-Subscribe Model:**

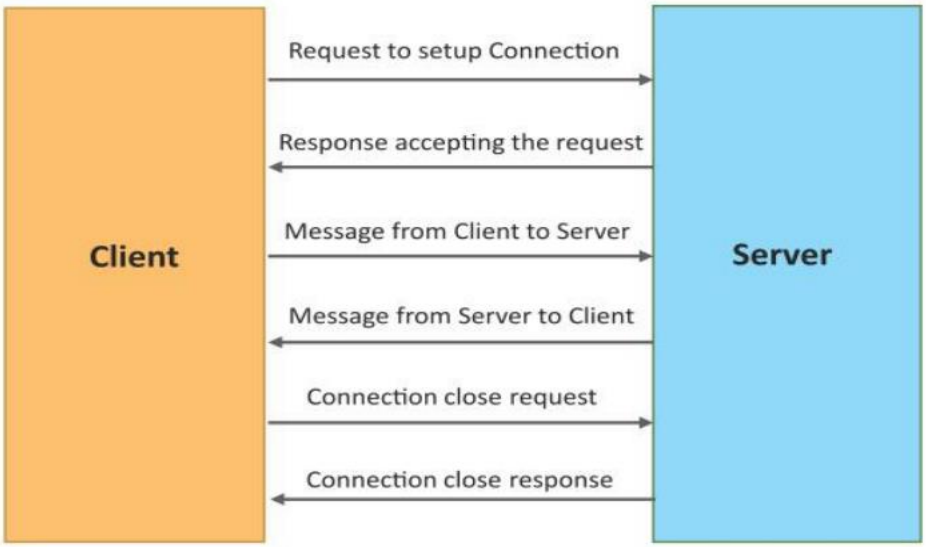



Involves publishers, brokers and consumers. Publishers are source of data. Publishers send data to the topics which are managed by the broker. Publishers are not aware of the consumers. Consumers subscribe to the topics which are managed by the broker. When the broker receives data for a topic from the publisher, it sends the data to all the subscribed consumers.

**3) Push-Pull Model:** in which data producers push data to queues and consumers pull data from the queues. Producers do not need to aware of the consumers. Queues help in decoupling the message between the producers and consumers.




**4) Exclusive Pair:** is bi-directional, fully duplex communication model that uses a persistent connection between the client and server. Once connection is set up it remains open until the client send a request to close the connection. Is a stateful communication model and server is aware of all the open connections.



