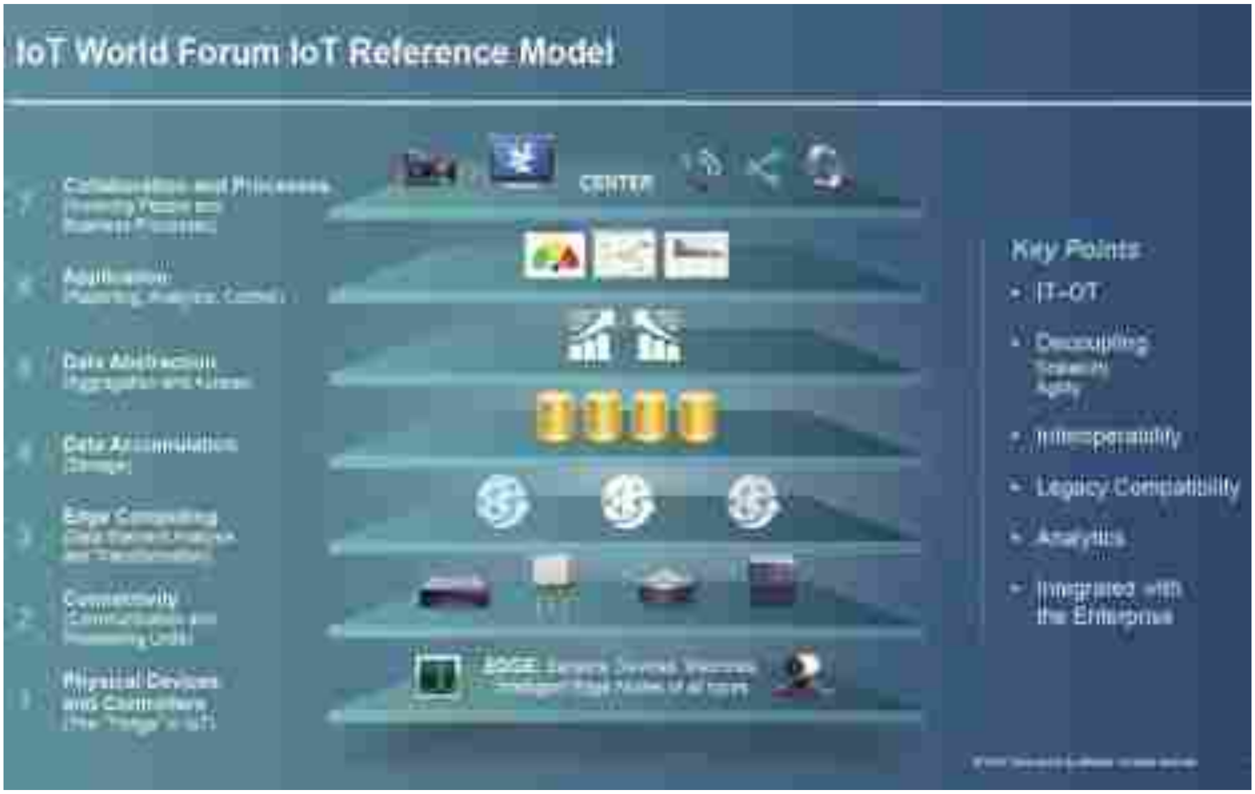
Q 1a) Define IOT and Reference model suggested by CISCO

Ans:

The Internet of Things (IoT) is the network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, actuators, and have connectivity which enables these things to connect and exchange data.

IOT reference model suggested by CISCO (Architectural view)



IoT World Forum IoT Reference Model

7 Collaboration and Processes (Involving People and Business Processes)

6 Application (Reporting, Analytics, Control)

5 Data Abstraction (Aggregation and Access)

4 Data Accumulation (Storage)

3 Edge Computing (Data Element Analysis and Transformation)

2 Connectivity (Communication and Processing Units)

1 Physical Devices and Controllers (The "Things" in IoT)

CENTER

EDGE: Sensors, Devices, Machines, Intelligent Edge Nodes of all types

Key Points

- IT-OT
- Decoupling
  Scalability
  Agility
- Interoperability
- Legacy Compatibility
- Analytics
- Integrated with the Enterprise

Architecture of IoT

- It defines basic architectural building blocks and their integration capability into multi-tiered systems.
- The reference model defining relation-ships among various IoT verticals, for example, transportation and healthcare
- Gives a blueprint for data abstraction
- Recommends quality 'quadruple' trust

- “Protection, Security, Privacy, and Safety”
- Defines no new architecture and no reinvent but existing architectures congruent with it

LAYER 1. Physical devices and controllers.

- They are the physical devices , also called as “THINGS” in IOT .
- Basically they are Embedded Devices, Embedded hardware/software like Sensors/Actuators , RFID, Hardware (Arduino, Raspberry Pi, Intel Edison, Beagle Bone Black and Wireless SoC…).
- They are ready to send and receive the information

1
Physical Devices
and Controllers
(The "Things" in IoT)
EDGE: Sensors, Devices, Machines,
Intelligent Edge Nodes of all types

LAYER 2. Connectivity (Communication and processing units)

Processing Units:

- ➢ Contains Routers and Gateways
- ➢ Main task is to deliver the right information at right time and to right machine i.e. reliable transmission.

Communication:

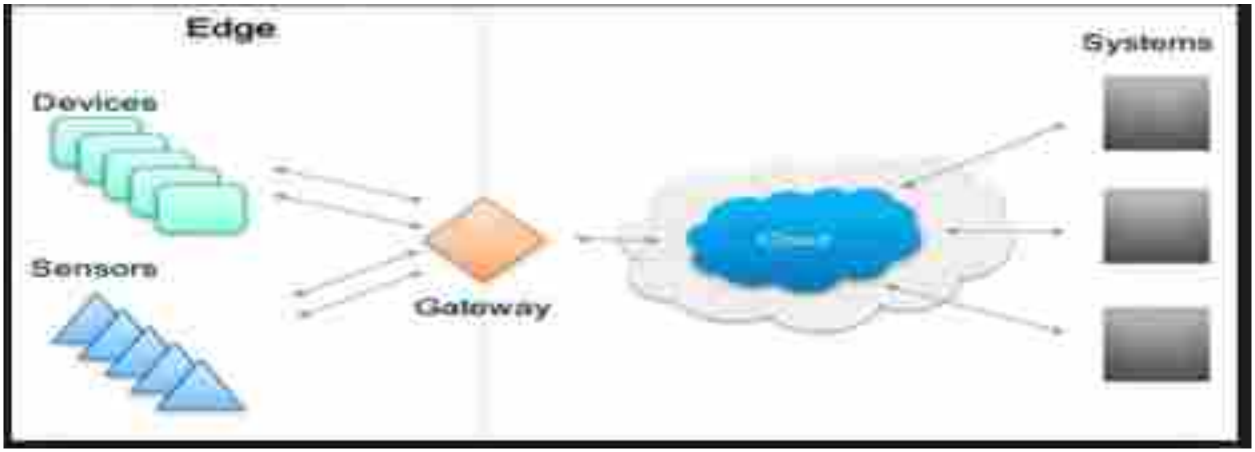
- ➢ Includes protocol handlers, message routers, message cache
- ➢ It can be between smart device and network/ internet directly

Popular Communication Protocols Used from sensors to gateways are:

- ZigBee, 6LOWPAN , Bluetooth, RFID
- WiFi, WiMax, 2G/3G/4G/5G

Layer 3 [Edge Computing Or Fog Computing]

- Edge computing is an architecture that uses edge devices / network edge like routers, gateways, switches, multiplexers, integrated access devices to do some preprocessing of data.



Layer 4 [Data Accumulation and storage]

Data management is done at backend server/cloud or data base centres

Main roles of layer 4 are:

- Convert data in motion to data at rest.
- Convert format from network packets to database relational tables.
- Convert Event based data to query based data (it bridges the gap between real time networking and non real time)
- The concept of BIG DATA is used at layer 4.

Layer 5 Data Abstraction

- Data abstraction is done at backend server/cloud or data base centres
- Abstraction means providing the essential and relevant information of the data by hiding the irrelevant one.
- Main roles are:
- Provide multiple storage systems to accommodate data from different IOT devices.
- Reconciling multiple data format from different sources.
- Combining data from multiple sources and simplifying the application i.e consolidating the data into one place.

Layer 6 Application

- Layer 6 deals with reporting, analysis and control
- i.e. the data is analysed and then send to controlling device like actuator.
- And then the data is passed to specific application like mobile application or webpage or to the business enterprise which require that data.

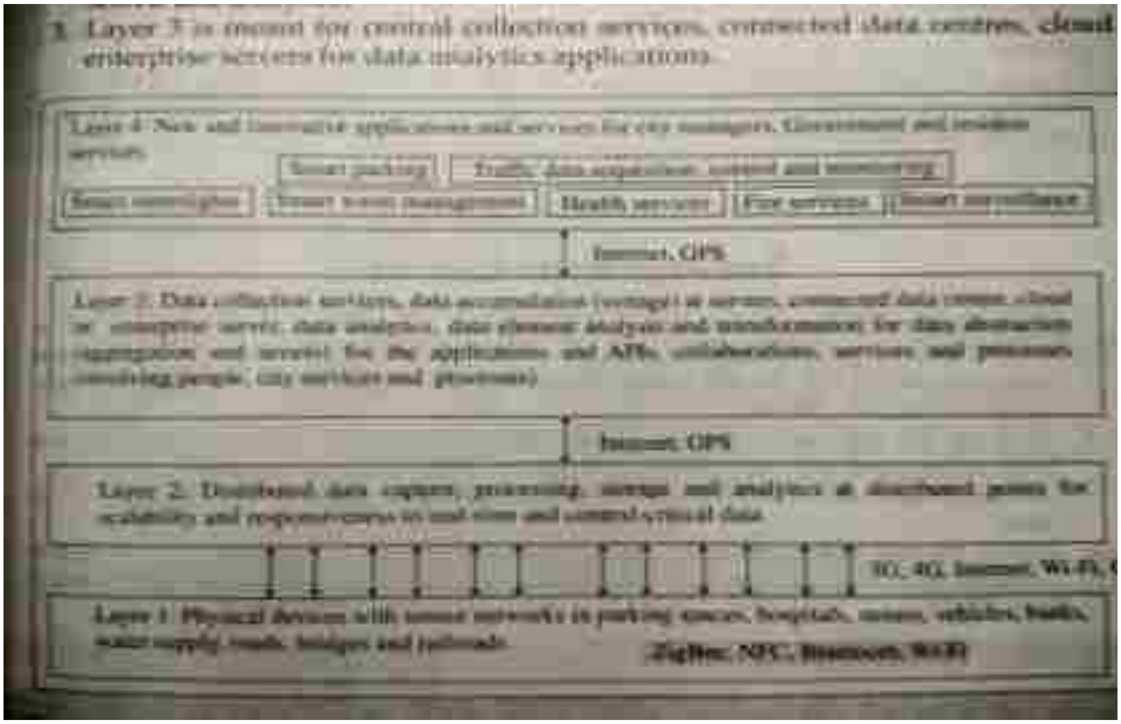
Layer 7 Collaboration and processes.

- It means involving people and business process.
- Basically multiple people are using same applications for a range of different purpose

- But in IOT the main objective is to empower people to do their work better, not the application.

Q1 b) Four layer architecture frameworks developed at CISCO for “Smart City”

Ans: Block diagram



3. Layer 3 is meant for central collection services, connected data centres, cloud enterprise servers for data analytics applications.

Layer 4: New and innovative applications and services for city managers, Government and resident services
Smart parking
Traffic data acquisition, control and monitoring
Smart streetlights
Smart waste management
Health services
Fire services
Smart surveillance
Internet, GPS
Layer 3: Data collection services, data accumulation (storage) at server, connected data centre, cloud or enterprise server, data analytics, data element analysis and transformation for data abstraction, aggregation and service for the applications and APIs, collaborations, services and processes involving people, city services and processes)
Internet, GPS
Layer 2: Distributed data capture, processing, storage and analytics at distributed points for scalability and responsiveness in real-time and control-critical data
3G, 4G, Internet, Wi-Fi, C
Layer 1: Physical devices with sensor networks in parking spaces, hospitals, streets, vehicles, banks, water supply, roads, bridges and railroads
ZigBee, NFC, Bluetooth, WiFi

Layer1:

- It consists of sensors, sensor networks and devices network in parking spaces, hospitals, streets, vehicles, water supply, roads, bridges and railroads. Bluetooth, Zigbee, NFC, WiFi are the protocols are used at this layer

Layer2:

- It captures the data at distributed computing points where data is processed, stored and analysed.

