INDIRA GANDHI DELHI TECHNICAL UNIVERSITY FOR WOMEN

IoT & its applications in AI

ASSIGNMENT

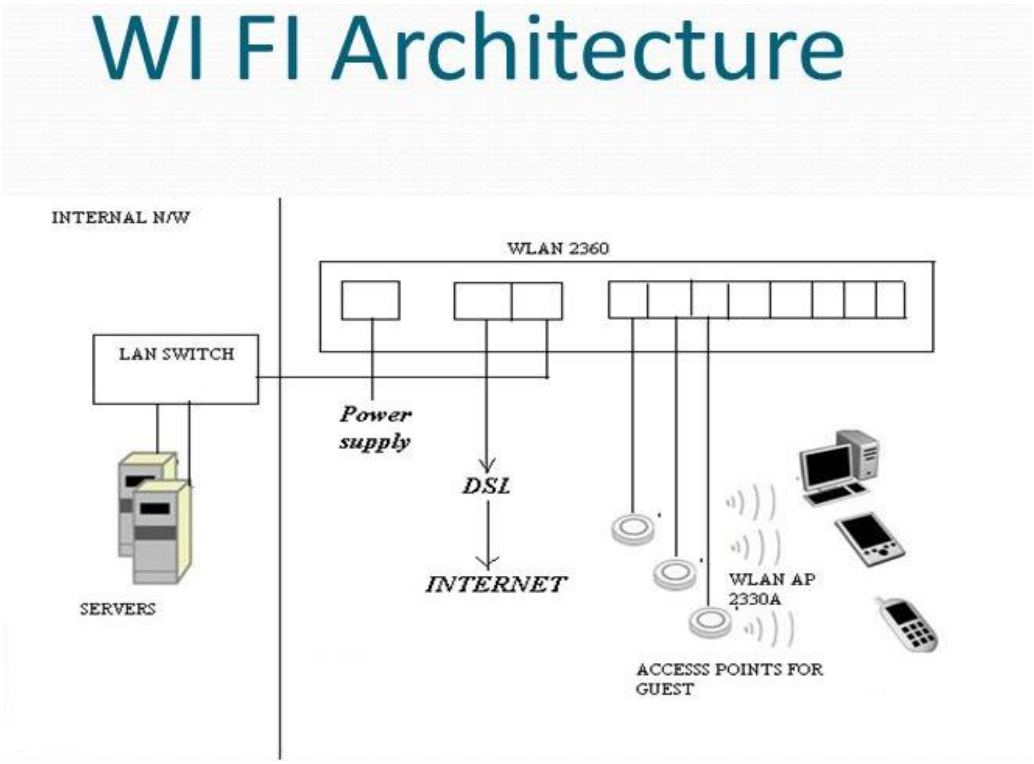
Submitted By:
Richansi Chauhan
Roll No.- 01402102019
M.Tech. CSE (2nd Semester)

Submitted To:
Prof. SRN Reddy

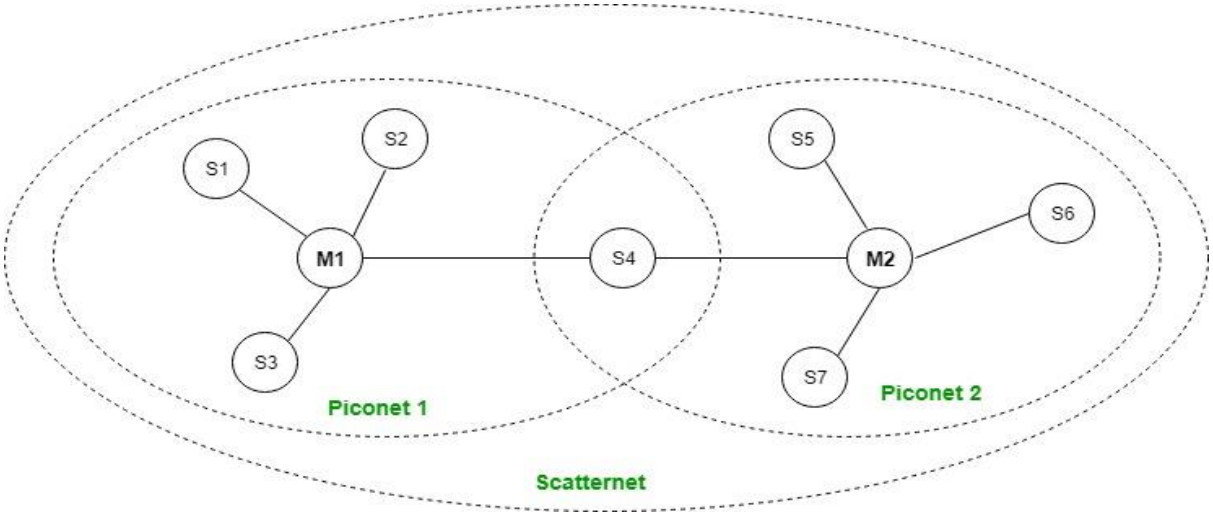
Q.1 Explain and draw the architecture of WiFi, Bluetooth and zigbee.

Ans:- **Wi-Fi Architecture**

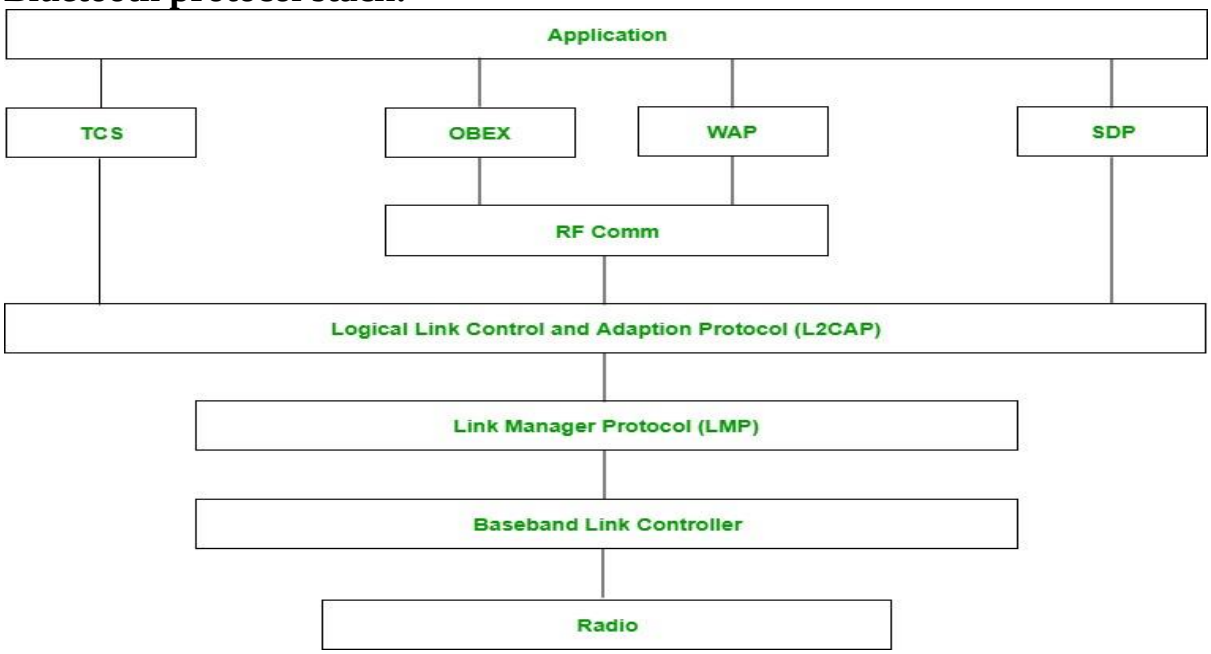
- 802.11 defines each computer, mobile, portable or fixed device as a station
- Difference between portable and mobile stations
- Basic Service Set (BSS)
- IBSSs and ad-hoc network
- Interconnected BSSs and Distribution System (DS)
- Access Points
- Creating large and complex networks by combining BSSs and DSs into an Extended Service Set, or ESS.
- A fundamental of 802.11: connection of the wireless network to existing wired networks. Use of a portal to do so.
- Services the DS must support:
- Station Services (SS)
- Authentication
- Deauthentication
- Privacy
- MAC Service Data Unit (MSDU) Delivery
- Distribution System Services (DSS)
- Association
- Reassociation
- Disassociation
- Distribution
- Integration



Bluetooth Architecture



Bluetooth protocol stack:



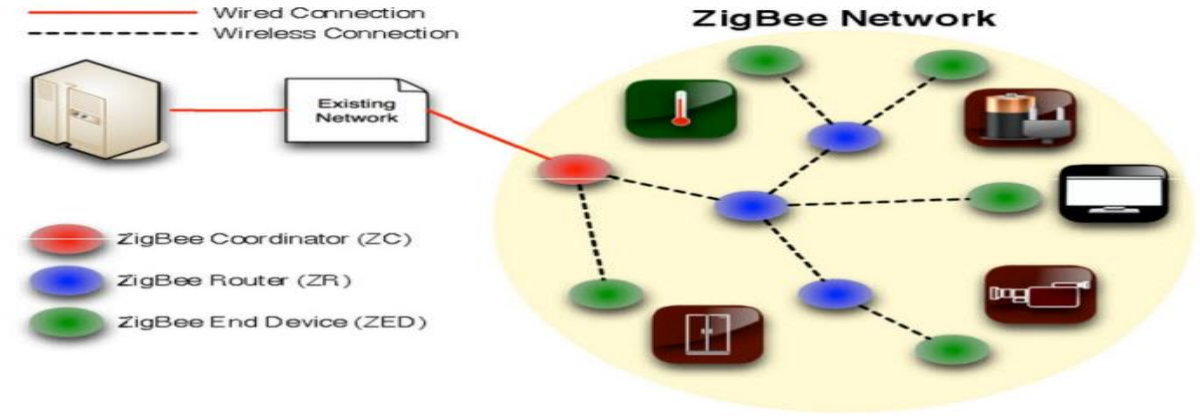
1. **Radio (RF) layer:**
   It performs modulation/demodulation of the data into RF signals. It defines the physical characteristics of bluetooth transceiver. It defines two types of physical link: connection-less and connection-oriented.
2. **Baseband Link layer:**
   It performs the connection establishment within a piconet.
3. **Link Manager protocol layer:**
   It performs the management of the already established links. It also includes authentication and encryption processes.
4. **Logical Link Control and Adaption protocol layer:**
   It is also known as the heart of the bluetooth protocol stack. It allows the communication between upper and lower layers of the bluetooth protocol stack. It packages the data packets received from upper layers into the form expected by lower layers. It also performs the segmentation and multiplexing.