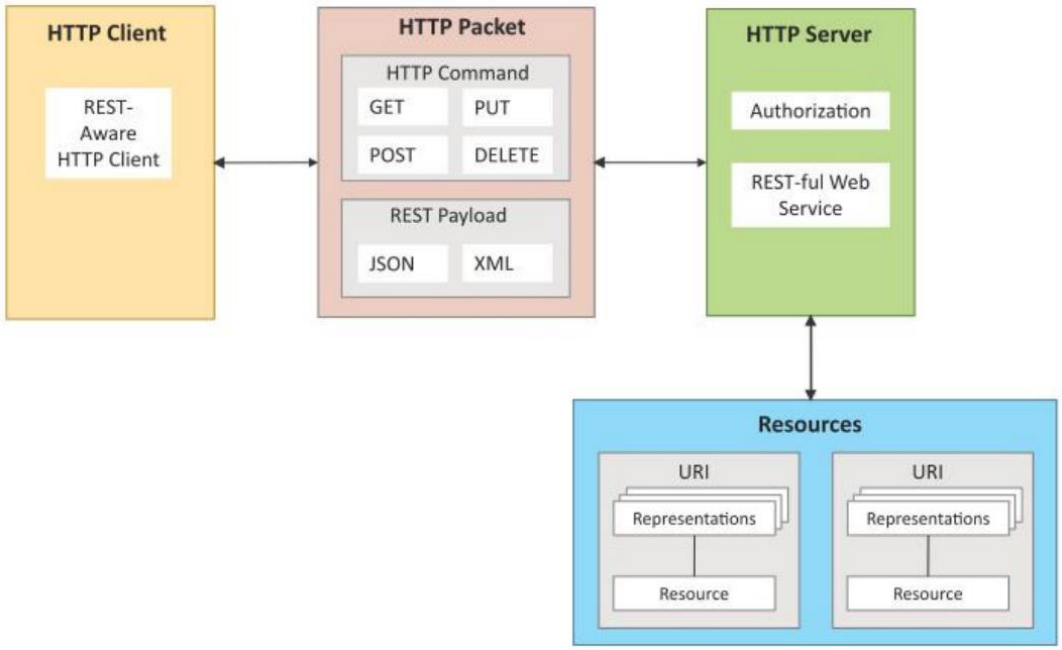
**IoT Communication APIs:**

**a) REST based communication APIs(Request-Response Based Model)**

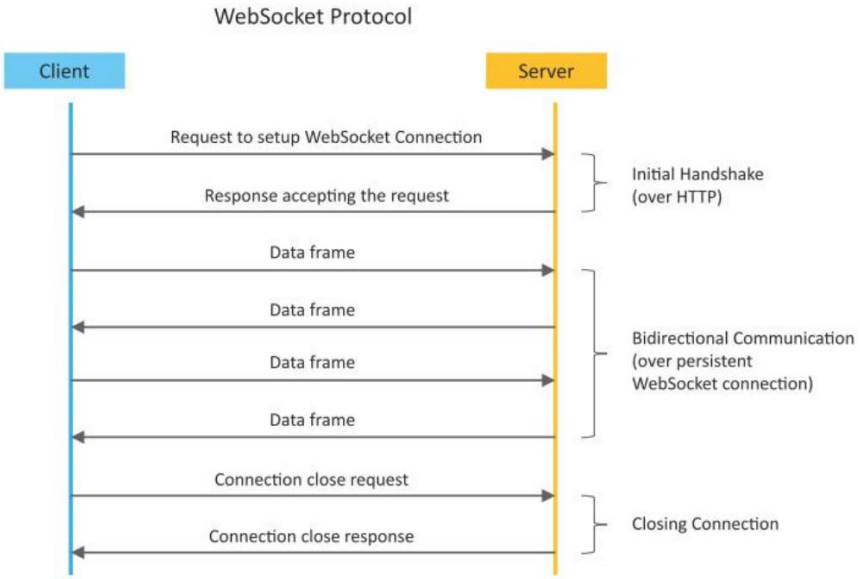
**b) WebSocket based Communication APIs(Exclusive Pair Based Model)**

**a) REST based communication APIs:** Representational State Transfer (REST) is a set of architectural principles by which we can design web services and web APIs that focus on a system's resources and have resource states are addressed and transferred.

**The REST architectural constraints:** Fig. shows communication between client server with REST APIs.




**WebSocket Based Communication APIs:** WebSocket APIs allow bi-directional, full duplex communication between clients and servers. WebSocket APIs follow the exclusive pair communication model.



**6. a) Define M2M? Differentiate between M2M and IOT** **CO3 L2 5M**

Differences between IoT and M2M:

1) Communication Protocols:
- Commonly uses M2M protocols include ZigBee, Bluetooth, Mod Bus, M-Bus, Wireless M-Bus etc.,
- In IoT uses HTTP, CoAP, WebSocket, MQTT, XMPP, DDS, AMQP etc.,

2) Machines in M2M Vs Things in IoT:
- Machines in M2M will be homogenous whereas Things in IoT will be heterogeneous.