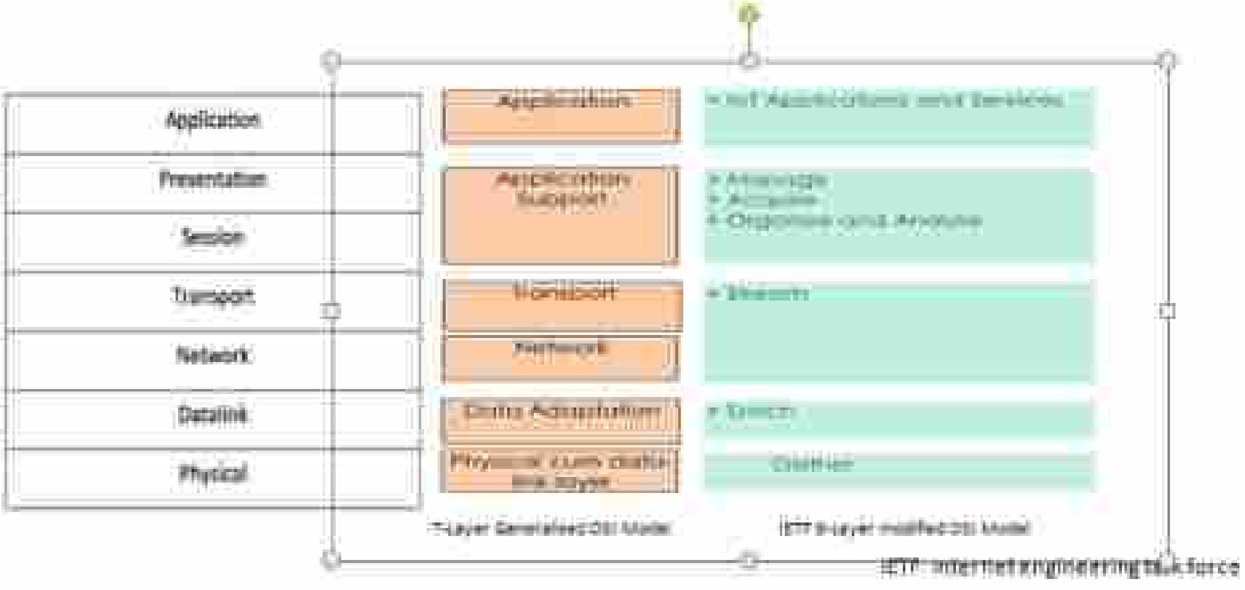
Layer3:

- Layer 3 is responsible for central collection services, connected data centres, cloud nad enterprise servers for data analytics applications.

Layer4:

- It consists of new innovative applications, such as waste containers monitoring, WSNs for power loss monitoring, bike sharing management and smart parking

Q1 c) Modified OSI model for the IOT/M2M Systems



IETF: internet engineering task force

Physical cum data link layer

- It is also called perception layer or object layer or device layer.
- The physical layer consists of the physical device called as 'Things' which can be sensors, actuators, RFID tags.
- These devices collects and gather the raw data from the environment, do some processing and have computational capabilities.

Data Adaptation layer

- The main function of data adaptation layer is data enrichment.
- Gateway is considered to work at data adaptation layer.
- Gateway and network layer and transport layer
- Both the layers deals with Data streaming i.e. transfer of data at a
- steady high-speed rate sufficient to support such applications as high-

- definition television (HDTV)

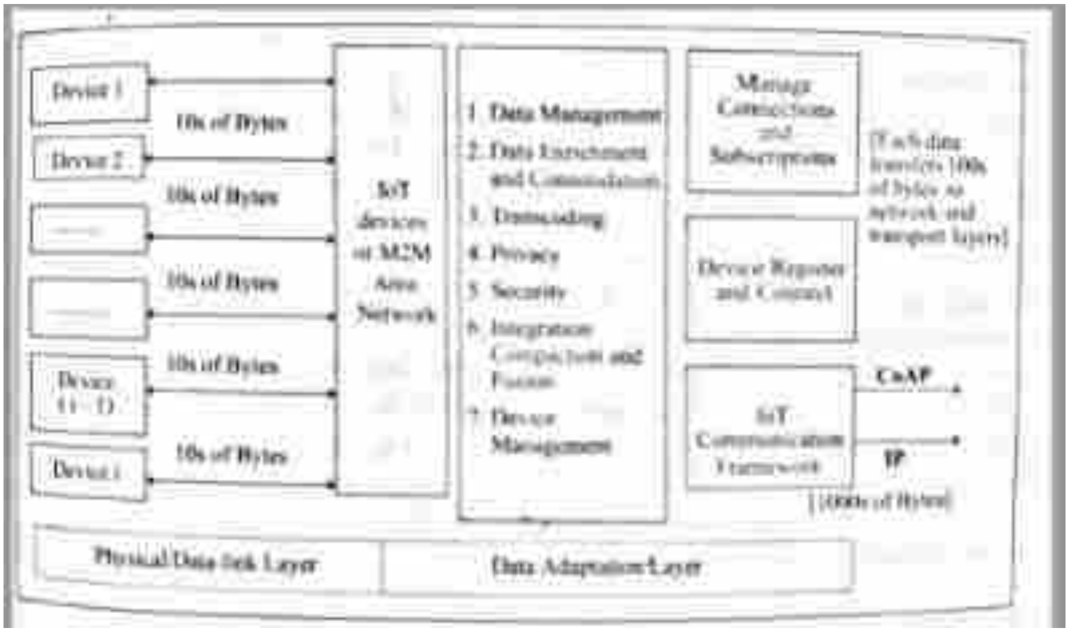
Transport layer

- The main functions of this layer are:
- Forwarding the data coming from sensors to the next upper management layer.
- It provides sufficient security features, bandwidth management etc.
- Application support layer/middleware layer:
- This layer is also called as service management layer or processing layer.
- It is called as application support as it allows the IOT application programmer to work with heterogeneous objects without consideration to specific hardware.

Application layer/business layer:

- This layer is responsible for building the business model, making reports, graphs, flowcharts,
- This layer also supports decision making process based on big data analysis.
- The various protocols which work at the App layer are HTTP/COAP, MQTT, AMQP etc

Q 2 a) Data Enrichment, data consolidation and device management at gateway


Device 1
Device 2
Device (i - 1)
Device i
10s of Bytes
10s of Bytes
10s of Bytes
10s of Bytes
10s of Bytes
IoT devices at M2M Area Network
1. Data Management
2. Data Enrichment and Consolidation
3. Transcoding
4. Privacy
5. Security
6. Integration Compaction and Fusion
7. Device Management
Manage Connections and Subscriptions
[Each data transfers 100s of bytes as network and transport layers]
Device Register and Control
IoT Communication Framework
CoAP
IP
[1000s of Bytes]
Physical Data link Layer
Data Adaptation Layer

Data Management and Consolidation Gateway functions

Transcoding:

- It involves change of data, protocol, format or code received form the IOT device in the format which is acceptable at the server and vice versa.
- It also involves filtering, compression, decompression etc.

Privacy:

- It defines that the as the data must be protected from conscious or unconscious transfer to untrustworthy destination using the internet.
- And it ensures that the stakeholder in future should not misuse the device end data or

application data.

IOT Gateway:

- The IOT gateway is considered to work at the adaptation layer of M2M architecture.
- It can be hardware or software or combination of both.
- The main functions of gateway at adaptation layer are:

Data management

- Data enrichment and consolidation
- Transcoding
- Privacy
- Security
- Integration, compaction and fusion
- Device management

Q 2 b) COAP (Constrained Application Protocol

- It is lightweight application layer protocol and web transfer protocol
- Used for the constrained network i.e. it is designed for the transportation of small data between resource constrained nodes

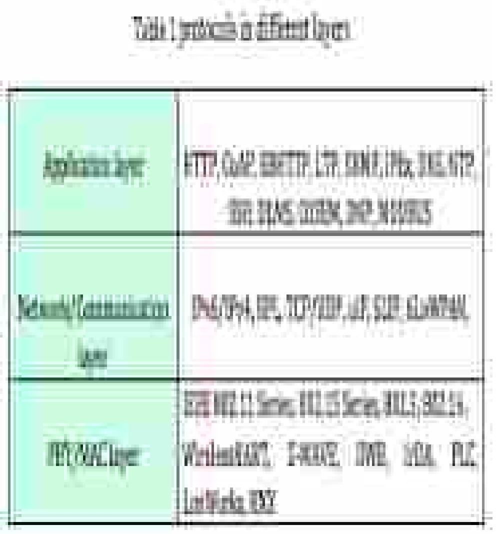


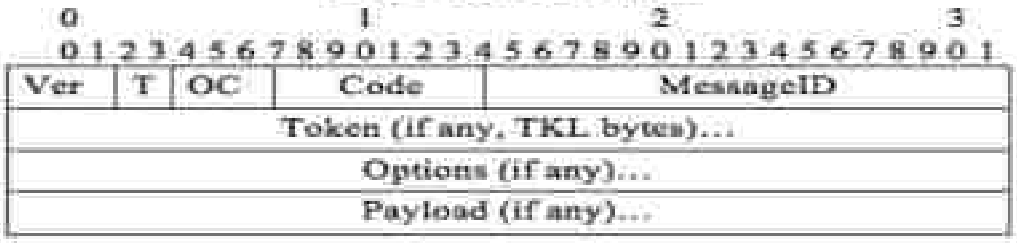
Table 1 protocols in different layers

Application layer	HTTP, CoAP, EBHTTP, LTP, SNMP, IPfix, DNS, NTP, SSH, DLMS, COSEM, DNP, MODBUS
Network/Communication layer	IPv6/IPv4, RPL, TCP/UDP, uIP, SLIP, 6LoWPAN
PHY/MAC layer	IEEE 802.11 Series, 802.15 Series, 802.3, 802.16, WirelessHART, Z-WAVE, UWB, IrDA, PLC, LonWorks, KNX

- COAP is used for CORE using ROLL network.

Message format of COAP

Table 3 Message Format

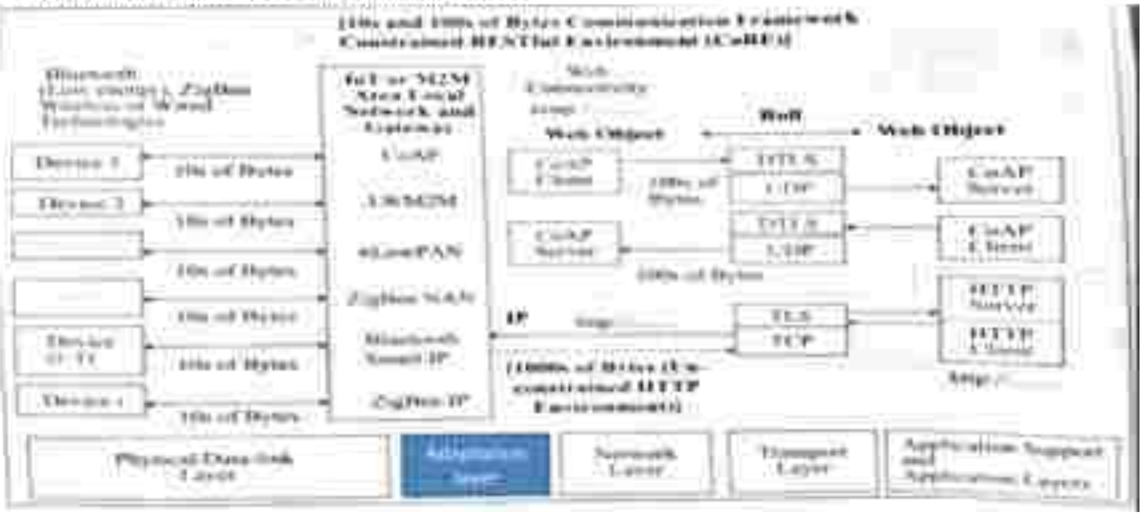


0 1 2 3

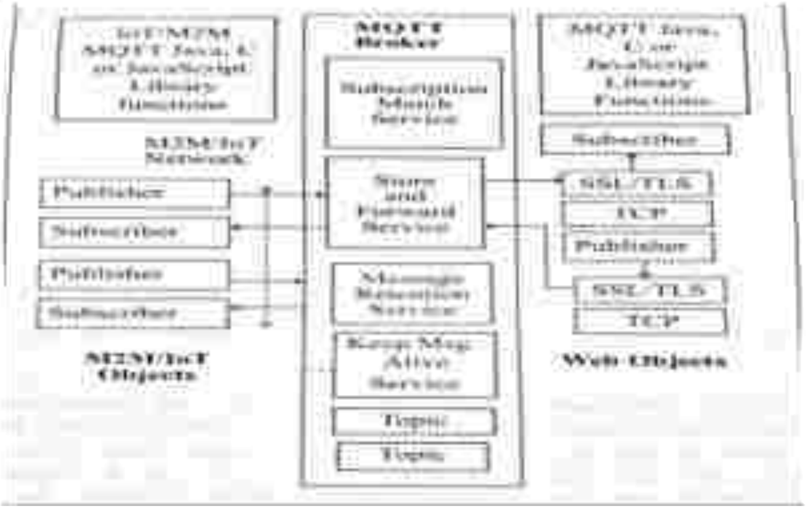
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

Ver	T	OC	Code	MessageID
Token (if any, TKL bytes)…				
Options (if any)…				
Payload (if any)…				

Web Communication Protocol for connected devices



Q 2 c) MQTT (Message Queuing Telemetry Transport)



- Designed to provide connectivity of M2M/IOT objects, web objects using different protocols.

MQTT BROKER :

Store and forward service:

- Publishers/ Subscribers can be M2M/IOT objects which communicate by using network protocols like Zigbee Or it can be Web Objects which communicate over IP network using SSL and TLS security protocols.
- The figure shows device objects and Web objects uses MQTT Java, C or JavaScript Library functions.