5. **SDP layer:**
   It is short for Service Discovery Protocol. It allows to discover the services available on another bluetooth enabled device.
6. **RF comm layer:**
   It is short for Radio Frontend Component. It provides serial interface with WAP and OBEX.
7. **OBEX:**It is short for Object Exchange. It is a communication protocol to exchange objects between 2 devices.
8. **WAP:**
   It is short for Wireless Access Protocol. It is used for internet access.
9. **TCS:**
   It is short for Telephony Control Protocol. It provides telephony service.
10. **Application layer:**
    It enables the user to interact with the application.

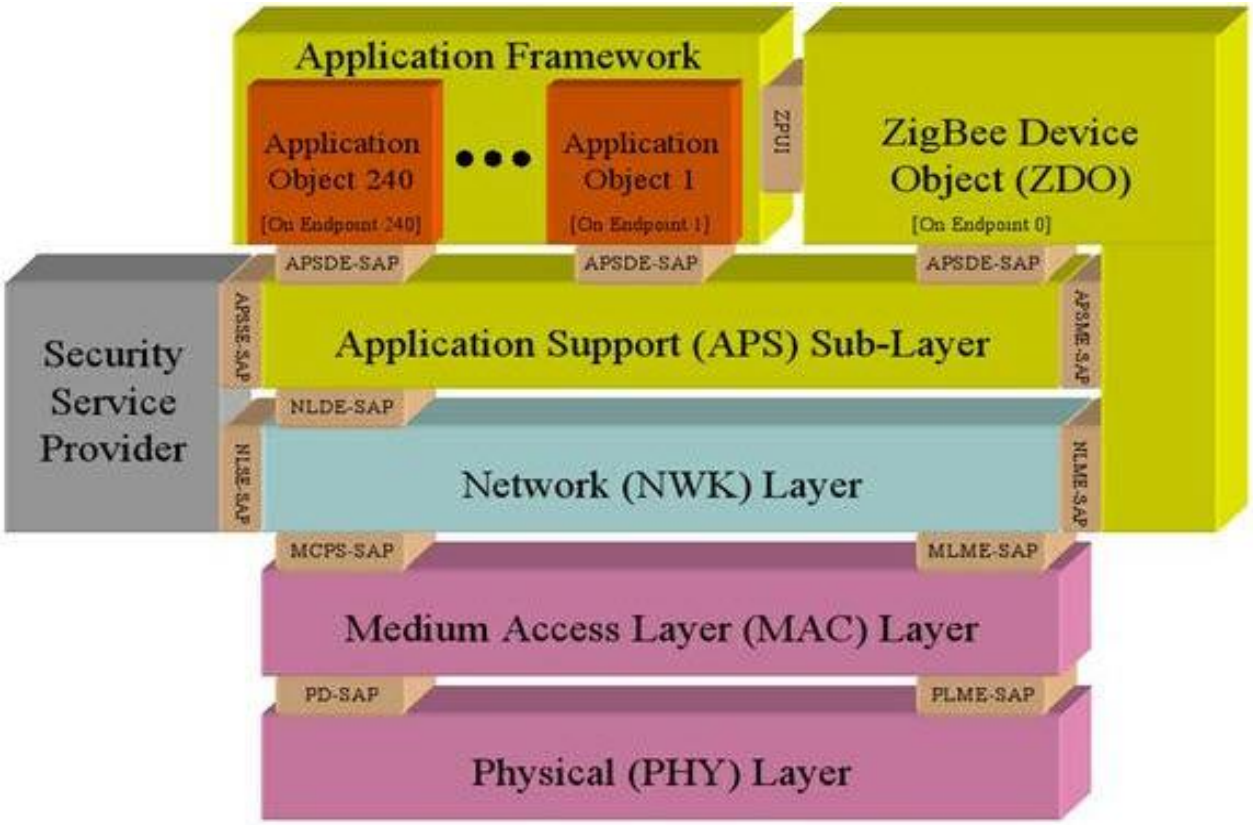
Zigbee Architecture

Zigbee system structure consists of three different types of devices such as Zigbee coordinator, Router and End device. Every Zigbee network must consist of at least one coordinator which acts as a root and bridge of the network. The coordinator is responsible for handling and storing the information while performing receiving and transmitting data operations. Zigbee routers act as intermediary devices that permit data to pass to and fro through them to other devices. End devices have limited functionality to communicate with the parent nodes such that the battery power is saved as shown in the figure. The number of routers, coordinators and end devices depends on the type of network such as star, tree and mesh networks.

Zigbee protocol architecture consists of a stack of various layers where IEEE 802.15.4 is defined by physical and MAC layers while this protocol is completed by accumulating Zigbee's own network and application layers.



Zigbee system architecture



Zigbee protocol architecture

Physical Layer: This layer does modulation and demodulation operations up on transmitting and receiving signals respectively.

MAC Layer: This layer is responsible for reliable transmission of data by accessing different networks with the carrier sense multiple access collision avoidance (CSMA). This also transmits the beacon frames for synchronizing communication.

Network Layer: This layer takes care of all network related operations such as network setup, end device connection and disconnection to network, routing, device configurations, etc.

Application Support Sub-Layer: This layer enables the services necessary for Zigbee device object and application objects to interface with the network layers for data managing services. This layer is responsible for matching two devices according to their services and needs.

Application Framework: It provides two types of data services as key value pair and generic message services. Generic message is a developer defined structure, whereas the key value pair is used for getting attributes within the application objects. ZDO provides an interface between application objects and APS layer in Zigbee devices. It is responsible for detecting, initiating and binding other devices to the network.

Q.2 Differentiate between WiFi, Bluetooth and zigbee.

Ans:-

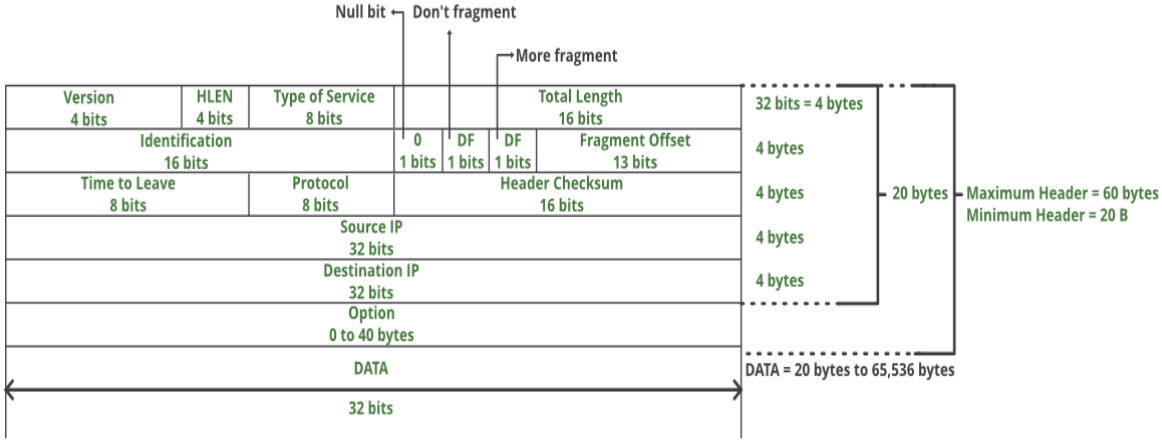
	ZigBee	Wi-Fi	Bluetooth
Range	10-100 meters	50-100 meters	10 – 100 meters
Networking Topology	Ad-hoc, peer to peer, star, or mesh	Point to hub	Ad-hoc, very small networks
Operating Frequency	868 MHz (Europe) 900-928 MHz (NA), 2.4 GHz (worldwide)	2.4 and 5 GHz	2.4 GHz
Complexity (Device and application impact)	Low	High	High
Power Consumption (Battery option and life)	Very low (low power is a design goal)	High	Medium
Security	128 AES plus application layer security		64 and 128 bit encryption
Typical Applications	Industrial control and monitoring, sensor networks, building automation, home control and automation, toys, games	Wireless LAN connectivity, broadband Internet access	Wireless connectivity between devices such as phones, PDA, laptops, headsets

	Bluetooth	ZigBee	Wi-Fi
IEEE Spec	IEEE 802.15.1	IEEE 802.15.4	IEEE 802.11b
Type of Module	HC-05	XBee Series 1	Arduino Yun*
Sleeping Mode	9 µA	12 µA	30 µA
Awake Mode	35 mA	50 mA	245 mA
Transmitting Mode	39 mA	52 mA	251 mA
Receiving Mode	37 mA	54 mA	248 mA
Power Supply	3.3 V	3.3 V	5 V

Q.3 Draw and explain the datagram header of IPv4 and IPv6. Differentiation between IPv4 and IPv6.