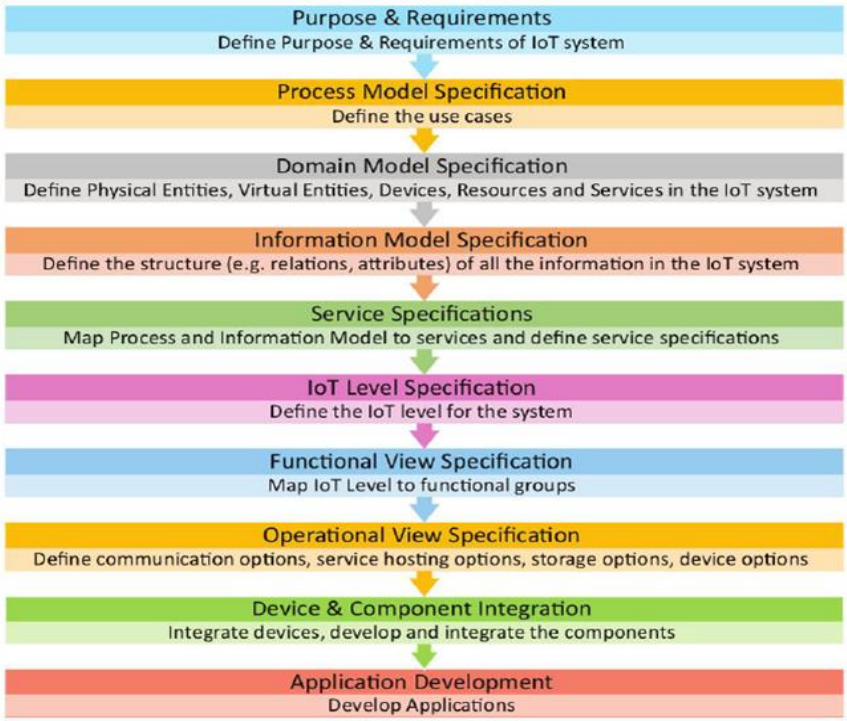
3) Hardware Vs Software Emphasis:
- the emphasis of M2M is more on hardware with embedded modules, the emphasis of IoT is more on software.

4) Data Collection &Analysis
- M2M data is collected in point solutions and often in on-premises storage infrastructure.
- The data in IoT is collected in the cloud (can be public, private or hybrid cloud).

5) Applications
- M2M data is collected in point solutions and can be accessed by on-premises applications such as diagnosis applications, service management applications, and on- premises enterprise applications.
- IoT data is collected in the cloud and can be accessed by cloud applications such as analytics applications, enterprise applications, remote diagnosis, and management applications, etc.

**b) For Home automation case study, explain the steps for IOT design Methodology in detail.** **CO3 L1 5M**

Ans:




Purpose: A home automation system that allows controlling the lights remotely using a web application

Behavior: Home automation system should support two modes: auto and manual

Auto: System measures the light level in the room and switches on the light when it is dark

Manual: Allows remotely switching lights on and off

System Management: System should provide remote monitoring and control functions

Data Analysis: System should perform local analysis of the data

Application Deployment:Application should be deployed locally, but should be accessible remotely

Security:Should provide basic security like user authentication

**(OR)**

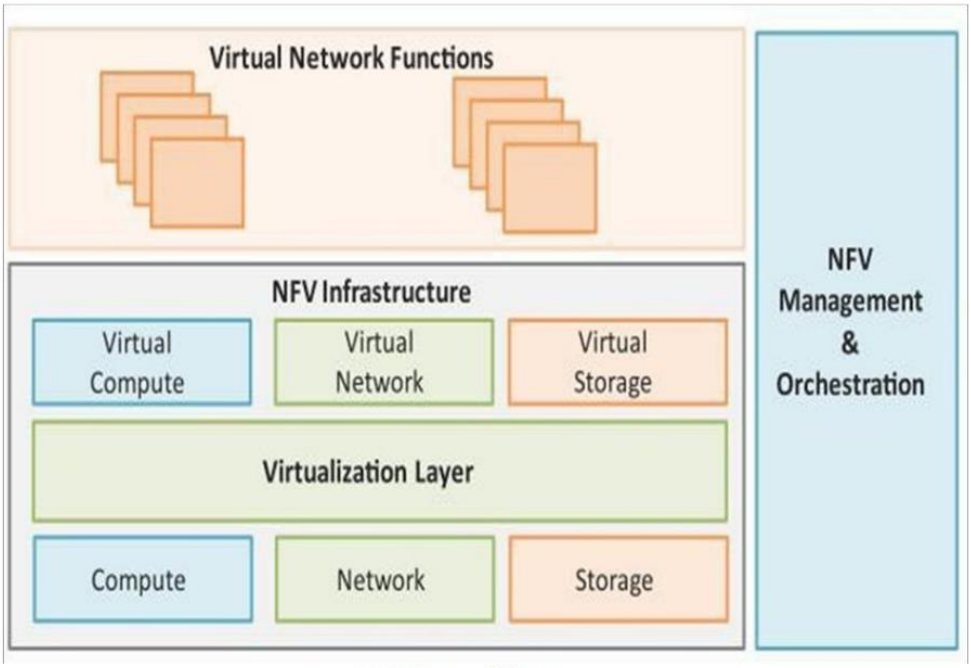
**7. a) Define SDN. Differentiate between SDN and NFV** **CO3** **L3** **5 M**