Subscription Match Service:

- It matches the subscriptions and forward it to right subscribers in order to route the message to the right endpoints.

Message Retention Service:

- When this field is set the broker retains the last receives message from the publisher for the new connected subscriber on the same topic.

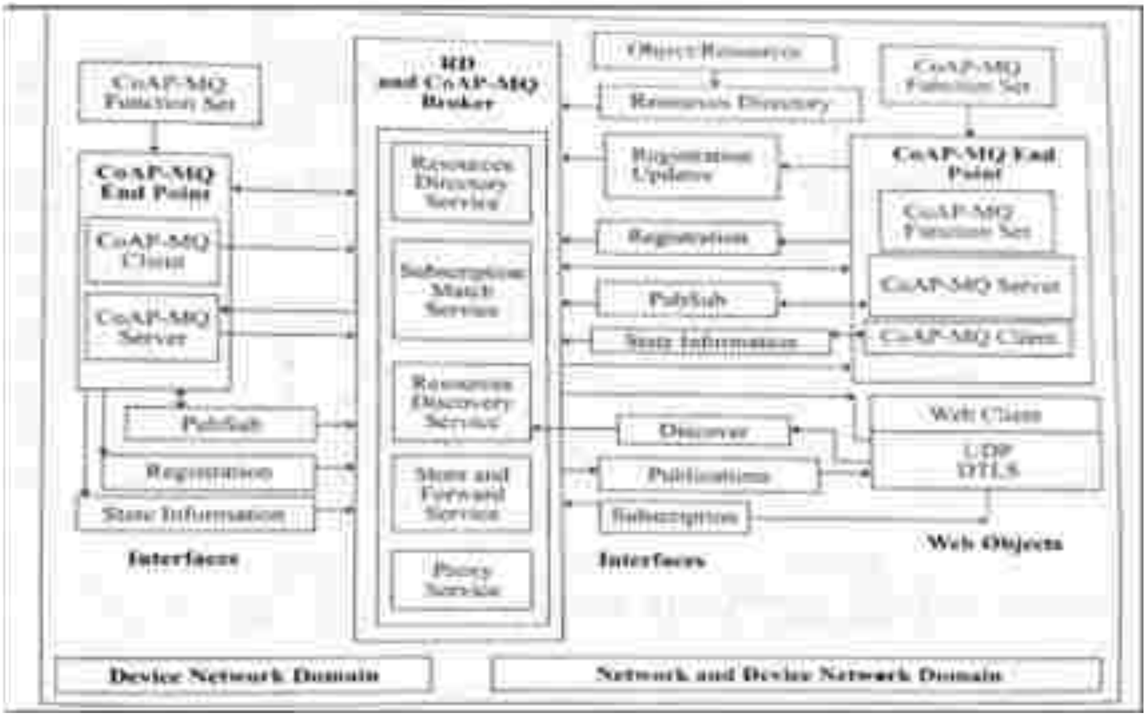
Keep Message Alive:

- Until the DISCONNECT message is received the message is saved by setting the "Keep Message Alive Service"

CoAP-MQ (Constrained Application Protocol –Message Queuing)

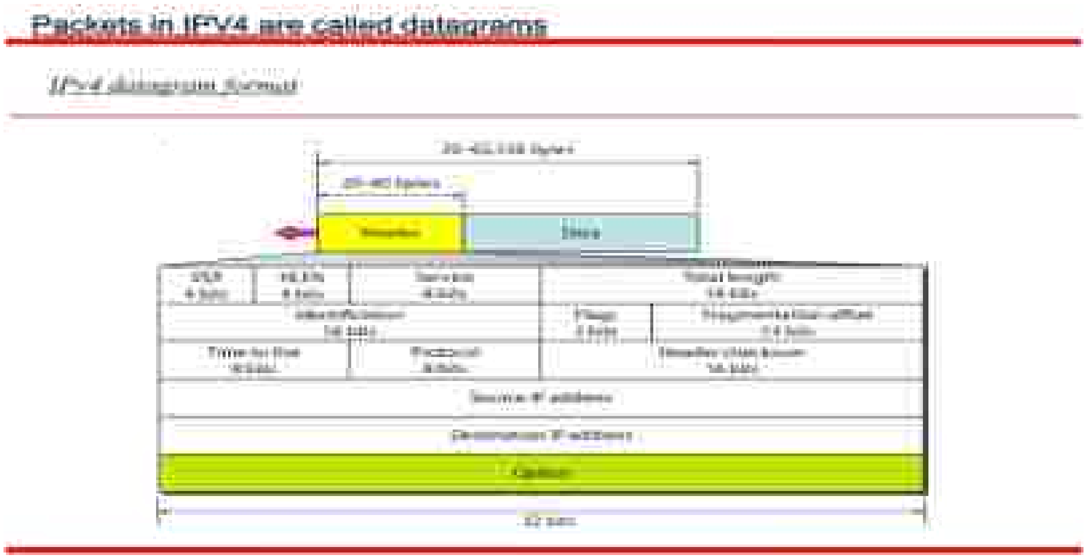
- The Constrained Application Protocol (CoAP) supports machine to machine communication across networks of constrained devices.
- The protocol is used for class of constrained devices includes devices that run for years from a small battery, and spend most of their time in a sleeping state with no network connectivity.

CoAP-MQ (Constrained Application Protocol – Message Queuing)



Q 3 a) IPv4

- IPv4 is an unreliable and connectionless datagram protocol
- If reliability is required, IPv4 must be paired with a reliable protocol such as TCP.
- This means that each datagram is handled independently, and each datagram can follow a different route to the destination and can arrive out of order.
- IPv4 provides no error control or flow control
- IPv4 relies on a higher-level protocol to take care of all these problems.
- Packets in IPV4 are called datagrams



Packets in IPV4 are called datagrams

IPv4 datagram format

Version (VER). This 4-bit field defines the version of the IPv4 protocol.

- ➢ Header length (HLEN). This 4-bit field defines the total length of the Datagram header in 4-byte words.
- ➢ This field is needed because the length of the header is variable (between 20 and 60 bytes).

I.e minimum 20 bytes and maximum 60 bytes

Types of service field

1	2	3	4	5	6	7
Precedence			Type of service			

Precedence : It is a 3 bit field which treats high priority packets as more important than other packets.

Type of Service: The last bit of Type of Service (bit 7) was defined as "Must Be Zero".

The TOS field specifies datagram's priority and request a route for low-delay, high-

throughput, or highly-reliable service

Services. This is 8-bit field,

Previously it is called as type of service and now is called differentiated services.

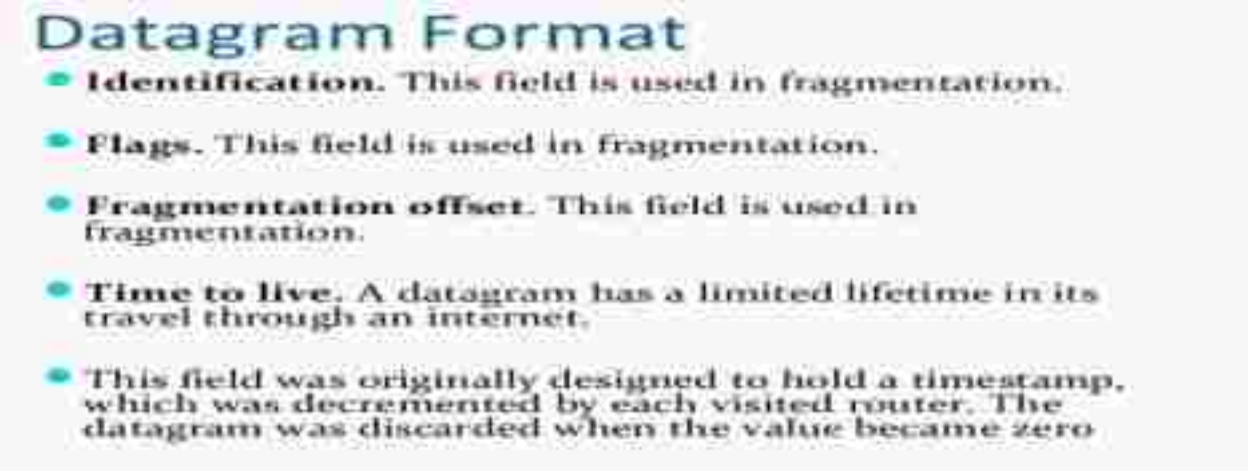
Type of service

-It assigns the priority to each IP packet

-Request specific treatment such as high throughput

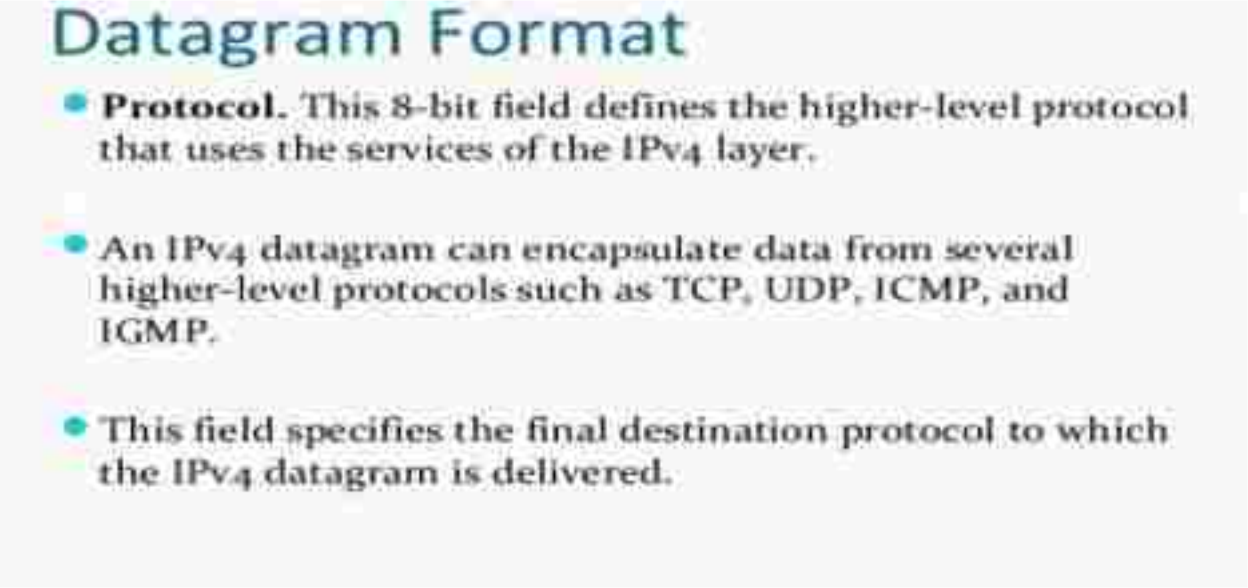
-Request a route for low-delay

-high reliability or low latency etc...

- 

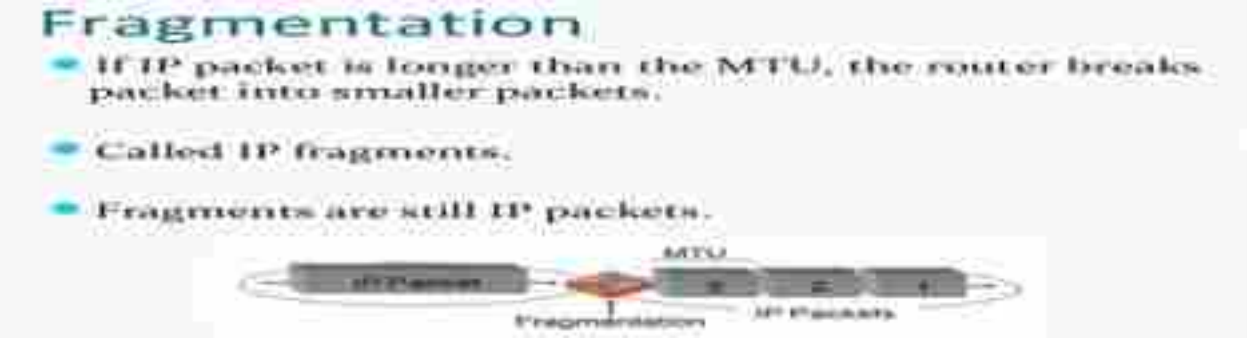
Datagram Format

- **Identification.** This field is used in fragmentation.
- **Flags.** This field is used in fragmentation.
- **Fragmentation offset.** This field is used in fragmentation.
- **Time to live.** A datagram has a limited lifetime in its travel through an internet.
- This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero



Datagram Format

- **Protocol.** This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer.
- An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP.
- This field specifies the final destination protocol to which the IPv4 datagram is delivered.