Ans: IPv4 Datagram Header

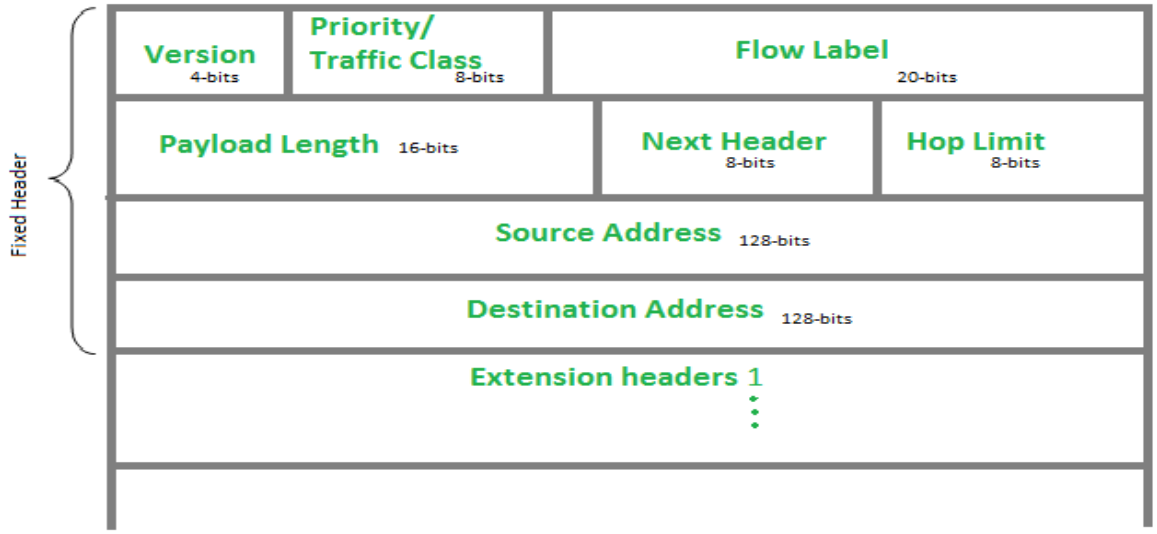
Size of the header is 20 to 60 bytes.



VERSION: Version of the IP protocol (4 bits), which is 4 for IPv4
HLEN: IP header length (4 bits), which is the number of 32 bit words in the header. The minimum value for this field is 5 and the maximum is 15.
Type of service: Low Delay, High Throughput, Reliability (8 bits)
Total Length: Length of header + Data (16 bits), which has a minimum value 20 bytes and the maximum is 65,535 bytes.
Identification: Unique Packet Id for identifying the group of fragments of a single IP datagram (16 bits)

Flags: 3 flags of 1 bit each: reserved bit (must be zero), do not fragment flag, more fragments flag (same order)
Fragment Offset: Represents the number of Data Bytes ahead of the particular fragment in the particular Datagram. Specified in terms of number of 8 bytes, which has the maximum value of 65,528 bytes.
Time to live: Datagram's lifetime (8 bits), It prevents the datagram to loop through the network by restricting the number of Hops taken by a Packet before delivering to the Destination.
Protocol: Name of the protocol to which the data is to be passed (8 bits)
Header Checksum: 16 bits header checksum for checking errors in the datagram header
Source IP address: 32 bits IP address of the sender
Destination IP address: 32 bits IP address of the receiver
Option: Optional information such as source route, record route. Used by the Network administrator to check whether a path is working or not.

IP version 6 Header Format:



Version (4-bits): Indicates version of Internet Protocol which contains bit sequence 0110.
Traffic Class (8-bits): The Traffic Class field indicates class or priority of IPv6 packet which is similar to *Service Field* in IPv4 packet. It helps routers to handle the traffic based on priority of the packet. If congestion occurs on router then packets with least priority will be discarded.
Flow Label (20-bits): Flow Label field is used by source to label the packets belonging to the same flow in order to request special handling by intermediate IPv6 routers, such as non-default quality of service or real time service. In order to distinguish the flow, intermediate router can use source address, destination address and flow label of the packets. Between a source and destination multiple flows may exist because many processes might be running at the same time. Routers or Host that do not support the functionality of flow label field and for default router handling, flow label field is set to 0. While setting up the flow label, source is also supposed to specify the lifetime of flow.
Payload Length (16-bits) : It is a 16-bit (unsigned integer) field, indicates total size of the payload which tells routers about amount of information a particular packet contains in its payload. Payload Length field includes extension headers (if any) and upper layer packet. In case length of payload is greater than 65,535 bytes (payload up to 65,535 bytes can be

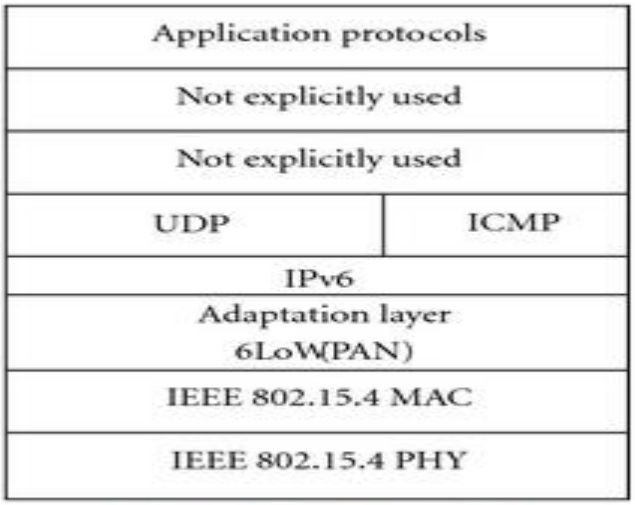
indicated with 16-bits), then the payload length field will be set to 0 and jumbo payload option is used in the Hop-by-Hop options extension header.
Next Header (8-bits): Next Header indicates type of extension header (if present) immediately following the IPv6 header. Whereas In some cases it indicates the protocols contained within upper-layer packet, such as TCP, UDP.
Hop Limit (8-bits): Hop Limit field is same as TTL in IPv4 packets. It indicates the maximum number of intermediate nodes IPv6 packet is allowed to travel. Its value gets decremented by one, by each node that forwards the packet and packet is discarded if value decrements to 0. This is used to discard the packets that are stuck in infinite loop because of some routing error.
Source Address (128-bits): Source Address is 128-bit IPv6 address of the original source of the packet.
Destination Address (128-bits): Destination Address field indicates the IPv6 address of the final destination (in most cases). All the intermediate nodes can use this information in order to correctly route the packet.
Extension Headers: In order to rectify the limitations of *IPv4 Option Field*, Extension Headers are introduced in IP version 6. The extension header mechanism is very important part of the IPv6 architecture. Next Header field of IPv6 fixed header points to the first Extension Header and this first extension header points to the second extension header and so on.

Basis for differences	IPv4	IPv6
Size of IP address	IPv4 is a 32-Bit IP Address.	IPv6 is 128 Bit IP Address.
Addressing method	IPv4 is a numeric address, and its binary bits are separated by a dot (.)	IPv6 is an alphanumeric address whose binary bits are separated by a colon (:). It also contains hexadecimal.
Number of header fields	12	8
Length of header filed	20	40
Checksum	Has checksum fields	Does not have checksum fields
Example	12.244.233.165	2001:0db8:0000:0000:0000:ff00:0042:7879
Type of Addresses	Unicast, broadcast, and multicast.	Unicast, multicast, and anycast.
Number of classes	IPv4 offers five different classes of IP Address. Class A to E.	IPv6 allows storing an unlimited number of IP Address.
Configuration	You have to configure a newly installed system before it can communicate with other systems.	In IPv6, the configuration is optional, depending upon on functions needed.

Q.4 Draw and explain 6LoWPAN protocol.

Ans:- 6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks), is a low power wireless mesh network where every node has its own IPv6 address. This allows the node to connect directly with the Internet using open standards.

6LoWPAN came to exist from the idea that the Internet Protocol could and should be applied even to the smallest devices, and that low-power devices with limited processing capabilities should be able to participate in the Internet of Things.



6LoWPAN protocol stack

6LoWPAN standards enable the efficient use of IPv6 over low power, low rate wireless networks on simple embedded nodes through an adaptation layer and optimisation of related protocols. The Maximum frame size of LOWPAN packet is 128 octets as specified by IEEE 802.15.4 while the frame size of IPv6 is 1280 octets. Thus an incompatibility exists in accommodating the IPv6 frame in a LOWPAN frame. In order to alleviate this issue, 6LoWPAN working group has suggested an additional adaptation layer between MAC layer and the network layer.

Q.5 Explain and differentiate TCP and UDP.

TRANSMISSION CONTROL PROTOCOL (TCP)	USER DATAGRAM PROTOCOL (UDP)
TCP is a connection-oriented protocol. Connection-orientation means that the communicating devices should establish a connection before	UDP is the Datagram oriented protocol. This is because there is no overhead for opening a connection, maintaining a connection, and

TRANSMISSION CONTROL PROTOCOL (TCP)	USER DATAGRAM PROTOCOL (UDP)
transmitting data and should close the connection after transmitting the data.	terminating a connection. UDP is efficient for broadcast and multicast type of network transmission.
TCP is reliable as it guarantees delivery of data to the destination router.	The delivery of data to the destination cannot be guaranteed in UDP.
TCP provides extensive error checking mechanisms. It is because it provides flow control and acknowledgment of data.	UDP has only the basic error checking mechanism using checksums.
Sequencing of data is a feature of Transmission Control Protocol (TCP). this means that packets arrive in-order at the receiver.	There is no sequencing of data in UDP. If ordering is required, it has to be managed by the application layer.
TCP is comparatively slower than UDP.	UDP is faster, simpler and more efficient than TCP.
Retransmission of lost packets is possible in TCP, but not in UDP.	There is no retransmission of lost packets in User Datagram Protocol (UDP).
TCP has a (20-80) bytes variable length header.	UDP has a 8 bytes fixed length header.

TRANSMISSION CONTROL PROTOCOL (TCP)	USER DATAGRAM PROTOCOL (UDP)
TCP is heavy-weight.	UDP is lightweight.
TCP doesn't supports Broadcasting.	UDP supports Broadcasting.
UDP is used by DNS, DHCP, TFTP, SNMP, RIP, and VoIP.	

Q.6 Explain MQTT protocol and its architecture.

Ans:- In IoT, message transmission between different devices is important because an IoT appliance has to deliver an instruction to a further appliance to manage system. Compared to polling protocol, Push protocol is the suitable message communication protocol for IoT appliances as it is constructed in poor bandwidth network. MQTT, XMPP and CoAP protocol were implemented through these push message services. These protocols are applicable according to different situations. In particular, MQTT has been utilized as part of many IoT gadgets and instant message delivery systems because it was intended to work on low power machines as a light-weight protocol.

The typical MQTT architecture can be divided into two main components

1. Client

Client could be a Publisher or Subscriber and it always establishes the network connection to the Server (Broker). It can do the following things:

1. Publish messages for the interested users.
2. Subscribe in interested subject for receiving messages.
3. Unsubscribe to extract from the subscribed subjects.
4. Detach from the Broker.