| **SDN** | **NFV** |
|---|---|
| SDN architecture mainly focuses on data centers. | NFV is targeted at service providers or operators. |
| SDN separates control plane and data forwarding plane by centralizing control and programmability of network. | NFV helps service providers or operators to virtualize functions like load balancing, routing, and policy management by transferring network functions from dedicated appliances to virtual servers. |
| SDN uses OpenFlow as a communication protocol. | There is no protocol determined yet for NFV. |
| SDN supports Open Networking Foundation. | NFV is driven by ETSI NFV Working group. |
| Various enterprise networking software and hardware vendors are initiative supporters of SDN. | Telecom service providers or operators are prime initiative supporters of NFV. |
| Corporate IT act as a Business initiator for SDN. | Service providers or operators act as a Business initiator for NFV. |
| SDN applications run on industry-standard servers or switches. | NFV applications run on industry-standard servers. |
| SDN reduces cost of network because now there is no need of expensive switches & routers. | NFV increases scalability and agility as well as speed up time-to-market as it dynamically allot hardware a level of capacity to network functions needed at a particular time. |
| Application of SDN:<br>• Networking<br>• Cloud orchestration | Application of NFV:<br>• Routers, firewalls, gateways<br>• WAN accelerators<br>• SLA assurance<br>• Video Servers<br>• Content Delivery Networks (CDN) |

**b) What do you mean by NVF? Explain with an Example** **CO3** **L1** **5 M**

**Network Function Virtualization(NFV)**

- Network Function Virtualization (NFV) is a technology that leverages virtualization to consolidate the heterogeneous network devices onto industry standard high-volume servers, switches and storage.
- NFV is complementary to SDN as NFV can provide the infrastructure on which SDN can run.




NFV Architecture

**Key elements of NFV:**

1) Virtualized Network Function(VNF):

VNF is a software implementation of a network function which is capable of running over the NFV Infrastructure (NFVI).

2) NFV Infrastructure(NFVI):

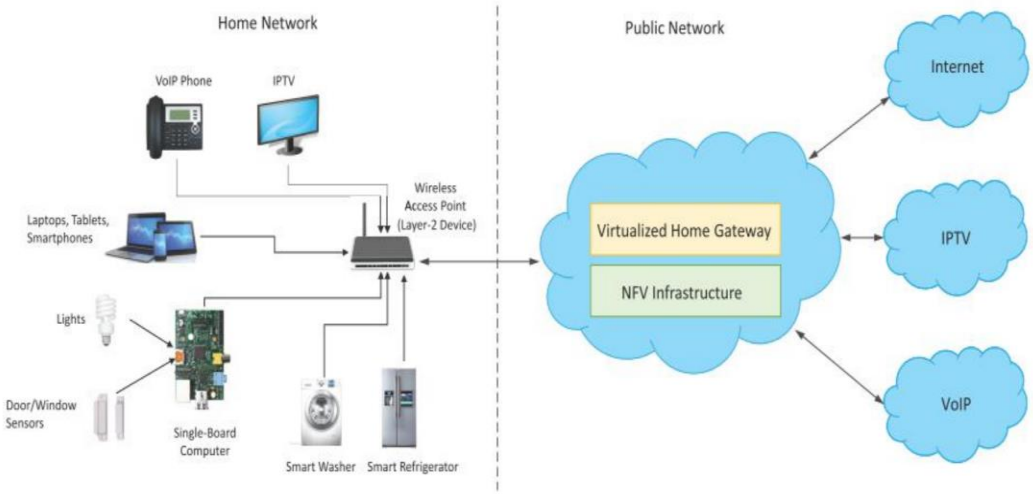
NFVI includes compute, network and storage resources that are virtualized.

3) NFV Management and Orchestration:

NFV Management and Orchestration focuses on all virtualization-specific management tasks and covers the orchestration and life-cycle management of physical and/or software resources that support the infrastructure virtualization, and the life-cycle management of VNFs.