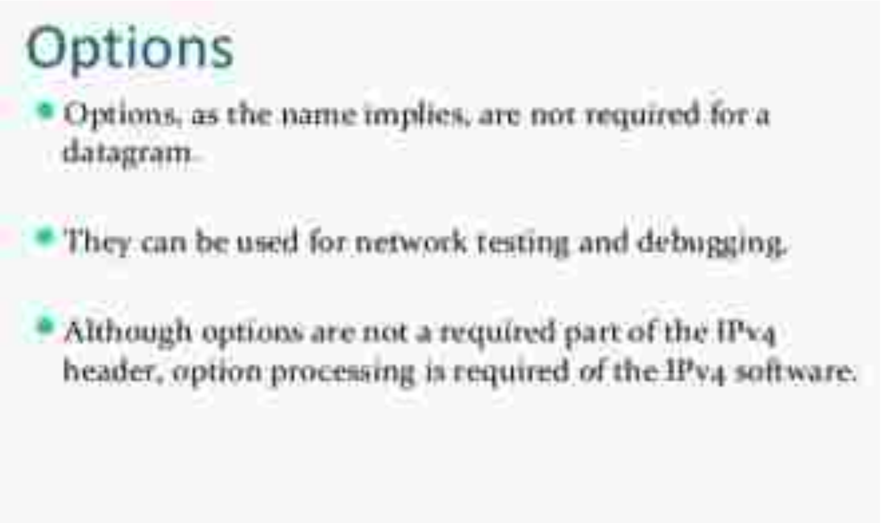
Fragmentation

- If IP packet is longer than the MTU, the router breaks packet into smaller packets.
- Called IP fragments.
- Fragments are still IP packets.

IP Packet
MTU
3
2
1
Fragmentation
IP Packets

- Source address. This 32-bit field defines the IPv4 address of the source. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.
- Destination address. This 32-bit field defines the IPv4 address of the destination.

- ➢ This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.



Options

- Options, as the name implies, are not required for a datagram.
- They can be used for network testing and debugging.
- Although options are not a required part of the IPv4 header, option processing is required of the IPv4 software.

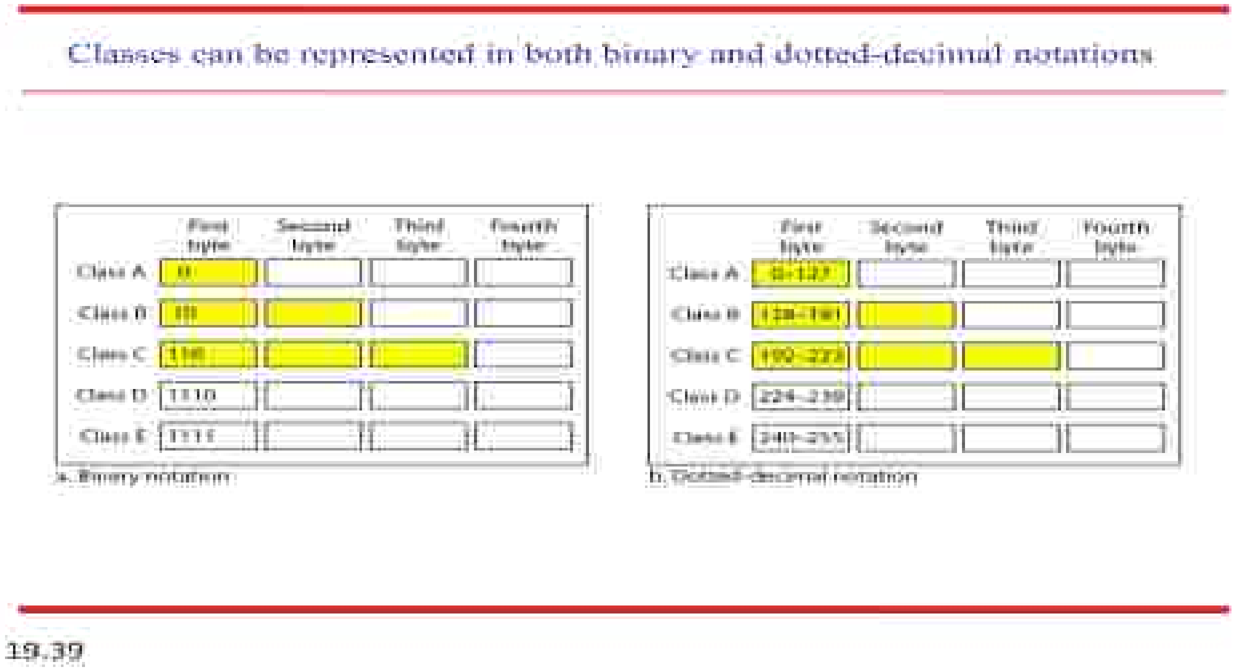
Q 3 b) Addressing

- Classful addressing
- Classless addressing
- Classful addressing:

➢ In classful addressing, the address space of IPv4 is divided into five classes:

A, B, C, D, and E.

➢ Each class has a number of blocks



Classes can be represented in both binary and dotted-decimal notations

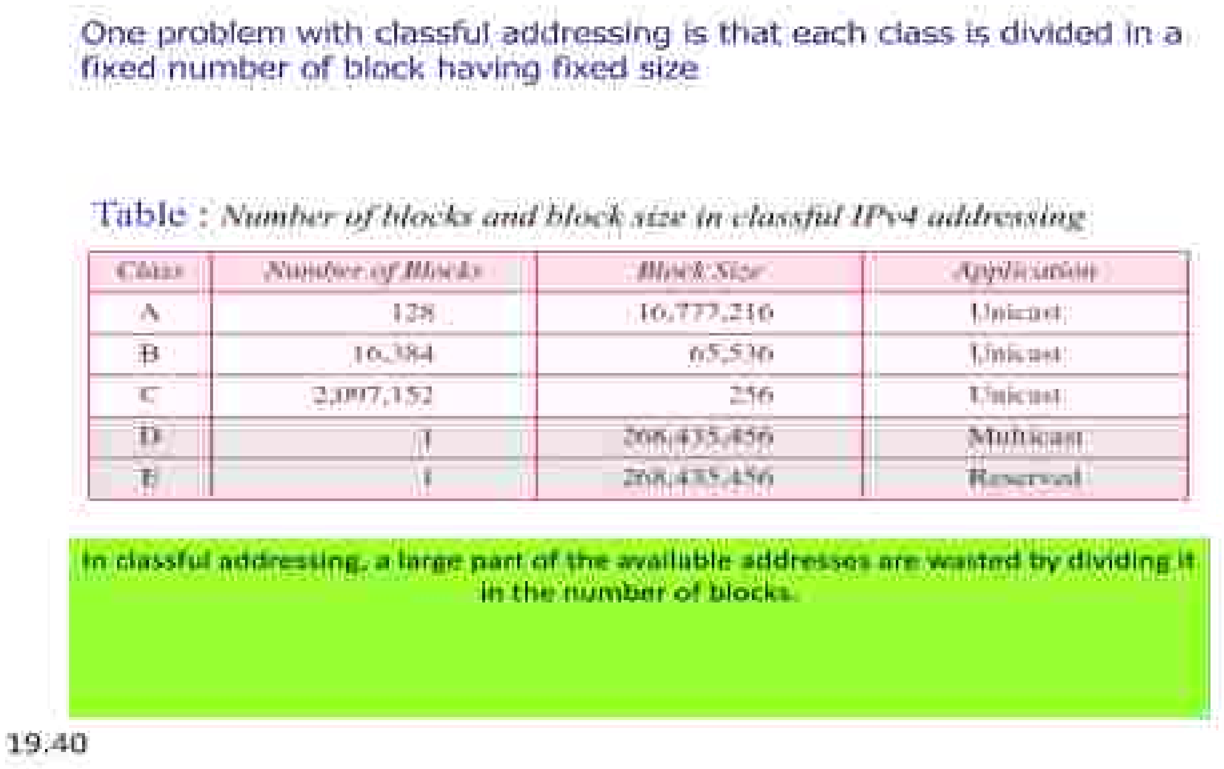
	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

19.39



One problem with classful addressing is that each class is divided in a fixed number of block having fixed size

Table : *Number of blocks and block size in classful IPv4 addressing*

Class	*Number of Blocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

In classful addressing, a large part of the available addresses are wasted by dividing it in the number of blocks.

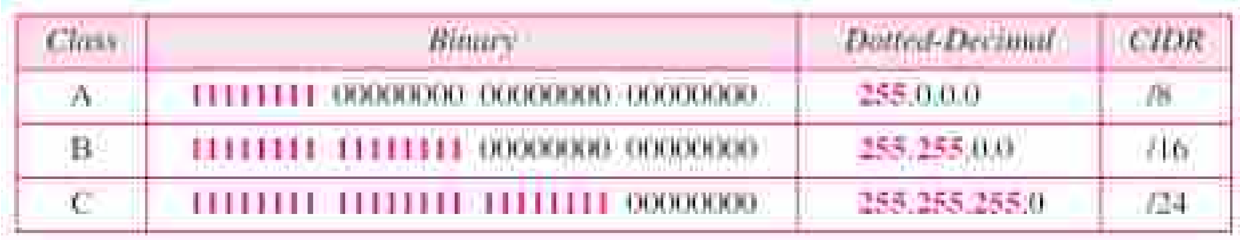
19.40

Netid and Hostid:

IP address in class A, B, or C is divided into netid and hostid.

- These parts are of varying lengths, depending on the class of the address

- In class A: one byte defines the netid and three bytes define the hostid.
- In class B: Two bytes define the netid and two bytes define the hostid.
- In class C: Three bytes define the netid and one byte defines the hostid



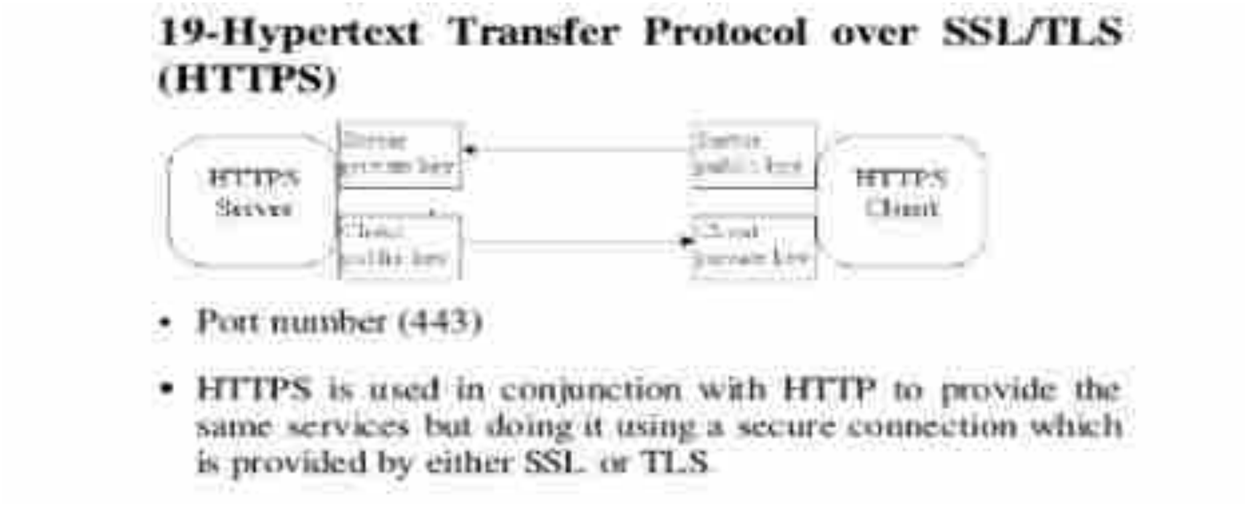
Class	Binary	Dotted-Decimal	CIDR
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

- Classless addressing: In classless addressing the organizations are not divided in class

  Any entity or any organization, small or large, which wants to access the Internet, is granted a block (range) of addresses

- The size of the block (the number of addresses) varies based on the nature and size of the entity.
- For example, a household may be given only two addresses; a large organization may be given thousands of addresses.
- An ISP, as the Internet service provider, may be given thousands or hundreds of thousands based on the number of customers it may server.
- In IPv4 addressing, a block of addresses can be defined as x.y.z.t /n In which x.y.z.t defines one of the addresses and the /n defines the mask.

Q 3 c) HTTPS

- 

19-Hypertext Transfer Protocol over SSL/TLS (HTTPS)

HTTPS Server
Server private key
Client public key
Server public key
Client private key
HTTPS Client

- Port number (443)
- HTTPS is used in conjunction with HTTP to provide the same services but doing it using a secure connection which is provided by either SSL or TLS

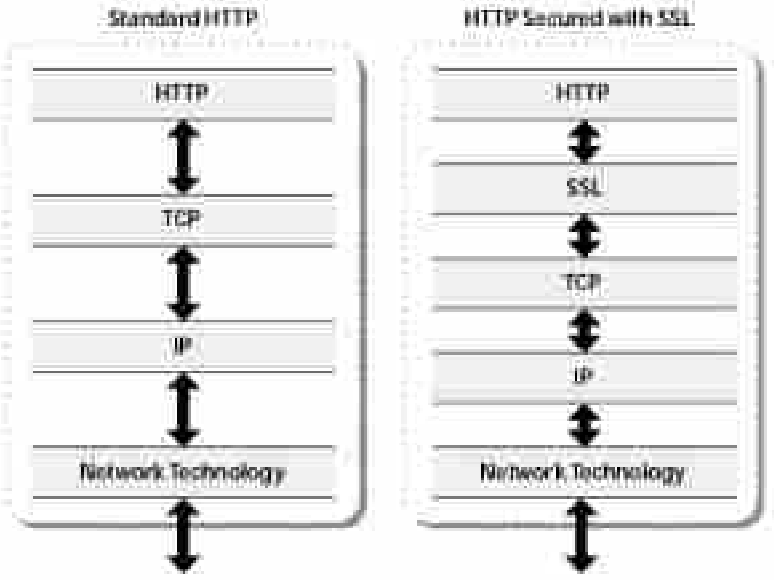


Figure 4.10 ▶ The SSL protocol inserts itself between an application like HTTP and the TCP transport layer. TCP sees SSL as just another application, and HTTP communicates with SSL much the same as it does with TCP.