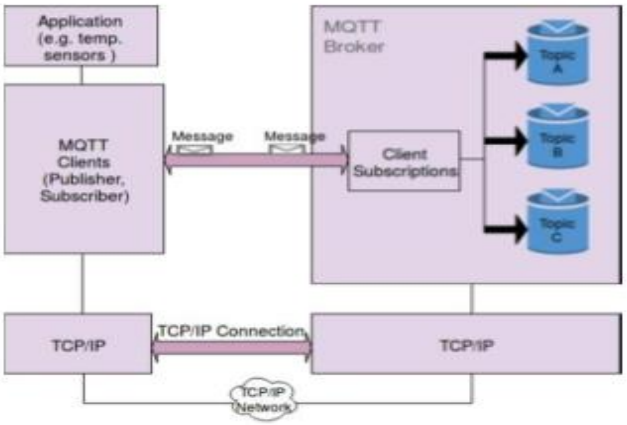


Figure 2. MQTT Architecture

2. Broker

Broker controls the distribution of information and mainly responsible for receiving all messages from publisher, filtering them, decide who is interested in it and then sending the messages to all subscribed clients. It can do the following things:

1. Accept Client requests.

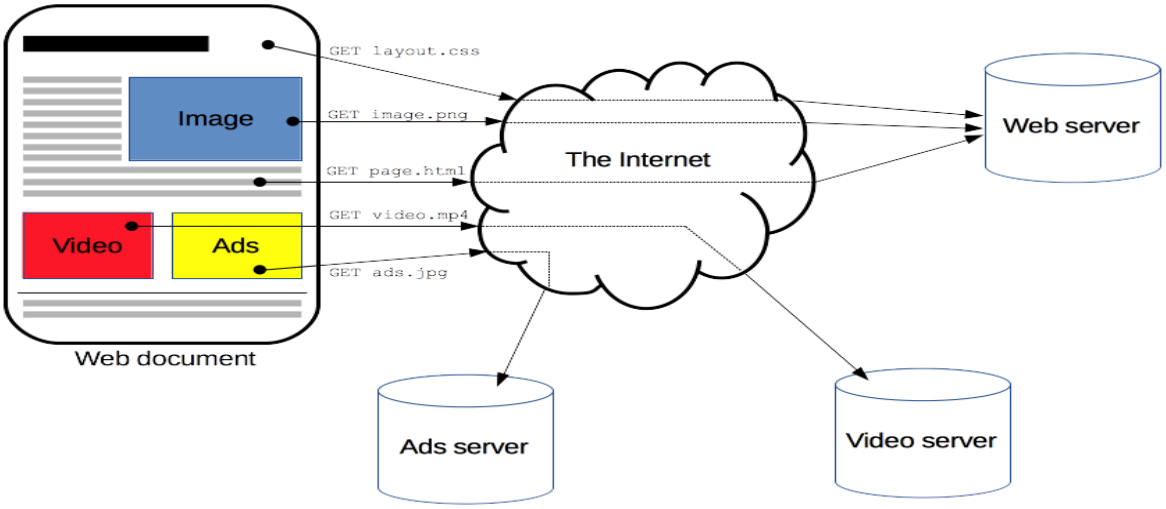
2. Receives Published messages by Users.
3. Processes different requests like Subscribe and Unsubscribe from Users.
4. After receiving messages from publisher sends it to the interested Users.

Figure 3. Working of MQTT

Q.7 Write a short note on: HTTP, CoAP, XMPP, AMQP.

Ans:- **1. HTTP**

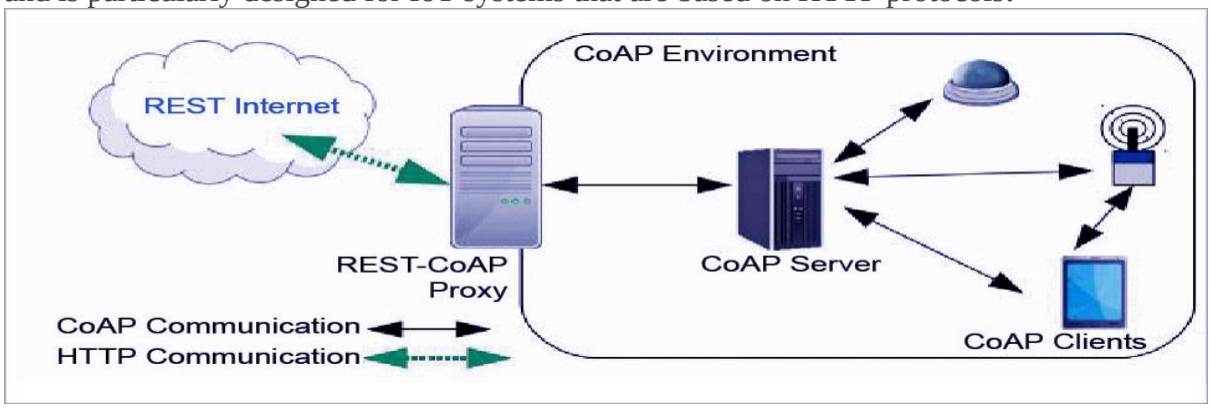
HTTP is a protocol which allows the fetching of resources, such as HTML documents. It is the foundation of any data exchange on the Web and it is a client-server protocol, which means requests are initiated by the recipient, usually the Web browser. A complete document is reconstructed from the different sub-documents fetched, for instance text, layout description, images, videos, scripts, and more.



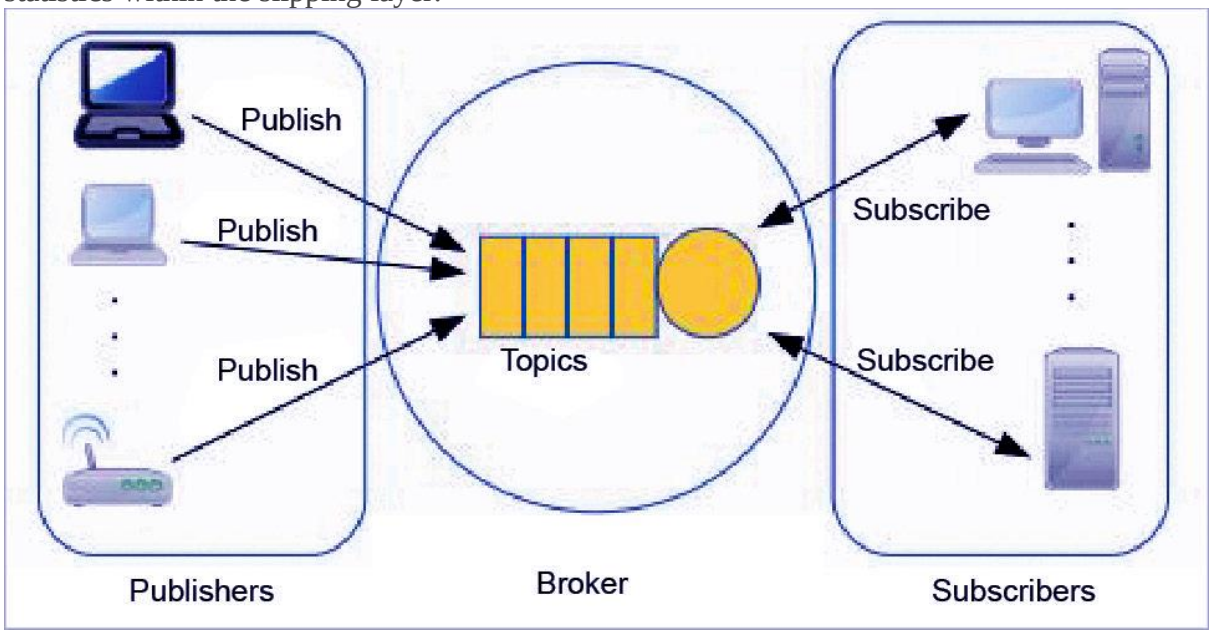
Clients and servers communicate by exchanging individual messages (as opposed to a stream of data). The messages sent by the client, usually a Web browser, are called *requests* and the messages sent by the server as an answer are called *responses*.

2.Constrained Application Protocol (CoAP)

CoAP is an internet utility protocol for constrained gadgets. It is designed to enable simple, constrained devices to join IoT through constrained networks having low bandwidth availability. This protocol is primarily used for machine-to-machine (M2M) communication and is particularly designed for IoT systems that are based on HTTP protocols.



CoAP makes use of the UDP protocol for lightweight implementation. It also uses restful architecture, which is just like the HTTP protocol. It makes use of dtls for the cozy switch of statistics within the slipping layer.



3. XMPP

XMPP is the Extensible Messaging and Presence Protocol, a set of open technologies for instant messaging, presence, multi-party chat, voice and video calls, collaboration, lightweight middleware, content syndication, and generalized routing of XML data.

XMPP was originally developed in the Jabber open-source community to provide an open, decentralized alternative to the closed instant messaging services at that time. XMPP offers several key advantages over such services:

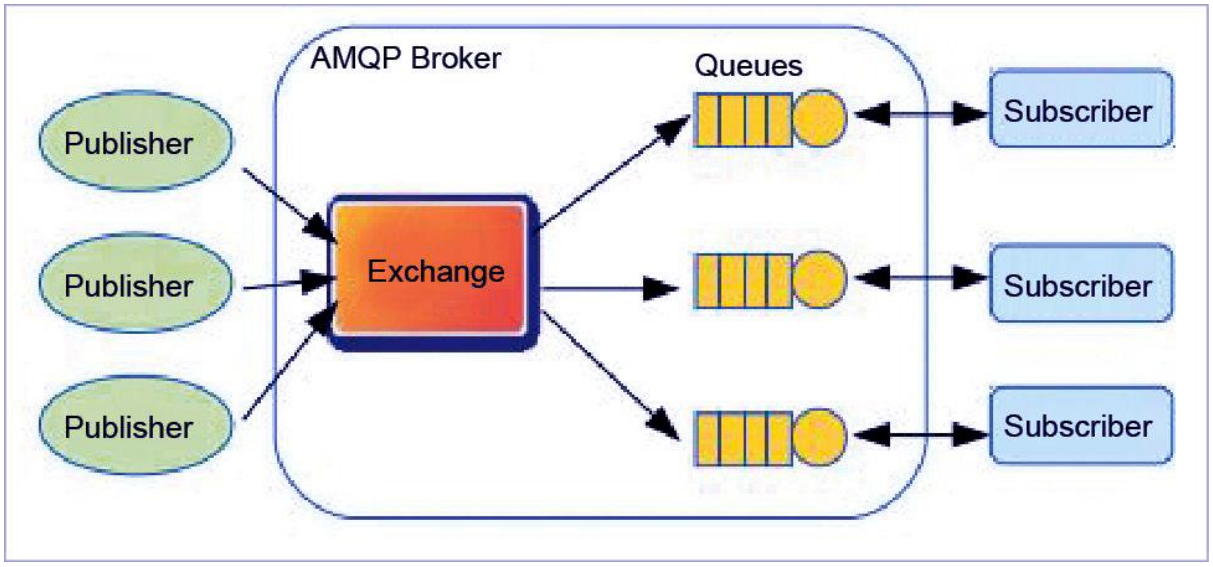
- **Open** — the XMPP protocols are free, open, public, and easily understandable; in addition, multiple implementations exist in the form clients, servers, server components, and code libraries.