**NFV Use Case**

NFV can be used to virtualize the Home Gateway. The NFV infrastructure in the cloud hosts a virtualized Home Gateway. The virtualized gateway provides private IP addresses to the devices in the home. The virtualized gateway also connects to network services such as VoIP and IPTV



## Unit –IV

**8. a) Explain about significance of cloud service in IOT. CO4 L3 5M**

Benefits And Functions of IoT Cloud:

1. IoT Cloud Computing provides many connectivity options, implying large network access. People use a wide range of devices to gain access to cloud computing resources: mobile devices, tablets, laptops. This is convenient for users but creates the problem of the need for network access points.

2. Developers can use IoT cloud computing on-demand. In other words, it is a web service accessed without special permission or any help. The only requirement is Internet access.

3. Based on the request, users can scale the service according to their needs. Fast and flexible means you can expand storage space, edit software settings, and work with the number of users. Due to this characteristic, it is possible to provide deep computing power and storage.

4. Cloud Computing implies the pooling of resources. It influences increased collaboration and builds close connections between users.

5. As the number of IoT devices and automation in use grows, security concerns emerge. Cloud solutions provide companies with reliable authentication and encryption protocols.

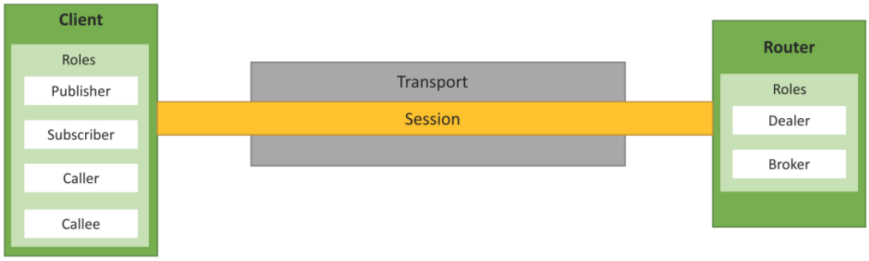
6. Finally, IoT cloud computing is convenient because you get exactly as much from the service as you pay. This means that costs vary depending on use: the provider measures your usage statistics. A growing network of objects with IP addresses is needed to connect to the Internet and exchange data between the components of the network.

**b) Discuss about WAMP protocol. CO4 L3 5M**

**WAMP : Web Application Messaging Protocol**

- Mainly used in cloud storage model for IoT & other messaging services
- WAMP is a routed protocol, with all components connecting to a WAMP Router, where the WAMP Router performs message routing between the component
- It is protocol for Web Socket (PUBSUB based protocol) : uses RPC Messaging Pattern
- Some Important Key Terminologies
  - Transport
  - Session
  - Clients (Publisher & Subscriber)
  - Router
  - Broker
  - Dealer
  - Application Code

Web Application Messaging Protocol (WAMP) is a sub-protocol of WebSocket which provides publish–subscribe and remote procedure call (RPC) messaging patterns.




• **Transport**: Transport is a channel that connects two peers.
• **Session**: Session is a conversation between two peers that runs over a transport.

- **Client**: Clients are peers that can have one or more roles.
  - In the publish–subscribe model, the Client can have the following roles:
    **Publisher**: Publisher publishes events (including payload) to the topic maintained by the Broker.
    **Subscriber**: Subscriber subscribes to the topics and receives the events including the payload.
  - In the RPC model, the Client can have the following roles:
    **Caller**: Caller issues calls to the remote procedures along with call arguments.
    **Callee**: Callee executes the procedures to which the calls are issued by the Caller and returns the results to the Caller.
- **Router**: Routers are peers that perform generic call and event routing.
  - In the publish–subscribe model, the Router has the role of a Broker.
    **Broker**: Broker acts as a Router and routes messages published to a topic to all the subscribers subscribed to the topic.
  - In the RPC model, the Router has the role of a Dealer.
    **Dealer**: Dealer acts a router and routes RPC calls from the Caller to the Callee and routes results from the Callee to the Caller.
- **Application code**: Application code runs on the Clients (Publisher, Subscriber, Callee or Caller).