- (HTTPS) Hypertext Transfer Protocol over Secure Socket Layer (SSL).
- First implementation of HTTP over SSL was issued in 1995 by Netscape.

Q 4 a) Cloud computing is a paradigm that allow multiple clients to access the network and share computing resources on-demand

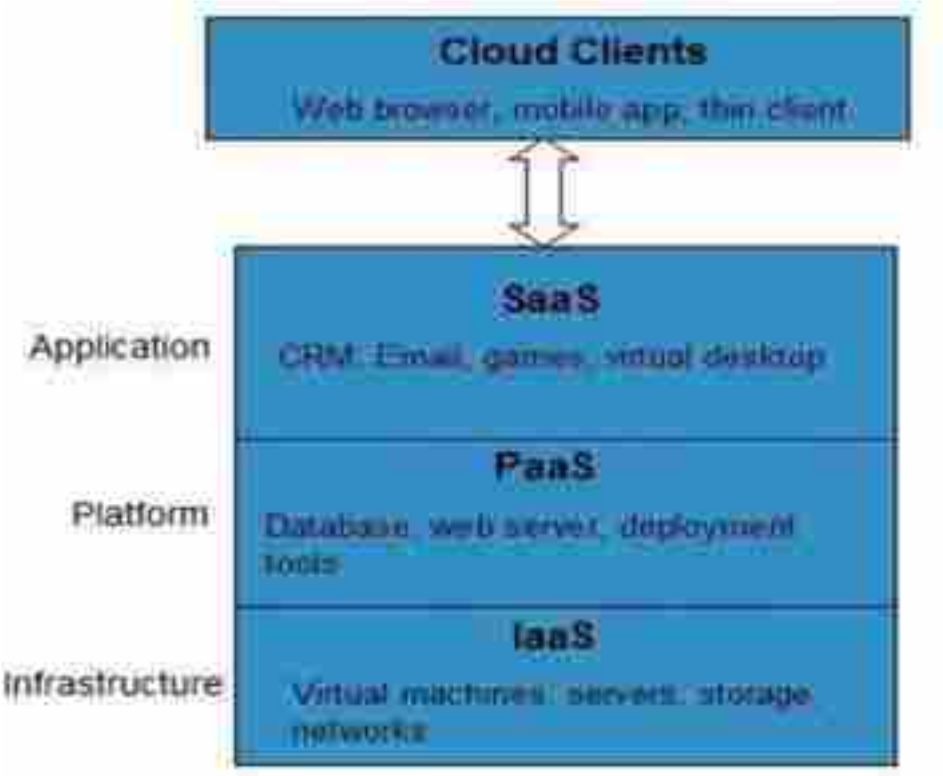
- Cloud computing is a method in which resources are retrieved from the Internet through web-based tools and applications, without using direct connection to a server.
- Cloud-based storage makes it possible to save the files to a remote database instead of keeping them on a hard drive or local storage device.

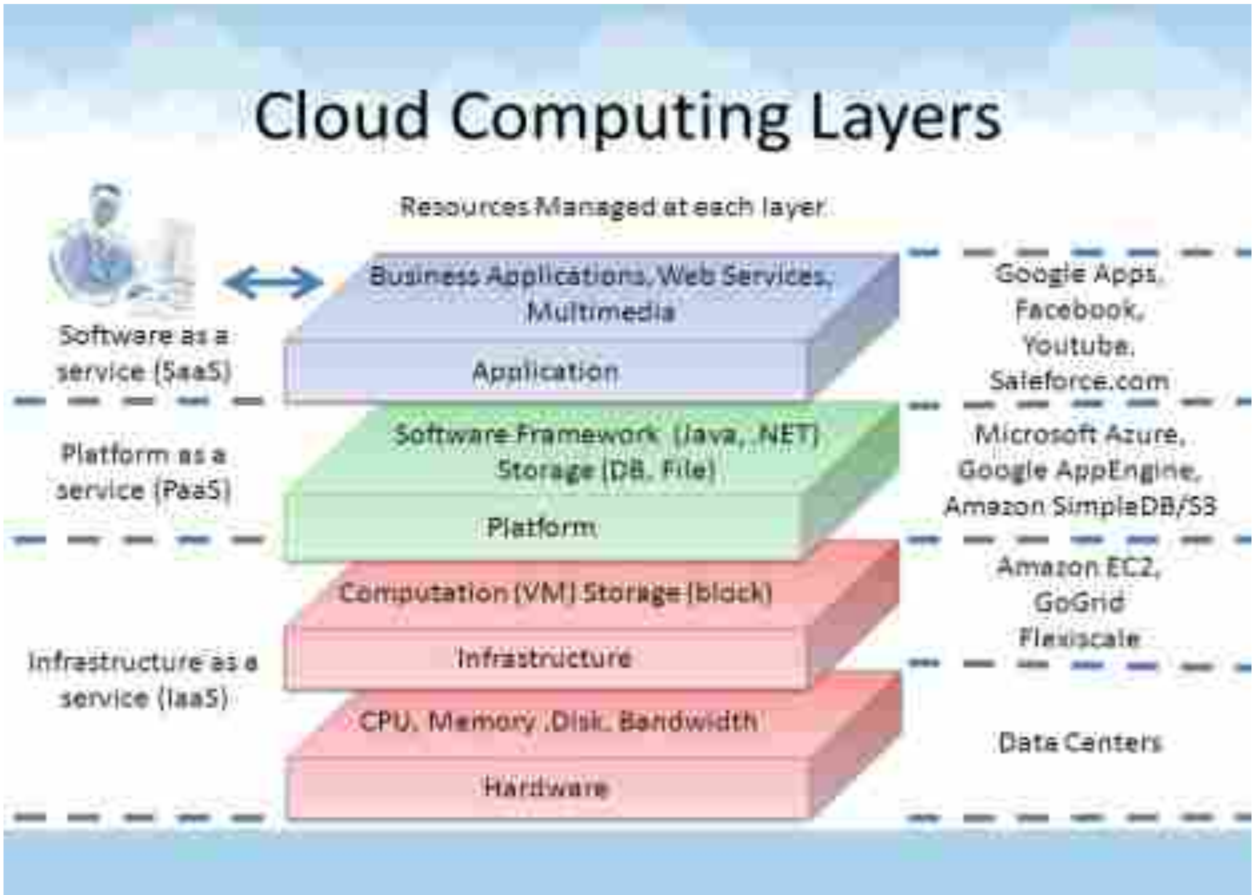
Q 4 b) Service Models

- Cloud computing is not a single piece of technology, like a microchip or a cell phone. Rather, it's a system, primarily comprised of three services:
- Service Models are the reference models on which the Cloud Computing is based. These can be categorized into three basic service models as listed below:

  1. Infrastructure as a Service (IaaS)
  2. Platform as a Service (PaaS)
  3. Software as a Service (SaaS)

Three basic service layers


Cloud Clients
Web browser, mobile app, thin client
Application
SaaS
CRM, Email, games, virtual desktop
Platform
PaaS
Database, web server, deployment tools
Infrastructure
IaaS
Virtual machines, servers, storage networks


Cloud Computing Layers

Resources Managed at each layer
Software as a service (SaaS)
Business Applications, Web Services, Multimedia
Application
Google Apps, Facebook, Youtube, Saleforce.com
Platform as a service (PaaS)
Software Framework (Java, .NET) Storage (DB, File)
Platform
Microsoft Azure, Google AppEngine, Amazon SimpleDB/S3
Infrastructure as a service (IaaS)
Computation (VM) Storage (block)
Infrastructure
Amazon EC2, GoGrid Flexiscale
CPU, Memory ,Disk, Bandwidth
Hardware
Data Centers

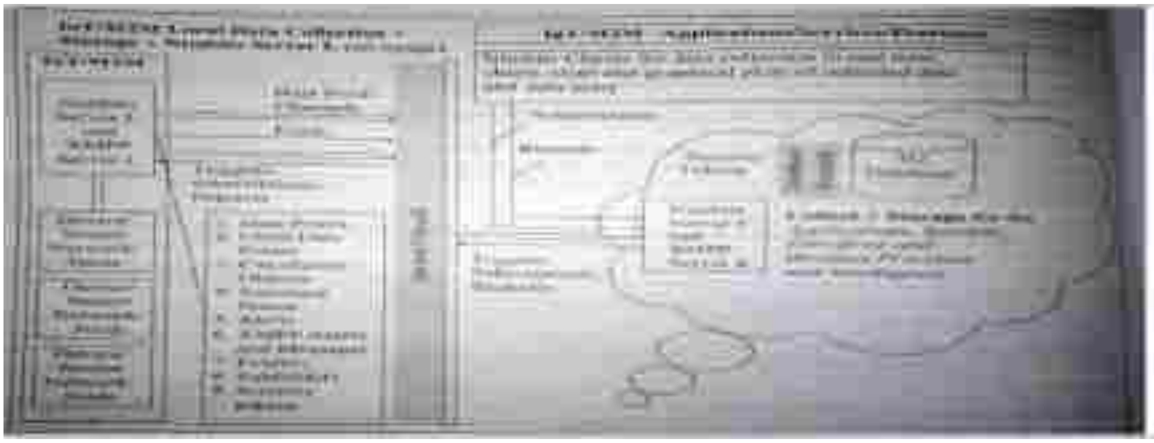
- The top layer is Software-as-a-Service (SaaS).
- SaaS supports accessing user's applications through a browser without the knowledge of Hardware or Software to be installed.
- Middle layer is Platform-as-a-Service (PaaS) which mainly supports deployment and dynamic scaling of .NET, Python and Java based applications.
- One such an example of PaaS is Google App Engine.
- Bottom layer contains basic hardware resources like Memory, Storage Servers. Hence it is denoted as Infrastructure-as-a-Service (IaaS). For example Amazon easy Storage Service (S3) and Amazon Elastic Compute Cloud (EC2).

Q4 c) Nimbits

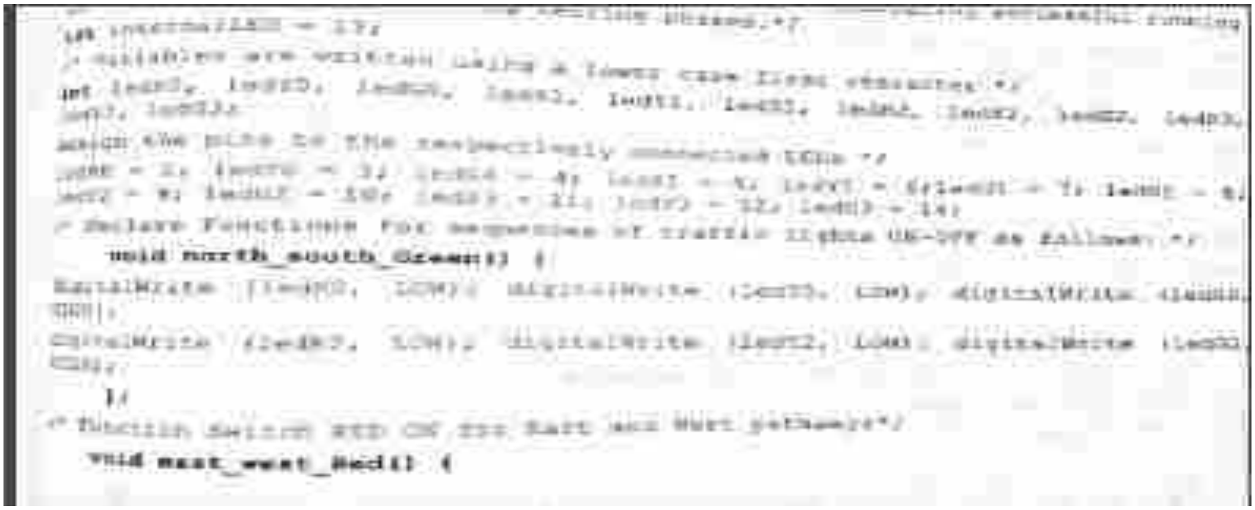
- It can be used to record and share data points on the cloud and lets users record their changing numeric, text based, GPS, json or xml values into them.
- The API lets users access the public server to push and pull their data from.
- The API also provides access to a chart image service, which can generate PNG format images of user data. The API uses HTTP calls and responses are formatted in JSON or TXT.