- **Standard** — the Internet Engineering Task Force (IETF) has formalized the core XML streaming protocols as an approved instant messaging and presence technology. The XMPP specifications were published as RFC 3920 and RFC 3921 in 2004, and the XMPP Standards Foundation continues to publish many XMPP Extension Protocols. In 2011 the core RFCs were revised, resulting in the most up-to-date specifications (RFC 6120, RFC 6121, and RFC 7622).
- **Proven** — the first Jabber/XMPP technologies were developed by Jeremie Miller in 1998 and are now quite stable; hundreds of developers are working on these technologies, there are tens of thousands of XMPP servers running on the Internet today, and millions of people use XMPP for instant messaging through public services such as Google Talk and XMPP deployments at organizations worldwide.
- **Decentralized** — the architecture of the XMPP network is similar to email; as a result, anyone can run their own XMPP server, enabling individuals and organizations to take control of their communications experience.
- **Secure** — any XMPP server may be isolated from the public network (e.g., on a company intranet) and robust security using SASL and TLS has been built into the core XMPP specifications. In addition, the XMPP developer community is actively working on end-to-end encryption to raise the security bar even further.
- **Extensible** — using the power of XML, anyone can build custom functionality on top of the core protocols; to maintain interoperability, common extensions are published in the XEP series, but such publication is not required and organizations can maintain their own private extensions if so desired.
- **Flexible** — XMPP applications beyond IM include network management, content syndication, collaboration tools, file sharing, gaming, remote systems monitoring, web services, lightweight middleware, cloud computing, and much more.
- **Diverse** — a wide range of companies and open-source projects use XMPP to build and deploy real-time applications and services; you will never get "locked in" when you use XMPP technologies.

4.Advanced Message Queuing Protocol (AMQP)

AMQP is a software layer protocol for message-oriented middleware environment. It supports reliable verbal exchange through message transport warranty primitives like at-most-once, at least once and exactly as soon as shipping.
The AMQP – IoT protocols consist of hard and fast components that route and save messages within a broker carrier, with a set of policies for wiring the components together. The AMQP protocol enables patron programs to talk to the dealer and engage with the AMQP model.
This version has the following three additives, which might link into processing chains in the server to create the favoured capabilities.

- **Exchange:** Receives messages from publisher primarily based programs and routes them to 'message queues'.
- **Message Queue:** Stores messages until they may thoroughly process via the eating client software.
- **Binding:** States the connection between the message queue and the change.



Q.8 Write down the differentiation between HTTP, CoAP, XMPP, AMQP and MQTT protocol.

Criteria	HTTP	CoAP	MQTT
Architecture	Client/Server	Client/Server or Client/Broker	Client/Broker
Abstraction	Request/Response	Request/Response or Publish/ Subscribe	Publish/Subscribe
Header Size	Undefined	4 Byte	2 Byte
Message size	Large and Undefined (depends on the web server or the programming technology)	Small and Undefined (normally small to fit in single IP datagram)	Small and Undefined (up to 256 MB maximum size)
Semantics/Methods	Get, Post, Head, Put, Patch, Options, Connect, Delete	Get, Post, Put, Delete	Connect, Disconnect, Publish, Subscribe, Unsubscribe, Close
Quality of Service (QoS) /Reliability	Limited (via Transport Protocol - TCP)	Confirmable Message or Non-confirmable Message	QoS 0 - At most once QoS 1 - At least once QoS 2 - Exactly once
Transport Protocol	TCP	UDP, TCP	TCP (MQTT-SN can use UDP)
Security	TLS/SSL	DTLS/IPSEC	TLS/SSL
Default Port	80/443 (TLS/SSL)	5683 (UDP)/5684 (DTLS)	1883/8883 (TLS/SSL)

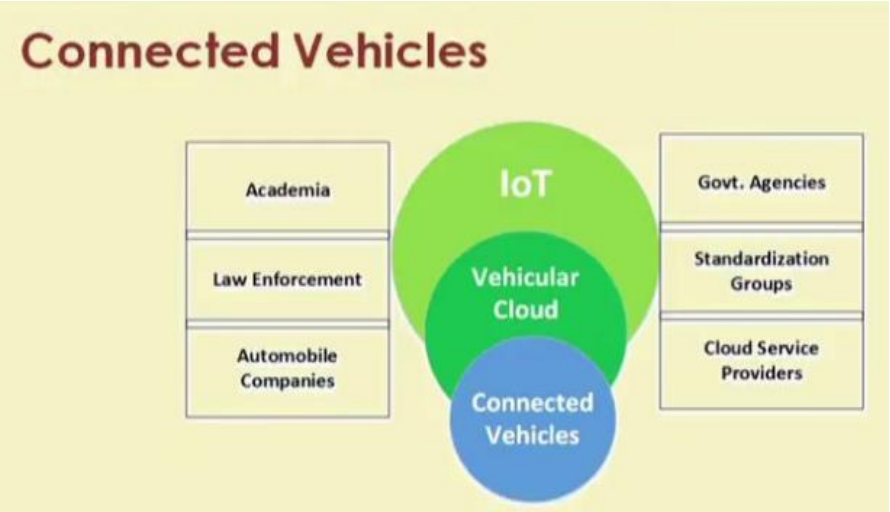
	AMQP	MQTT	XMPP
goal	replacement of proprietary protocols	messaging for resource-constrained devices	instant messaging, adopted for wider use
format	binary	binary	XML-based
API	divided into classes (> 40 methods in RabbitMQ)	simple (5 basic operations with 2-3 packet types for each)	different XML items with multiple types
reliability	publisher/subscriber acknowledgements, transactions	acknowledgements	Acknowledgments and resumptions (XEP-198)
security	SASL, TLS/SSL	no built-in TLS/SSL, header authentication	SASL, TLS/SSL
extensibility	extension points	none	extensible

	AMQP	CoAP	MQTT	REST / HTTP	XMPP
TRANSPORT	TCP/IP	UDP/IP	TCP/IP	TCP/IP	TCP/IP
INTERACTION MODEL	Point-to-Point Message Exchange	Request-Reply (REST)	Publish-and-Subscribe	Request-Reply	Point-to-Point Message Exchange
SCOPE	Device-to-Device Device-to-Cloud Cloud-to-Cloud	Device-to-Device	Device-to-Cloud Cloud-to-Cloud	Device-to-Cloud Cloud-to-Cloud	Device-to-Cloud Cloud-to-Cloud
AUTOMATIC DISCOVERY	-	✓	-	-	-
CONTENT AWARENESS	-	-	-	-	-
QoS	Limited	Limited	Limited	-	-
INTEROPERABILITY LEVEL	Structural	Semantic	Foundational	Semantic	Structural
SECURITY	TLS + SASL	DTLS	TLS	HTTPS	TLS + SASL
DATA PRIORITIZATION	-	-	-	-	-
FAULT TOLERANCE	Implementation-Specific	Decentralized	Broker is SPoF	Server is SPoF	Server is SPoF

Q.9 Write a detailed case study on any 4 area: Smart cities and Smart Homes; Connected Vehicles; Industrial IoT; Agriculture; Activity Monitoring.

Ans:- **1.Connected Vehicles**

Connected Vehicles are equipped with sensors, networking and communicating devices which are capable of communicating with other devices within the vehicle, with the other similar vehicles and with fixed infrastructure.



Vehicles-to-Everything(V2X) Paradigm

- Main component of future intelligent Transport System (ITS).
- Enables Vehicles to wirelessly share a diverse range of information.
- Information sharing may be with other vehicles, pedestrians, or fixed infrastructure (mobile towers, parking meters, etc.)
- Allows for traffic management, ensuring on road and off-road safety, mobility for travelling.
- Follows a distributed architecture, where contents are widely distributed over the network.
- Not restricted to single source information provider.
- Designed mainly for high mobile environments.
- Can share information to nodes in vicinity, as well as remotely located.
- Has greatly enhanced travel efficiency .as well as safety.
- The network is mainly used as a tool for sharing and disseminating information.

Features of TCP/IP in V2X

- Designed mainly for handling information exchanging between a single pair of entities.
- Information exchange dependent on the location of data.
- Can only identify the address of endpoints, which alone is not useful for content distribution.
- Increase in number of wireless devices, restricts the mobility of nodes.

Content Centric Networking (CCN)

- CCN is derived from information Centric Networking (ICN)Architecture.
- Focuses more on the data than its actual location.

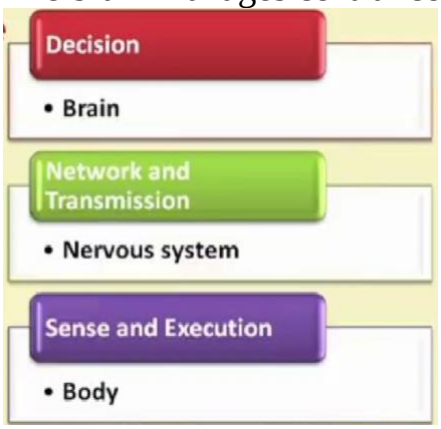
- Hierarchically named data.
- Hierarchical data is transmitted directly instead of being part of a conversation.
- Enables scalable and efficient data dissemination.
- In network caching allows for low data traffic.
- Works well in highly mobile environments.