**(OR)**

**9.a) Explain an IOT application using Amazon Web Services. CO4 L1 5M**

**Ans:** AWS IoT (Amazon internet of things) is an Amazon Web Services platform that collects and analyzes data from internet-connected devices and sensors and connects that data to AWS cloud applications. AWS IoT can collect data from billions of devices and connect them to endpoints for other AWS tools and services, allowing a developer to tie that data into an application.

An AWS user accesses AWS IoT with the AWS Management Console, software development kits (SDKs) or the AWS Command Line Interface. An application accesses the service through AWS SDKs. AWS IoT APIs are divided into the control plane, which includes service configuration, device registration and logging; and the data plane, which includes data ingestion.

The IoT service includes a Rules Engine feature that enables an AWS customer to continuously ingest, filter, process and route data that is streamed from connected devices. A developer can configure rules in a syntax that's similar to SQL to transform and organize data. The feature also allows the user to configure how data interacts with other big data and automation services, such as AWS Lambda, Amazon Kinesis, Amazon Machine Learning, Amazon DynamoDB and Amazon Elasticsearch Service. Each rule consists of an SQL statement and an action list that defines and executes the rule using an editable JSON-based schema.

Device Shadows is an optional rule that enables an application to query data from devices and send commands through REST APIs. Device Shadows provide a uniform interface for all devices, regardless of limitations to connectivity, bandwidth, computing ability or power.

The optional Device Registry feature lets a developer register and track devices that are connected to the service, including metadata for each device such as model numbers and associated certificates. A developer can define a Thing Type to manage similar devices according to common characteristics. Each Thing associated with a Thing Type can have up to 50 attributes and three searchable attributes. A developer can also opt to have applications communicate directly to the IoT service.

**b) Write short notes on (i) Forest Fire Detection (ii) Smart Irrigation. CO4 L3 5M**

**Ans: Forest Fire Detection:**

Forest fires (wildfires) are common hazards in forests, particularly in remote or unmanaged areas. It is possible to detect forest fires, elevated CO2, and temperature levels using Internet of Things (IoT) sensors. You can deploy IoT, satellite and solar sensors in remote areas without the need for internet, cellular/mobile or mains power.

**Remote IoT sensor networks:**

CO2 and temperature IoT sensors are battery and solar-powered, using a combination of LoRaWAN and satellite communications to provide coverage even in remote areas.

A LoRaWAN gateway can provide network coverage for up to 15KM outdoors, while satellite backhaul ensures traffic can be sent back to your monitoring system without requiring mobile or internet coverage.

1. LoRaWAN Satellite gateway is deployed, preferably in an elevated location.
The gateway provides LoRaWAN coverage for up to 15KM or more outdoors, providing low-power wireless access to the network of sensors.
2. CO2/Temperature sensors are deployed throughout the coverage area. CO2 levels can be monitored every 15 minutes (or less), along with temperature, battery status, etc.
3. Upon receiving sensor data, the gateway transmits the data back via satellite (in an optimised manner) to a cloud platform or dashboard.
4. Triggers and alerts can be configured.

**Smart Irrigation:**

The Smart irrigation System has wide scope to automate the complete irrigation system. Here we are building a IoT based Irrigation System using ESP8266 NodeMCU Module and DHT11 Sensor. It will not only automatically irrigate the water based on the moisture level in the soil but also send the Data to ThingSpeak Server to keep track of the land condition. The System will consist a water pump which will be used to sprinkle water on the land depending upon the land environmental condition such as Moisture, Temperature and Humidity.

Before starting, it is important to note that the different crops require different Soil Moisture, Temperature and Humidity Condition. We are using such a crop which will require soil moisture of about 50-55%. So when the soil loses its moisture to less than 50% then Motor pump will turn on automatically to sprinkle the water and it will continue to sprinkle the water until the moisture goes upto 55% and after that the pump will be turned off. The sensor data will be sent to ThingSpeak Server in defined interval of time so that it can be monitored from anywhere in the world.

**Components Required**

- NodeMCU ESP8266
- Soil Moisture Sensor Module
- Water Pump Module
- Relay Module
- DHT11
- Connecting Wires

**Schema Prepared by: P.RatnaPrakash, Asst.Prof, IT**

**B.Krishnaiah, Asst.Prof,IT** **HOD, IT**