The basic services offered by Nimbits are

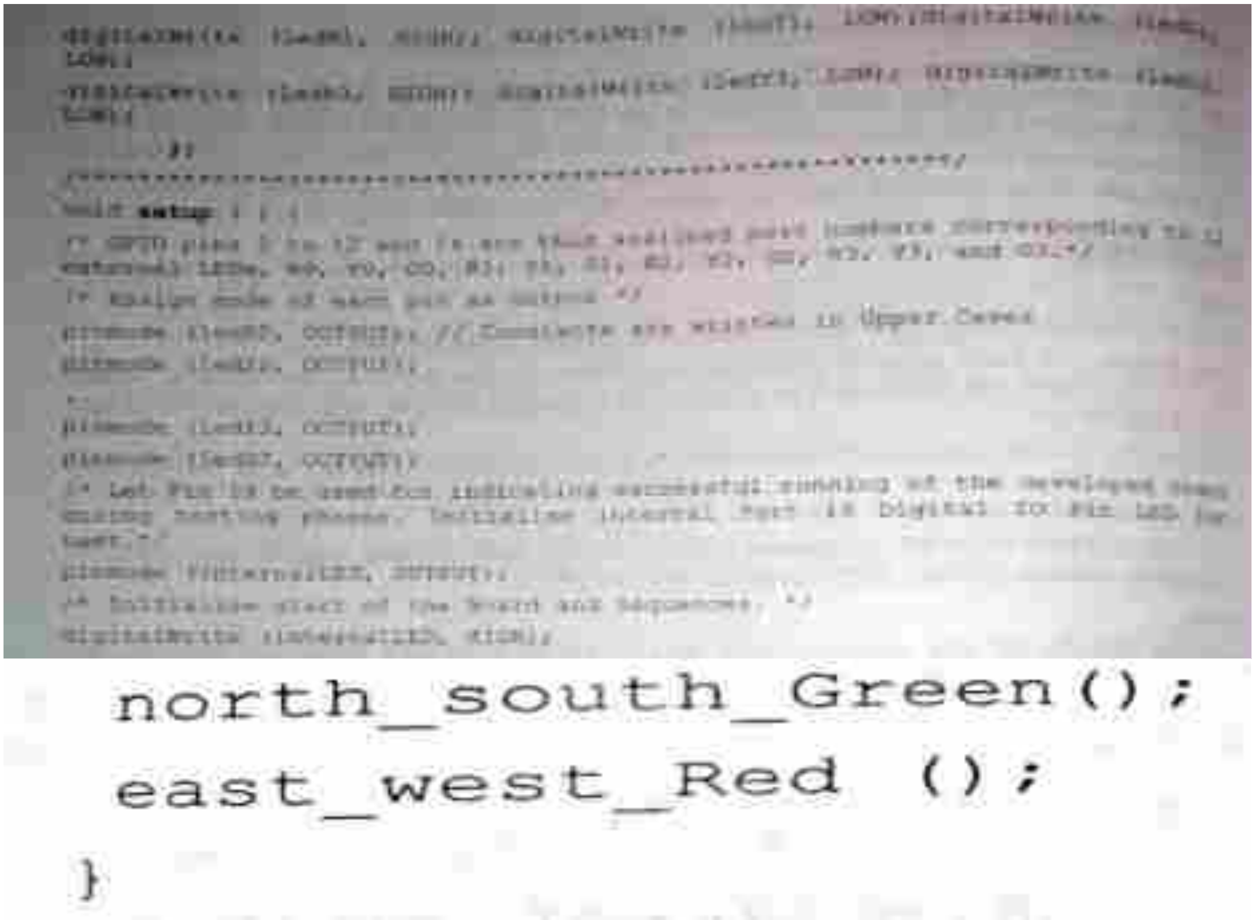
- It provides edge computing locally on embedded systems, build up on local application.
- Nimbits enables Iot an open source distributed cloud.
- It pushes important data to the cloud when connected to internet.
- It supports multiprogramming languages, including Arduino, push functions from Arduino cloud, javascript, HTML.
- It provides rule engine for connecting sensors, persons and software to cloud and to one another.
- Nimbits data points can relay data between the software systems or hardware devices like Arduino, using cloud as backend.



Q 5 a) Traffic light control using arduino



```
int [illegible]LED = 13;          /* [illegible] phases,*/   [illegible] the sequential timing
/* Variables are written using a lower case first character */
int ledN2,  ledS3,  ledN4,  leds1,  ledT1,  ledE1,  ledW4,  ledW2,  ledE2,  ledN3,
ledW3, ledS2;
Assign the pins to the respectively connected LEDs */
ledN = 2; ledS3 = 3; ledN4 = 4; leds1 = 5; ledT1 = 6;ledE1 = 7; ledW2 = 8;
ledE2 = 9; ledW2 = 10; ledE3 = 11; ledW3 = 12; ledN3 = 14;
/* Declare Functions for sequences of traffic lights ON-OFF as follows */
   void north_south_Green() {
digitalWrite (ledN2,  LOW); digitalWrite (ledS3, LOW); digitalWrite (ledN4,
HIGH);
digitalWrite (ledE2,  LOW); digitalWrite (ledW2, LOW); digitalWrite (ledS3,
LOW);
   };
/* Function switch RED ON the East and West pathways*/
  void east_west_Red() {
```



```
digitalWrite (Led[illegible]), HIGH); digitalWrite (Led[illegible]), LOW);digitalWrite (Led[illegible],
LOW);
digitalWrite (Led[illegible], HIGH); digitalWrite (Led[illegible], LOW); digitalWrite (Led[illegible],
LOW);
        });
/**************************************************************************/
void setup ( ) {
/* GPIO pins 2 to 12 and [illegible] are thus assigned port hardware corresponding to [illegible]
external LEDs, [illegible], and [illegible].*/
/* Assign mode of each pin as output */
pinMode (Led0, OUTPUT); // Constants are written in Upper Cases
pinMode (Led1, OUTPUT);
..
pinMode (Led13, OUTPUT);
pinMode (Led[illegible], OUTPUT);
/* Let Pin 13 be used for indicating successful running of the developed code
during testing phase. Initialise internal Port 13 Digital IO Pin LED for
test.*/
pinMode (internalLED, OUTPUT);
/* Initialise start of the Board and Sequences. */
digitalWrite (internalLED, HIGH);
north_south_Green();
east_west_Red ();
}
```

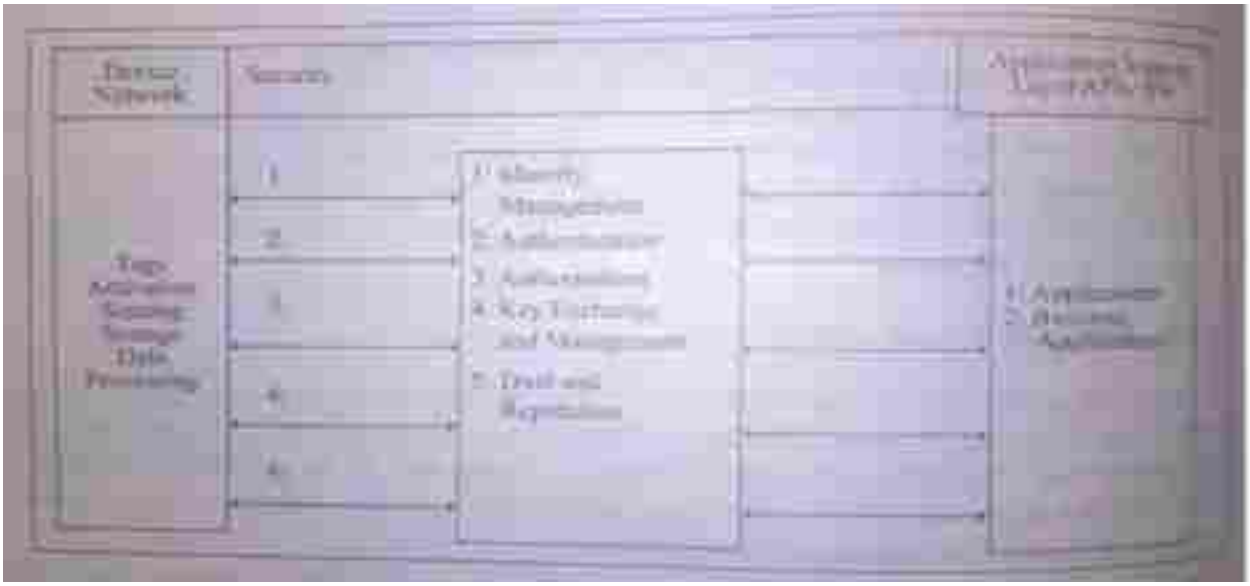
Q 5 b) Eclipse IOT stack

- It is open source IoT technology that enables the development of software for all the five levels.
- It has many features like:
- It is set of Java frameworks, protocols, development tools OSGi services (Open services gateway initiative).
- It consists of the components and frameworks for IOT solutions.
- Reading data from sensors and devices.
- Sensors senses the analog data and send the analog data to 10 bit ADC (Analog to digital converter).
- ADC send the 10 bit parallel data to PISO converter.
- Parallel IN serial out (PISO) converter converts parallel data to serial data.

- This serial output connects to SPI (Serial Peripheral Interface) Pin of Arduino board.

Q 6 a) Security requirements

- Security functional group contains five sets of functions which are required for ensuring security and privacy. Five functional components of security are defined in IOT are:
- Identity management
- Authentication
- Authorization
- Key exchange and management
- Trust and reputation



<u>Identity Management:</u>

- An object's identity should always be unique compared to the other objects from its family.
- Unique identity can be called core identity, as an object can also have several temporary identities.
- Secure Authentication/Authorization
- IoT devices should be authenticated with strong usernames IDs and password before being allowed to communicate with other IoT devices on the network.

- Authentication token/session should always be unique to each user along with user id, app id and device id.

Key exchange and management

- Public key cryptography also known as asymmetric encryption keeps the data safe and secure during transmissions.
- In a public key encryption system, any person can encrypt a message using the receiver's public key. That encrypted message can only be decrypted with the receiver's private key.

Trust and reputation

- Trusted IoT Device:
- The trusted IoT devices should be able to communicate with the intended hosting services only.
- And the firmware / software should be frequently updated.

Threat Analysis

- Threat analysis means uncovering the security design flaws after specifying stride category, data flow diagram, elements between which interactions occur during the stride and specified the processes which are activated for analysis.

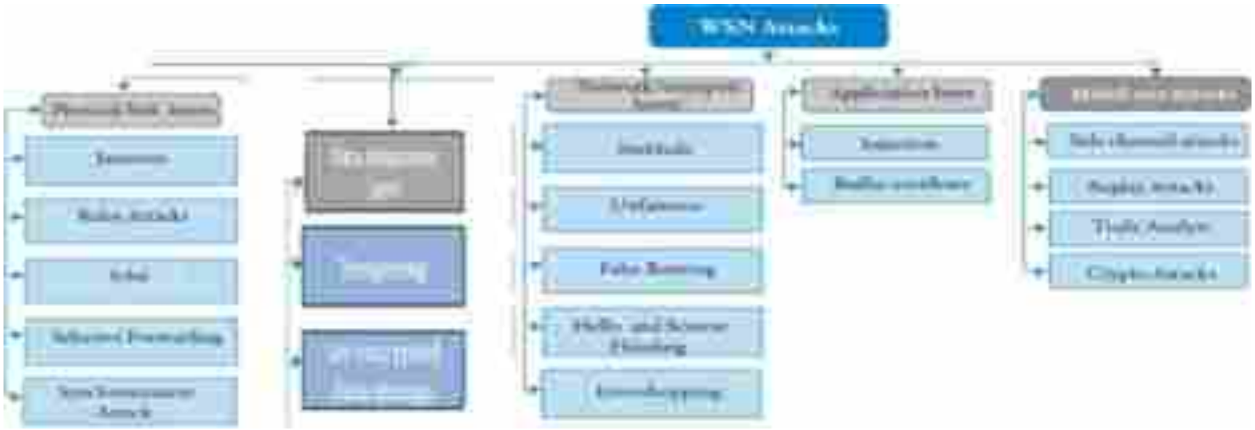
Q 6 b) Layered attacker model

Network tomography

- It refers to the study of vulnerabilities and security aspects for network monitoring in complex system such as RFID, WSN or IoT networks.
- It also helps in observing each network section.

Security Tomography:

- It enables finding the attack vulnerable sections/subsections on observation for behaviors using finite number of objects or threats in a complex set of subsystems.
- The main attacks depending upon the layers are:



Q 7 a) Characteristics requirement of Wireless Sensor Networks

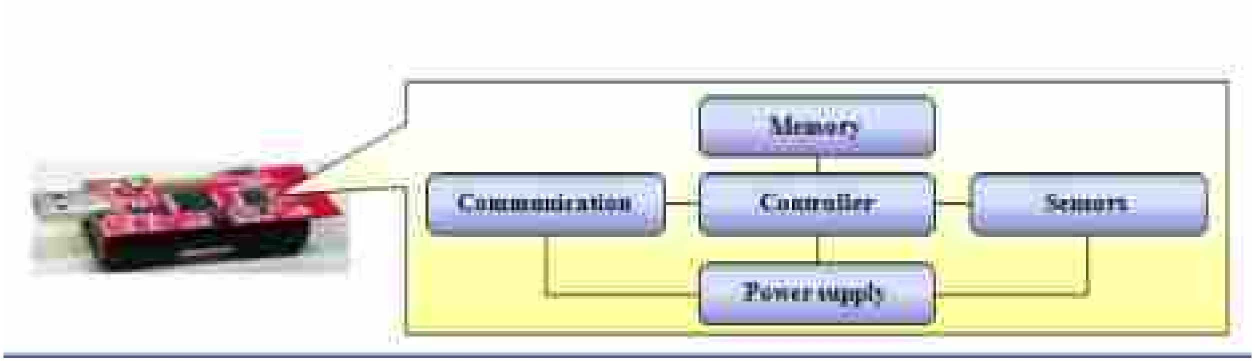
Wireless Sensor Networks mainly consists of sensors. Sensors are -

- Low power (as battery operated)
- Limited resources
- Limited memory
- Low bandwidth
- Energy constrained due to their small size.
- As can also be deployed in extreme environmental conditions therefore are prone to enemy attacks.
- As deployed in an ad hoc manner hence have the property of self organizing and self healing.
- Each node has an unique address.