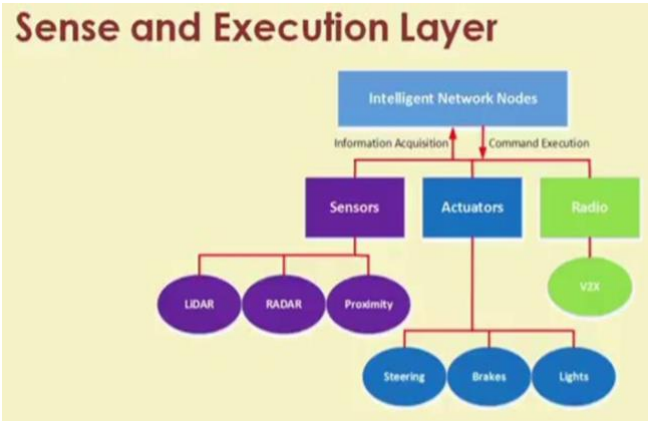
Vehicular Ad-hoc Networks (VANETs)

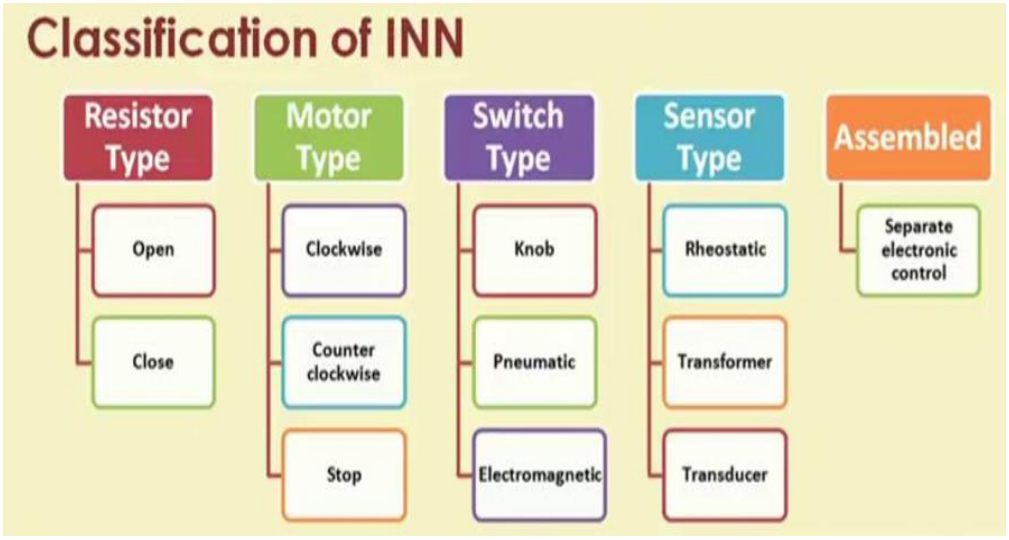
- Based on:
    1. Dedicated Short-Range Communication (DSRC)
    2. Wireless Access in Vehicular Environment (WAVE)
- Routing protocols derived from MANETs.
- High throughput achievable in mobile environments.
- Guaranteed low-latency in Mobile environments.

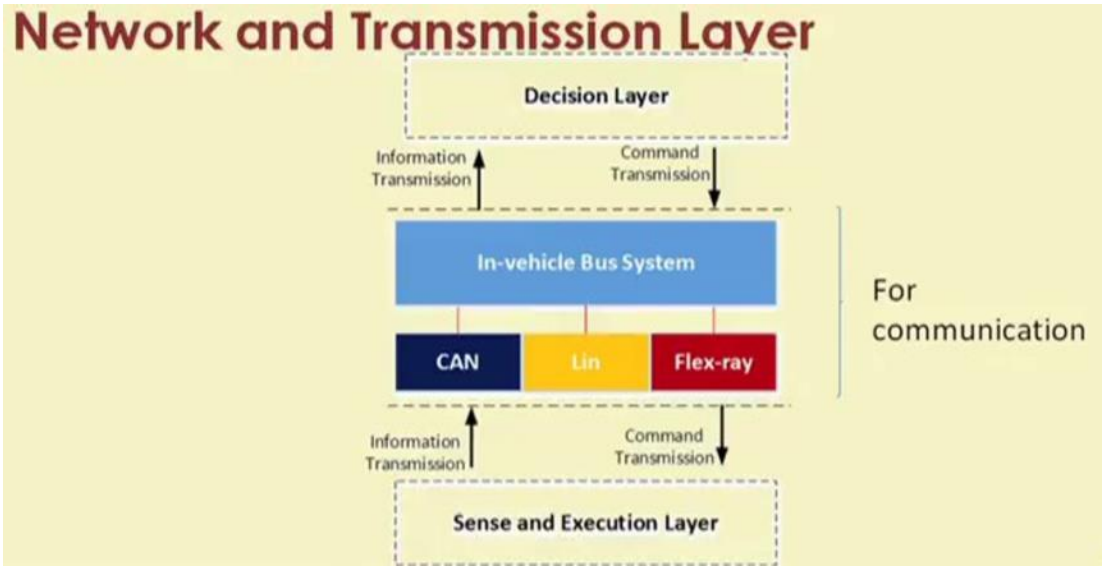
Body and Brain Architecture

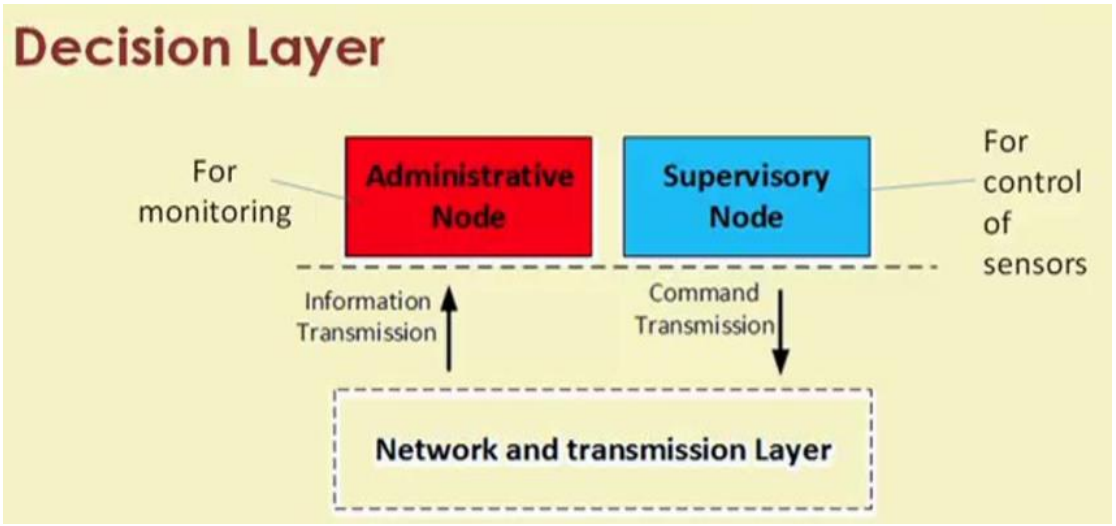
- An in-vehicle networking architecture.
- Three layered Architecture.
- The body consists of intelligent networking nodes (INN) which constantly collect information from the vehicle.
- The brain manages central coordination.





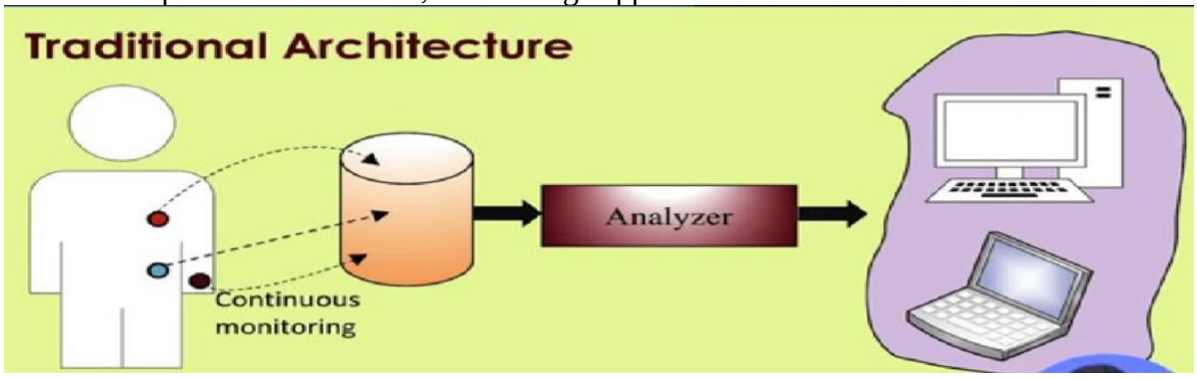






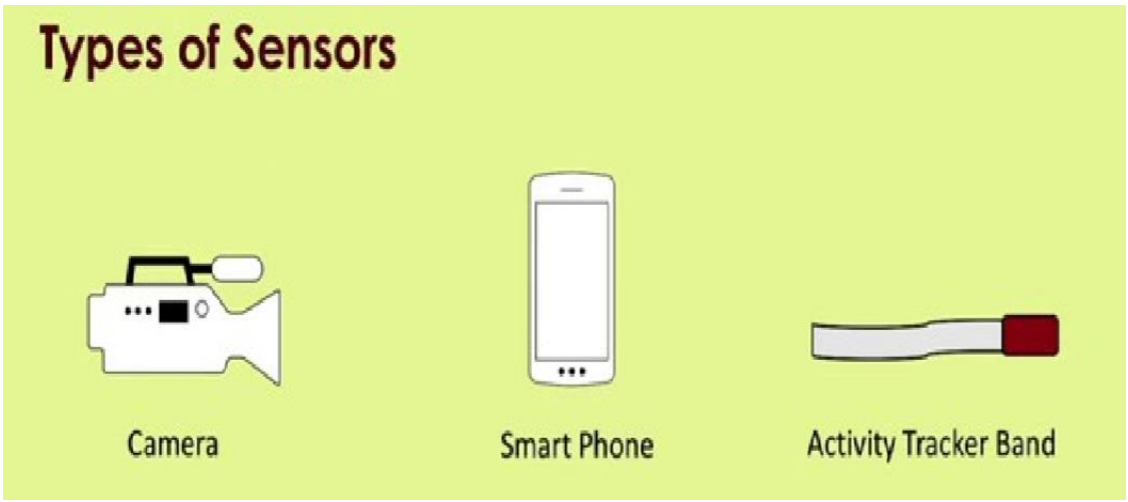
2.Activity Monitoring

Wearable sensors have become very popular for different purposes such as: medical, child care, elderly-care, etc. These sensors help in monitoring the physical activities of humans. Particularly in IOT scenarios, activity monitoring plays an important role for providing better quality of life and safe guarding humans. Provides information accurately in a reliable manner and provides continuous, monitoring support.