Q 7 b) Hardware components of Sensor Node

The main architecture of sensor node includes following components:

- Controller module
- Memory module
- Communication module
- Sensing modules
- Power supply module



Controller

- Main functionality
- It is core of a wireless sensor network.
- It collects data from the sensors, processes this data, decides when and where to send it, receives data from other sensor nodes and decides on actuator's behavior.
- It is CPU of sensor node as it executes various programs ranging from time critical signal processing and communication protocols to application protocols.

Communication module

- The communication module of a sensor node is called "Radio Transceiver".
- The essentially tasks of transceiver is to "transmit" and "receive" data between a pair of nodes.
- Depends upon the

  a) Choice of transmission medium

  b) Transceivers

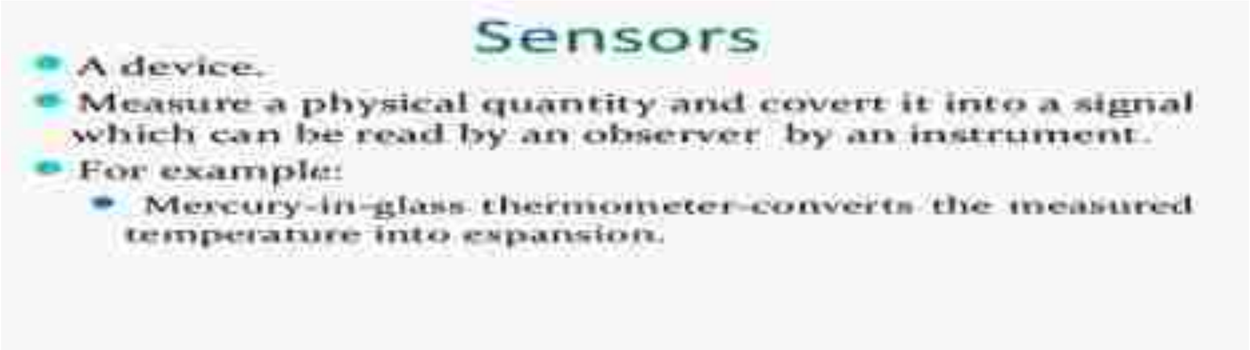
Choice of transmission medium

- Both wired and wireless communication can be used.
- Wired communication:
- It can be carried out by using field buses like LON, CAN etc.

Wireless communication

- It can be radio frequencies, light, ultrasound etc
- It provides relatively high data rate and does not require the

- line of sight between sender and receiver.
- It uses communication frequency between 433 MHz to 2.4 GHz.



Sensors

- A device.
- Measure a physical quantity and covert it into a signal which can be read by an observer by an instrument.
- For example:
  - Mercury-in-glass thermometer-converts the measured temperature into expansion.

Main categories

- Passive
- Passive, narrow-beam
- Active sensors

<u>Passive, narrow-beam sensor</u>

- Omnidirectional
- They are also self powered in the sense that they obtain energy from the environment
- Example: Camera

<u>Active sensors</u>

- It is a sensor that requires external power to operate.
- Examples: the carbon microphone, thermistors, strain gauges, capacitive and inductive sensors, Radar etc.
- Other name: parametric sensors (output is a function of a parameter - like resistance)

Important parameter of sensors: Area of coverage

<u>Area of coverage:</u>

- It defines the distance or the region between sensor and the object to be detected.

<u>Actuator</u>

- A device or mechanism capable of performing a physical action for example motor, light bulb, LEDs etc

Memory:

- Memory is required to store programs and intermediate data; usually, different types of memory are used in WSN for programs and data.
- Random Access Memory (RAM) to store intermediate sensor readings, packets from other nodes, and so on.
- RAM is fast, its main disadvantage is that it loses its content if power supply is interrupted.
- Read-Only Memory (ROM) Program code can be stored in Read-Only Memory (ROM) or in Electrically Erasable Programmable Read-Only Memory (EEPROM) or flash memory.
- Flash memory is similar to EEPROM but data can be erased or written in blocks instead of only a byte at a time. It can also serve as intermediate storage of data in case RAM is insufficient or the power supply of RAM should be shut down.

Power supply module

- It should provides as much energy as possible
- includes following requirements

Two Options

- Through batteries
- Energy scavenging

Energy supply from batteries

- They store energy and provide power.
- A normal battery store about 2.2-2.5 ampere hour at 1.5 V

Traditional batteries

- Primary batteries – not rechargeable
- Secondary batteries – rechargeable, only makes sense in combination with some form of energy harvesting

Requirements include

- Low self-discharge
- Long shelf live
- Capacity under load
- Efficient recharging at low current
- Good relaxation properties (seeming self-recharging)
- Voltage stability (to avoid DC-DC conversion)

Q 8a) Optimization goals and figures of merit:

- The main challenge for a network is how to optimize a network.
- Optimization and figures of merit depend upon certain parameters like:
    - Quality of service
    - Energy efficiency
    - Scalability
    - Robustness

Quality of service involves:

A) Low level networking device observable attributes like: Bandwidth, delay, jitter, packet loss rate

B) High level, user observable also called as subjective attributes like: Quality of voice communication or video transmission.

In WSNs, the high level attribute depends upon the application.

Quality of service: Some generic possibilities are

- Event detection/reporting probability : Means the event that actually happened is detected or not or reported or not to the information sink.
- Event classification error- If events are not only to be detected but also to be classified, the error in classification must be small.
- Event detection delay -It is the delay between detecting an event and reporting it to any/all interested sinks

Energy efficiency:

- The Energy efficiency of the WSN can be increased by considering various aspects.
- Energy per correctly received bit: It defines the average energy consumed in transporting

  and receiving one bit of information, after considering all possible intermediate hops from source to destination.

2) Energy per reported event:

- It defines the average energy consumed in reporting one event. Since same event can be reported from various sources. Hence redundant information can be reduced.

Delay/Energy trade-off:

- In case of reporting of urgent events a huge amount of energy is consumed. Here a trade-off (balance) between Delay/Energy is an important aspect.

4) Network lifetime:

- It is the time for which network is operational.
- Possible definitions are: Time to first node death, Network half-life:, Time to partition:

Time to first node death: The time at which the first node runs out of energy or stop working.

Network half-life: The time at which 50% of the nodes runs out of energy or stop working.

Time to partition: The time at which network get divided into further networks or there is partition between source and sink.

Time to loss coverage: It is the time when the nodes stop observing or monitoring an spot in its deployment range.

Also called as the time at which nodes lost its coverage in deployment range.

Time to failure of first event notification: It is the time when the unreachable part of the network stop reporting any events.

It happens when partition between source and sink has occurred.

Scalability: With WSN potentially consisting of thousands of nodes, the ability to maintain performance characteristics irrespective of the size of the network is referred to as scalability.

- The need for extreme scalability has direct consequences for the protocol design as the complexity will increase and can effect the performance.

Scalability:

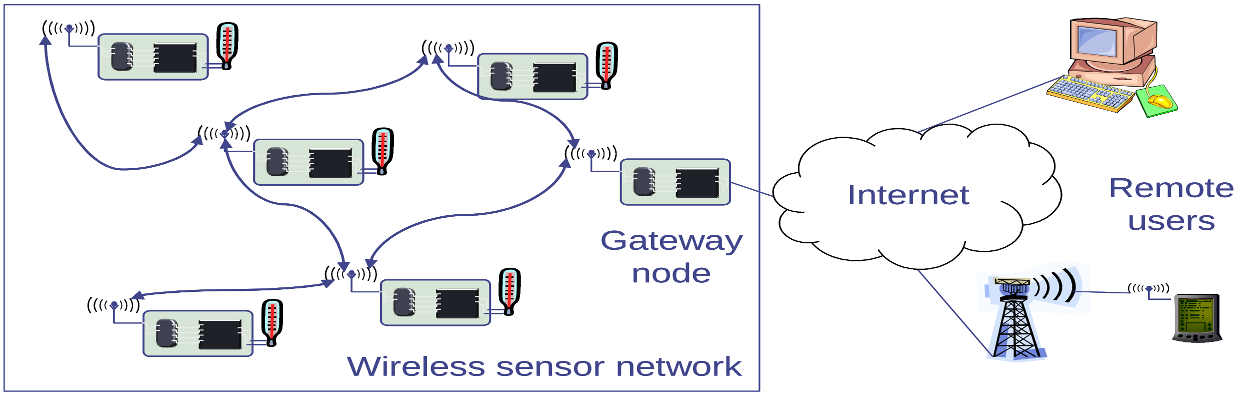
- With WSN potentially consisting of thousands of nodes.
- Architectures and protocols should implement appropriate scalability support rather than trying to be as scalable as possible
- Applications with a few dozen nodes might admit more-efficient solutions than applications with thousands of nodes

Robustness:

- Wireless sensor networks should also exhibit an appropriate robustness
- They should not fail just because a limited number of nodes run out of energy, or because their environment changes and severs existing radio links between two nodes
- If possible, these failures have to be compensated by finding other routes

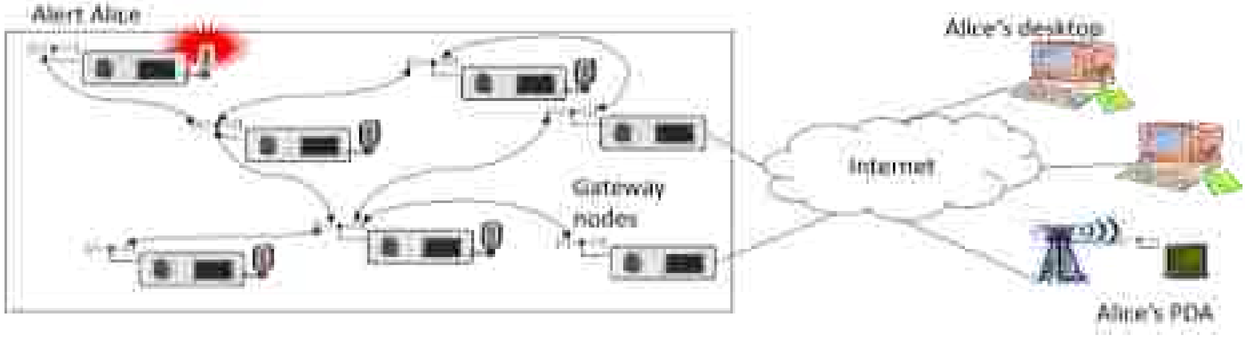
Q 8 b) Gateway concepts for WSN

- Gateways allows the WSN to exchange the data with other devices like mobile phones.
- Gateway node bridges a gap between WSN and other communication devices
- Gateway is equipped with a radio transceiver or some standard wireless communication technique like IEEE 802.11.



- Challenges in WSN to Internet communication
- Let an sensor node ‘ALICE ’wants to deliver an alarm message to some Internet host.
- But here occurs some issues like
- How to handle the several gateways.
- Choose “best” gateway (integrates routing & service discovery)

- Finding the host IP address to which it has to be forwarded.



Q 9 a) Physical layer and transceiver design

- considerations in WSNs
- Some of the crucial factors which influence PHY design in WSN are:
- Consume Less power.
- Small transmit power.
- Most hardware should be switched off or operated in a low-power standby mode most of the time.
- Low data rates
- Low implementation complexity and costs
- Low degree of mobility.
- A small form factor for the overall node.
- Main challenges in Physical layer and transceiver design

In sensor networks, the main challenges are:

- Transceiver architectures that should be simple, low-cost and robust enough to provide the desired service.
- To find modulation schemes

Energy usage profile of transceiver:

- The radiated transmitted power from the transceiver is very typically around 1 mW (around 0 dBm). But the transceiver architectures its self consumes more energy than is actually radiated.