Advantages

- Continuous monitoring of activities in daily observation of human behavior and repetitive pattern in their activities.
- Easy integration and fast equipping.
- Long term monitoring
- Utilization of sensors of handheld devices
    1. Accelerometer
    2. Gyroscope
    3. GPS
    4. others



Data Analysis Tools

- Statistical: sensor data
- Machine Learning Based: Sensor data
- Deep Learning Based
    - Sensor data
    - Images
    - Videos

Approaches

- In place

  - On the device
  - Power intensive
  - No network connection required
- Network Based
  - Larger and processing intensive methods can be applied
  - Group based analytics possible
  - Low power consumption
  - Average to good network connection

3. Smart cities and Smart Homes:-

In this case study we are going to talk about how IOT can help in building the smart cities and smart homes as you know that throughout the world and even in countries like India, there is a lot of focus on building smart cities. Of course, the scope of smart cities in each of these different countries is different and the scope again depends on the priority areas of each of these countries and their government. Now for instance in India, since the last few years, there have been a couple of cities that have been identified and phase wise these cities have been given funds to build or to transform them as smart cities.

So, when we talked about smart cities; what is it. So, in addition to the regular infrastructure that is there in any city for example, the urban infrastructure consisting of office buildings residential areas hospitals school's transportation police and so on you also need something in addition to make the cities smart. So, what is this in addition let us talk about. So, smart means what smart means that it is in terms of the services that are given to the respective stake holders of these cities. So, citizens are able to do things in a better manner in an improved manner then usual and how is that made possible that is made possible with the help of nothing, but the ICT technologies information and communication technologies which also includes electronics embedded electronics different other advanced topologies in electrical in an electrical sciences and so on. So, computers electronics put together can make these cities smart.

Example:- So, first of all let us consider any smart city. So, if we are talking about a smart city we need to have the basic components for example, transport there has to be a railways there has to be hospitals there has to be schools there has to be let us say traffic control traffic control waste management waste management banking then.

So, like this these are some of the different things in a smart city right and one thing I have missed which is very much essential is the police. So, as you can see that we have to transform all of these different components of any city to be smart. So, for which the technology is that we have studied. So, far in the previous lectures will have to be taken help of. So, definitely will have to take help of sensors sensor networks sensor networks then actuators then the different other communication technologies RFID, NFC, ZWAVE and so and so forth. So, many different things that we have covered in all these previous lectures of this course on IoT, so, all these will have to be used in order to make this transformation. So, these are the different ICT information and communication technologies that will have to be used right.

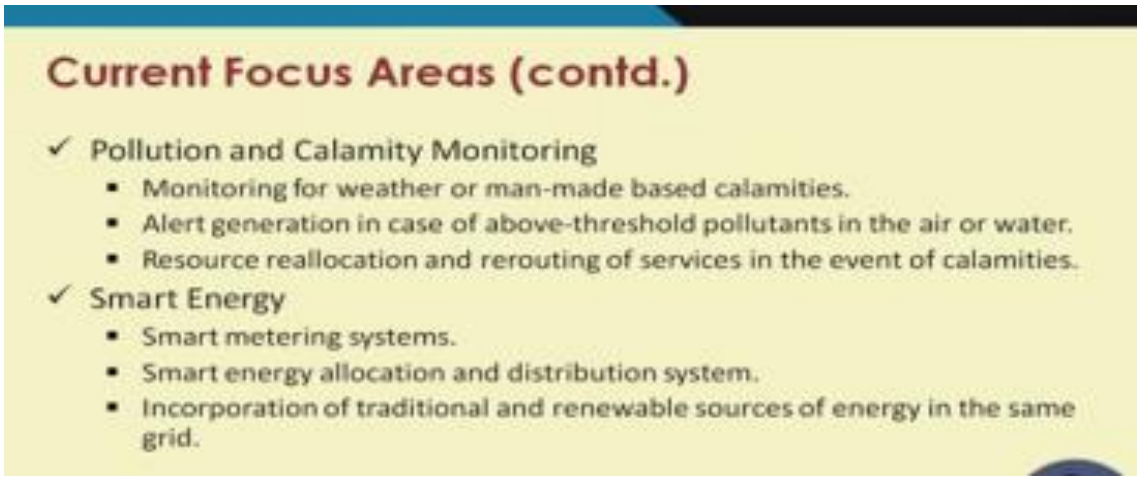
Analogy:-

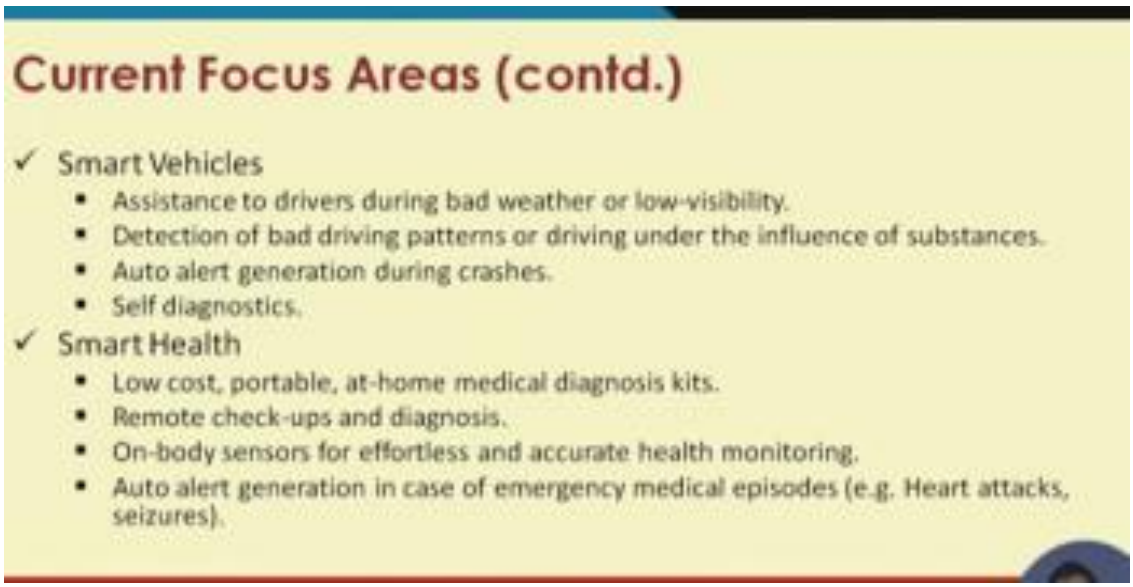
Humans	Smart Cities
Skeleton	Buildings, Industries, People
Skin	Transportation, Logistics
Organs	Hospital, Police, Banks, Schools
Brain	Ubiquitously embedded intelligence
Nerves	Digital telecommunication networks
Sensory Organs	Sensors, Tags
Cognition	Software

So, all these basically necessitate the building of smart cities using advanced ICT tools. So, let us draw some analogy when we talk about a human when we talk about a human humans have the skeleton the skin the organs different types of organs brains nerves sensory organs cognition and so on in the smart city as well in the same way has as a human has a skeleton skin and organs smart cities or rather any city rather any city has buildings industries people transportation logistics hospital police banks schools. So, these are there, but on top of that if there is a human with skeleton skin and organs, but no brains no nerves no sensory organs no cognition. So, you do not have you know life in that human you do not have any life in that human.

- **Focused Areas:-**
  1. **Application Focus Areas:-** These are some of the application focus areas we have smart economy. So, because of the ever increasing competitiveness you need to improve you need to improve your infrastructure the economy to make it smart. So, I will talk about that in more detail shortly now you need to also improve the citizen participation in any good governance in any good governance you need to improve you need to increase the citizen participation and how is that possible you need to take help of the ICT tools.

2. Current Focus Areas:- Now, we have the different focus areas we have smart homes smart parking lots in a smart home situation we need to have I will talk about smart homes in more detail later on, but in a smart home situation we have the health monitoring done in a smart way at home this you know the medical data made available to the doctors whenever there is a health criticality the corresponding house physician would be informed the physician can take requisite action based on the severity of severity or criticality of the of the health of the patient.





- **IOT Challenges in Smart Cities:-** There are different IOT challenges in smart cities security and privacies one. So, because you know all these different infrastructure are made available to all different types of citizens. So, you know you expose yourself to different types of attacks the government officers there are different files etcetera you know you make yourself vulnerable to different types of attacks privacy leaks and so on when you open up more and more

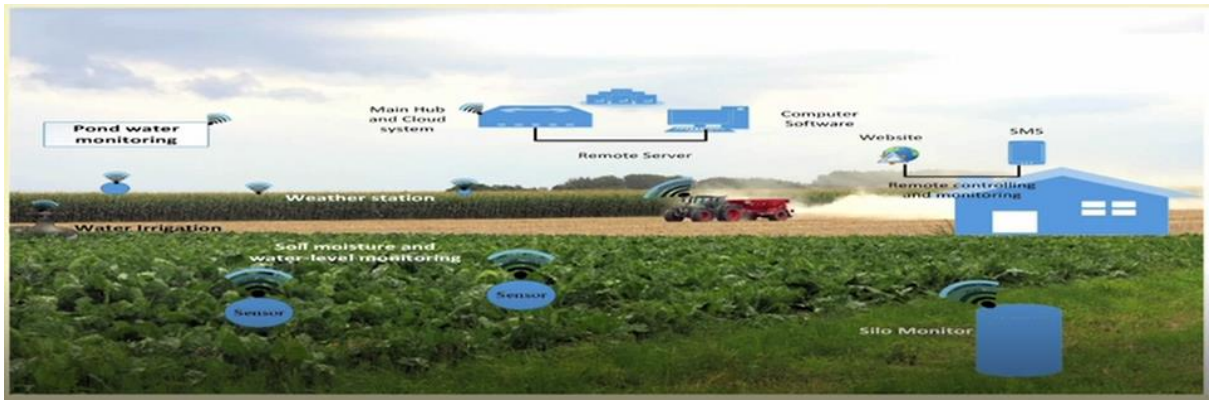
  There are legal and social issues as well for example, services that are based on users user provided information may be subject to local or other national and international laws and that also has to be taken care of in a very smart way individual and informed consent is required for

using humans as data sources big data issues are there you know huge volumes of data coming at high speeds and you know different types of vary various types of data media you know text data and so on.

4.Agriculture

Another important domain for Iot is the agriculture domain where the IoT system plays a vital role for soil and crop monitoring and provides a proper solution accordingly.
Using smart farming through IoT technologies helps farmers to reduce waste generation and increase productivity.

Agricultural use of IoT more specifically on use of IoT for smart irrigation. A smart irrigation management system, the system's name is AgriSens.



What is going to happen to the use of IoT in agriculture and what is going to happen in the future? So, the picture that we see in front of us is an agricultural field, a hypothetical one where there are different types of sensors that are planted sensors such as for soil, moisture and water level monitoring for automated irrigation performance performing automated irrigation. Automated recycling of organic waste, vermicomposting automated sowing and weeding and so on and so forth, so many different things automated systems fitted with sensors fitted with different actuators are going to be used for making agriculture smarter.

So, the objectives of this smart water management system, the AgriSens system are how less water can be used for getting more yield in terms of crop productivity; that means, and typically you know. So, what happens is for plants such as rice; that means, paddy plants wheat and so on. These basically are dependent on the soil moisture, the water level in the soil and so on and so forth and many other climatic factors.

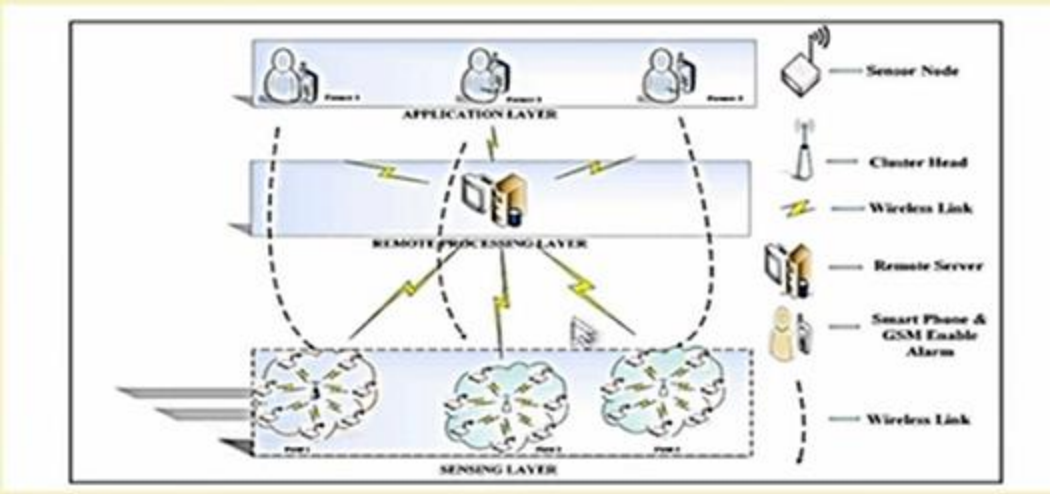
Smart Water Management using loT
Objectives

- More yields with less water
- Save limited water resource in a country
- Automatic Irrigation
- Dynamic irrigation treatments in the different phases of a crop's life cycle
- Remote monitoring and controlling

Proposed architecture

- Sensing and actuating layer
- Processing, storage, and service layer

- Application layer



The proposed architecture of the AgriSens system for offering smart water management. So, we have different layers of the system so we have the sensing layer the remote processing layer and the application layer the sensing layer basically has different types of sensors soil moisture water level etcetera which through data from different clusters this through data through their cluster heads to the remote processing server and different analytics and run and those data are made available to the different applications in the application layer.

Design

- Integrated design for sensors
- Integrated design for sensor node
- Integrated design for remote server

The EC 05 soil moisture sensor has been put there and has been dug inside the surface of the earth. So, soil moisture sensor is basically put inside and is installed inside the level the level of the mud level of mud or level of earth. So, it is inside it is dug inside.

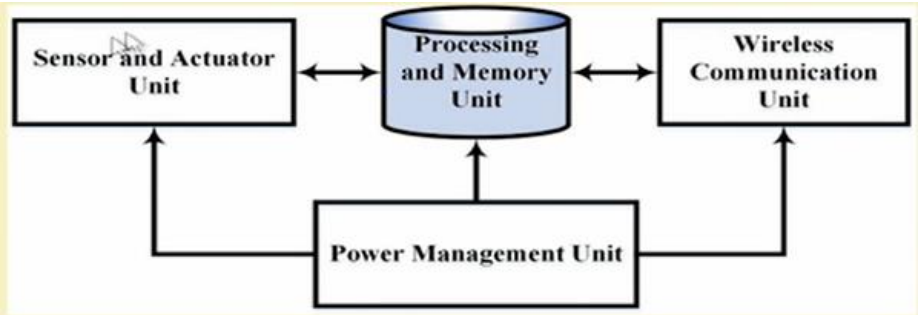


Fig 2: The block diagram of a sensor node

So, this is the overall design of the sensor node. So, here basically what we have is apart from the sensors and actuators we have a processing unit and the memory unit we have wireless communication unit and we have the power management unit.

Integrated design for remote server

- Repository data server: Communicates with the deployed IoT gateway in the field by using GPRS technology.
- Web server: To access field data remotely.
- Multi users server: Sends field information to farmer's cell using SMS technology and also executes farmer's query and controlling messages.

There are different sensors, one is a soil moisture sensor which is basically buried in ground and there is another sensor which is the water level sensor. So, the soil moisture sensor basically as this name says that it basically sensors the soil moisture and the water level sensor is how much is the stagnant water level in this particular grid. So, this is what it measures. And these two sensor data are sent to this particular node.

Q.10 Write a security issues in IoT.

Ans:- 1. Insufficient testing and updating

2. Brute-forcing and the issue of default passwords

3. IoT malware and ransomware

4. IoT botnets aiming at cryptocurrency

5. Data security and privacy concerns (mobile, web, cloud)

6. Small IoT attacks that evade detection

7. AI and automation

8. Home Invasions

9. Remote vehicle access

10. Untrustworthy communication

Q11. Differentiate between HTTP, AMQP, and MQTT.

	HTTP	AMQP	MQTT
Get	Yes	Yes	No
Caching Read	Yes	No	Yes
Put	Yes	No	No

Post	Yes	Yes	No
Delete	Yes	No	No
Content filtering	No	Yes	No
Typed headers	No	Yes	No
Resumeable transfer	Yes	Yes	Yes
Transactions	No	Yes	No
SSL/TLS	Yes	Yes	Yes
Kerberos	Yes	Yes	No
SASL	No	Yes	Yes
Symmetric Protocol	No	Yes	No
Socket Multiplexing	No	Yes	Yes
Out-of-order messaging	No	Yes	Yes
Server initiated transfers	No	Yes	No
Single packet send	Yes	Yes	Yes
Store-and-forward	No	Yes	Yes
Publish-and-subscribe	No	Yes	Yes
Defined error recovery	No	Yes	No
Well defined addresses	Yes	Yes	Yes
Content-based routing	No	Yes	No
Credit-based flow control	No	Yes	No

Criteria	MQTT	CoAP	AMQP	HTTP
1. Year	1999	2010	2003	1997
2. Architecture	Client/Broker	Client/Server or Client/Broker	Client/Broker or Client/Server	Client/Server
3. Abstraction	Publish/Subscribe	Request/Response or Publish/Subscribe	Publish/Subscribe or Request/Response	Request/Response
4. Header Size	2 Byte	4 Byte	8 Byte	Undefined
5. Message Size	Small and Undefined (up to 256 MB maximum size)	Small and Undefined (normally small to fit in single IP datagram)	Negotiable and Undefined	Large and Undefined (depends on the web server or the programming technology)
6. Semantics/ Methods	Connect, Disconnect, Publish, Subscribe, Unsubscribe, Close	Get, Post, Put, Delete	Consume, Deliver, Publish, Get, Select, Ack, Delete, Nack, Recover, Reject, Open, Close	Get, Post, Head, Put, Patch, Options, Connect, Delete
7. Cache and Proxy Support	Partial	Yes	Yes	Yes
8. Quality of Service (QoS)/ Reliability	QoS 0 - At most once (Fire-and-Forget), QoS 1 - At least once, QoS 2 - Exactly once	Confirmable Message (similar to At most once) or Non-confirmable Message (similar to At least once)	Settle Format (similar to At most once) or Unsettle Format (similar to At least once)	Limited (via Transport Protocol - TCP)
9. Standards	OASIS, Eclipse Foundations	IETF, Eclipse Foundation	OASIS, ISO/IEC	IETF and W3C
10. Transport Protocol	TCP (MQTT-SN can use UDP)	UDP, SCTP	TCP, SCTP	TCP
11. Security	TLS/SSL	DTLS, IPSec	TLS/SSL, IPSec, SASL	TLS/SSL
12. Default Port	1883/ 8883 (TLS/SSL)	5683 (UDP Port)/ 5684 (DLTS)	5671 (TLS/SSL), 5672	80/ 443 (TLS/SSL)
13. Encoding Format	Binary	Binary	Binary	Text
14. Licensing Model	Open Source	Open Source	Open Source	Free
15. Organisational Support	IBM, Facebook, Eurotech, Cisco, Red Hat, Software AG, Tibco, ITSO, M2Mi, Amazon Web Services (AWS), InduSoft, Fiorano	Large Web Community Support, Cisco, Contiki, Erika, IoTivity	Microsoft , JP Morgan, Bank of America, Barclays, Goldman Sachs, Credit Suisse	Global Web Protocol Standard