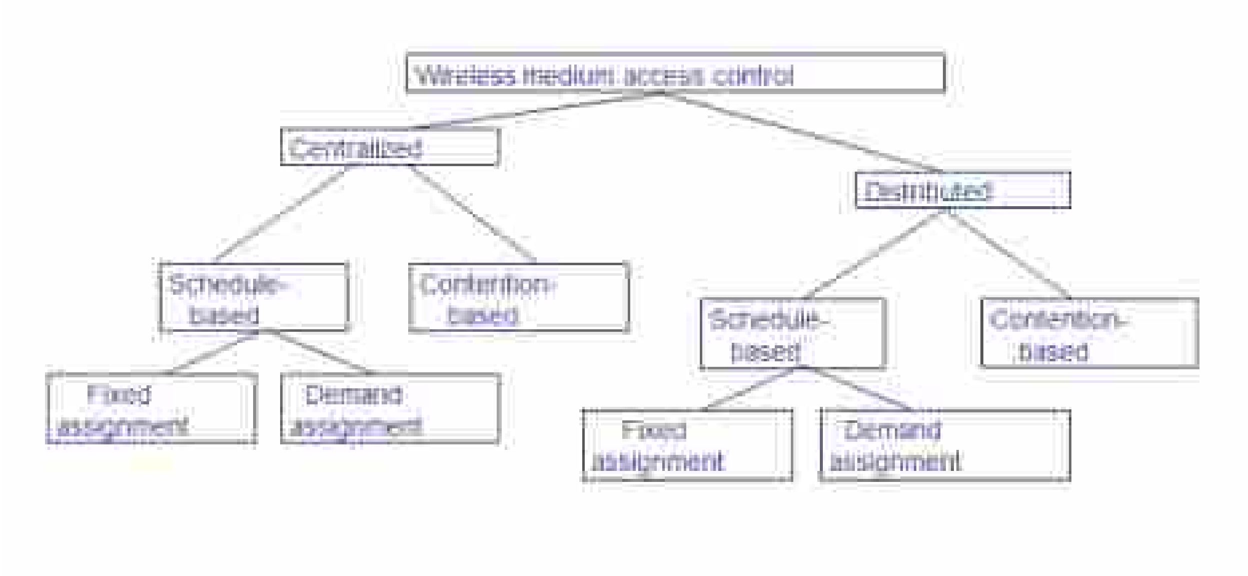
- For example a transceiver working at frequencies beyond 1 GHz consumes 10 to 100 mW of power to radiate 1 mW.
- Similarly for 2.4-GHz CMOS transceivers, to radiate power of 1mW, it consumes 32 mW.
- Whereas the receiver even consume more or less the same power, for example in Mica motes, 21 mW are consumed by transmitter and receiver.
- Many practical transmitter designs have efficiencies below 10 %.
- To reduce average power consumption
- Put the transceiver into sleep state instead of just idling:
- As lot of energy is wasted in simply keeping the transceiver in idle mode all the time. Hence better to change the state from active to sleep state to save energy.
- Limitation: Startup energy/startup time
- A transceiver has to spend upon waking up from sleep mode, During this startup time, no transmission or reception of data is possible.
- For example, the µAMPS-1 transceiver needs a startup time of 466 ms and a power dissipation of 58 mW.
- Choice of modulation scheme
- Power consumption can be decreased by choosing the type of a modulation scheme.
- Modulation scheme effects on many factors like
- Data rate
- Symbol rate
- Implementation complexity
- Target BER

The expected channel characteristics.

- β (Modulation index)
- For small packet sizes, the binary modulation schemes like PSK and FSK are more energy efficient.
- If β is reduced to β = 1, then m-ary modulation scheme will be truly better than binary modulation.

- The main advantage of m-ary modulation scheme is that it decreases the startup time, the transmitted packet lengths also reduces, it reduces the bit error rate.
- Hence a optimal decision should be taken to have a proper balance between the modulation scheme and other parameters as to increase transmission robustness, since these also have energy costs:

Q 9 b) Wireless Sensor Network MAC protocols



- Centralized medium access
- One central station control the nodes and guide when to access the medium. Example: Polling, TDMA schedules.

  Advantage:

  Simple, quite efficient (e.g., no collisions)

  Disadvantage:

- It burdens the central station.
- Not directly feasible for wireless sensor network.
- But it can be useful for network divided into small clusters.
- Schedule- vs. contention-based MACs

  <u>Schedule-based MAC</u>

- A schedule exists, a participant can use the resource at a particular time (TDMA component)

- Schedule can be fixed or computed on demand or mixed

  Advantage: Usually avoid collisions, overhearing, idle listening.

- Needed: time synchronization!

- Contention-based/ unscheduled MACs

- Before sending a message, a sensor listens to the medium. If it's busy, wait a random time then retry again and if it's free then it will send the message

- Any transmitter can access the channel at any time.

- Contention-based/ unscheduled MACs

  Advantages:

- It can adapt for changes like change in node density, traffic load etc.

- The sensors don't have to be synchronized together.

  Disadvantages:

- Consumes more power than scheduled MAC protocols since all sensors have listen to the channel all the time.

- Risk of collision is there.

Medium Access Control
Unscheduled MAC
Scheduled MAC
Contention based

Q 10 a) CSMA

- CSMA protocols are contention-based, where neighbors try their luck to transmit their packet.

- The node sense the channel before transmitting.

- If the channel is busy then the node selects other random channel, repeats the carrier sensing and after a number of unsuccessful trials it just backsoff.
- And if the channel is idle then it start transmitting.

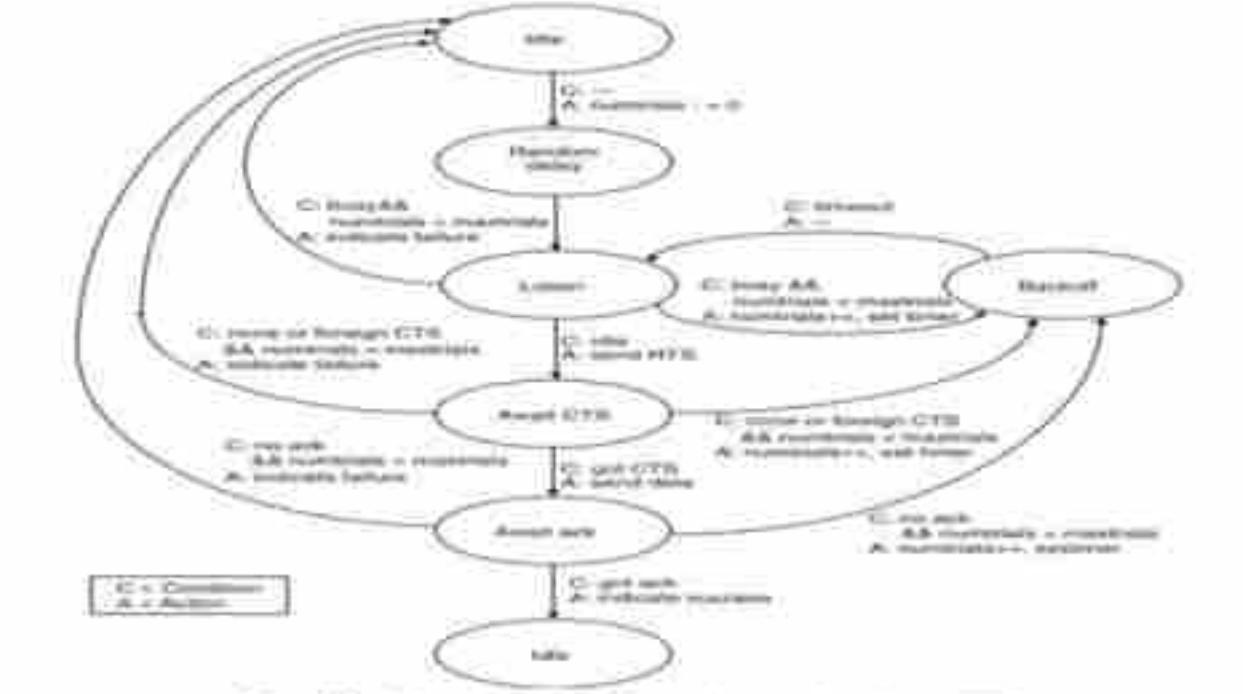

Idle
C: —
A: numtrials := 0
Random delay
C: busy && numtrials = maxtrials
A: indicate failure
C: timeout
A: —
Listen
C: busy && numtrials < maxtrials
A: numtrials++, set timer
Backoff
C: idle
A: send RTS
C: none or foreign CTS && numtrials = maxtrials
A: indicate failure
Await CTS
C: none or foreign CTS && numtrials < maxtrials
A: numtrials++, set timer
C: no ack && numtrials = maxtrials
A: indicate failure
C: got CTS
A: send data
Await ack
C: no ack && numtrials < maxtrials
A: numtrials++, set timer
C = Condition
A = Action
C: got ack
A: indicate success
Idle

Figure 5.8 Schematic of the CSMA protocol presented in reference [illegible]

Working: (Finite state automation)

- Step1: ["Idle state"] :

  Normally the nodes are in "idle state".

- Step 2: ["Random delay"]

  When it receives the packet from upper layer for transmission to lower layer (called as downstream node), it restarts the "Random Delay".

- Step3: ["Listen"]

  The nodes perform carrier listening for some time. If the medium is found busy, it goes to "Backoff" mode.

- Step4: ["Backoff"]

Here nodes wait for a random amount of time for the channel to be free and then goes to sleep state. “Backoff” period is used by application layer for the “phase change” i.e. to desynchronize the

- Step5: [“Await CTS state”]

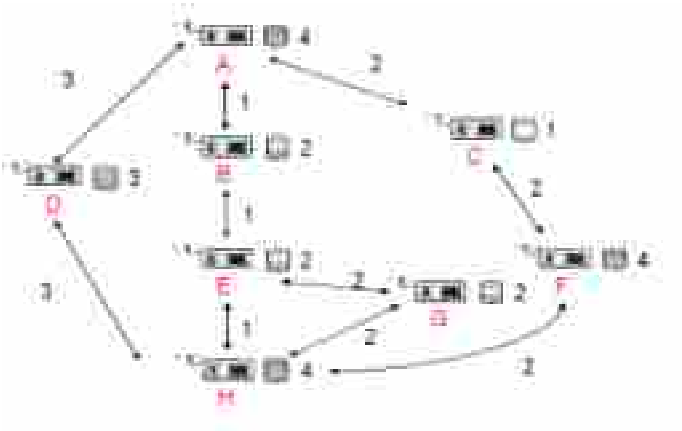
  Here the node waits for CTS packet. If CTS packet arrives in time then node sends its data packet and waits for ACK

- Step6: [“Await ack state”]

  It can be explicit ack or parent node piggybacks the ack on packet and then forwarded to grandparent.

Q 10 b) Various aspects of Energy-Efficient unicast

- Minimize energy per packet (or per bit)
- Maximize network lifetime
- Routing considering available battery energy
- Maximum total available battery capacity
- Minimum battery cost routing(MBCR)
- Min-Max Battery Cost Routing(MMBCR)
- Conditional Max-Min Battery Capacity Routing(CMMBCR)
- Minimize variance in power levels.
- Minimize energy per packet (or per bit)
- Calculate the total energy required to transport a packet over multihop from source to destination and then minimize the total amount of energy by selecting a good route.
- The minimum energy/bit is obtained by considering the path from: A-B-E-H =4.
- Reducing the hop count increases the energy consumption.



Maximize network lifetime

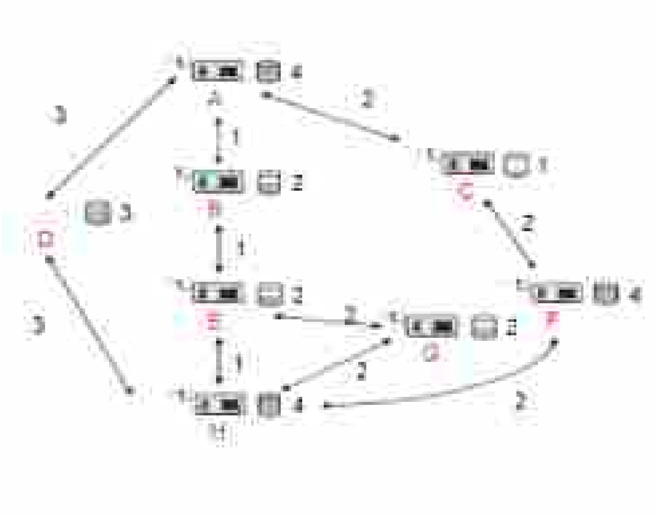
- For energy efficient transmission the network should be able to fulfill its duty as long as possible.
- There are many events that determine the network lifetime like: Time until first node fails, loss of coverage, partitioning

Routing considering available battery energy

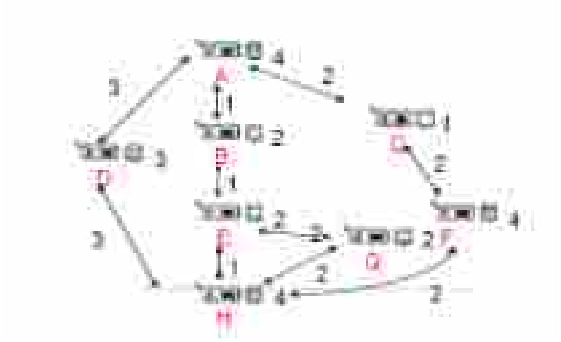
- Batteries are the finite energy supply in nodes and are limiting factor in network lifetime. Hence information about the battery status is required in making routing decisions.

Some possibilities are:

- Maximum Total Available battery Capacity
- Minimum battery cost routing (MBCR)
- Min–Max Battery Cost Routing(MMBCR)
- Maximum Total Available battery Capacity
- Path metric: Sum of battery levels
- To determine the maximum available battery capacity, choose the route where sum of the available battery is maximized.



- Min–Max Battery Cost Routing
- Instead of using the sum of reciprocal battery levels, simply the largest reciprocal level of all nodes along a path is used as the cost for this path
- Example: A-D-H (1/3)



- Conditional max-min battery capacity routing
- If there are routes along which all nodes have a battery level exceeding a given threshold
- Then select the route that requires the lowest energy per bit.
- If there is no such route, then pick that route which maximizes the minimum battery level