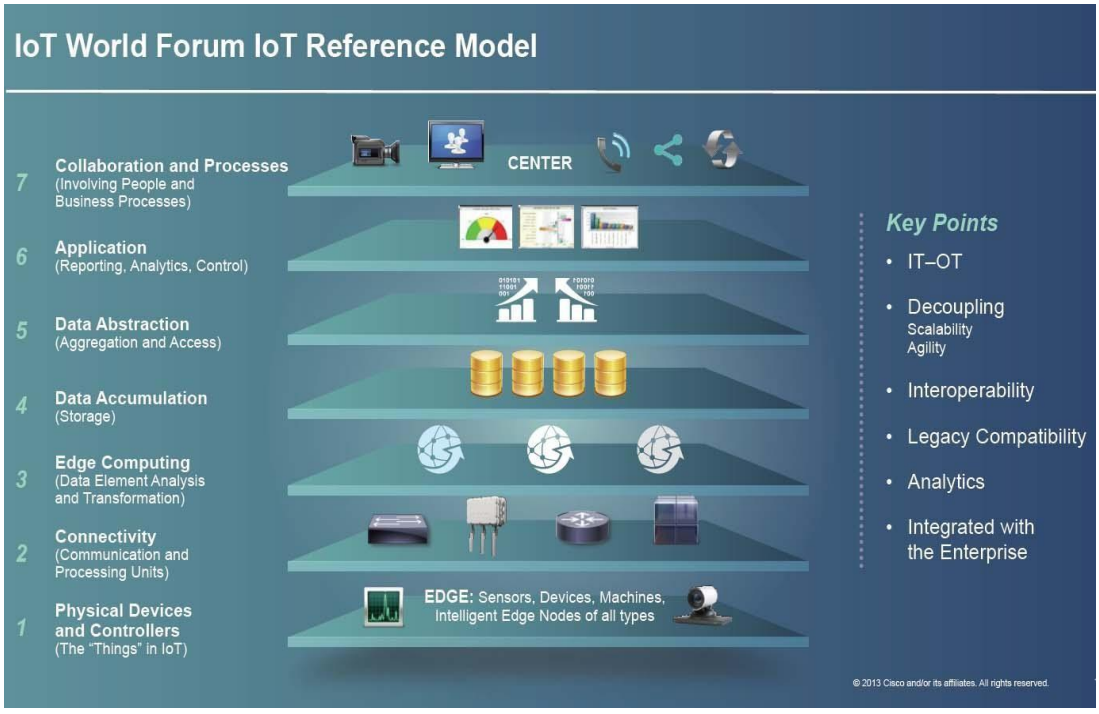
Sub:	IoT&WSN					Sub Code: 18EC741
Date:	March 2022	Duration:	90 Minutes	Max Marks:	50	Sem / Sec: A,B,C,D

Answer any FIVE FULL Questions

1 a. What is IoT? Explain Conceptual framework with necessary equations and reference model by CISCO [8]
b. What are three architectural functionalities of M2M framework. Compare IoT and M2m.[8]
c. Explain CoAP for IoT/M2M [4]

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

b. By CISCO



- A reference architecture of IoT
- The IOT reference model has 7 levels called "LAYERS" OR "TIERS".
- Each level is defined with some terminology.
- Each level perform some specific function.
- The model describes how the task at each layer should be handled to maintain simplicity and scalability.
- IN IOT the data flows in both directions i.e.
  From top to bottom (LAYER 7 to LAYER 1) – control pattern. From bottom to top (LAYER 1 to LAYER 7) – monitoring pattern
  But Basically follows top-down approach (means consider top layer design first and then move to the lowest).

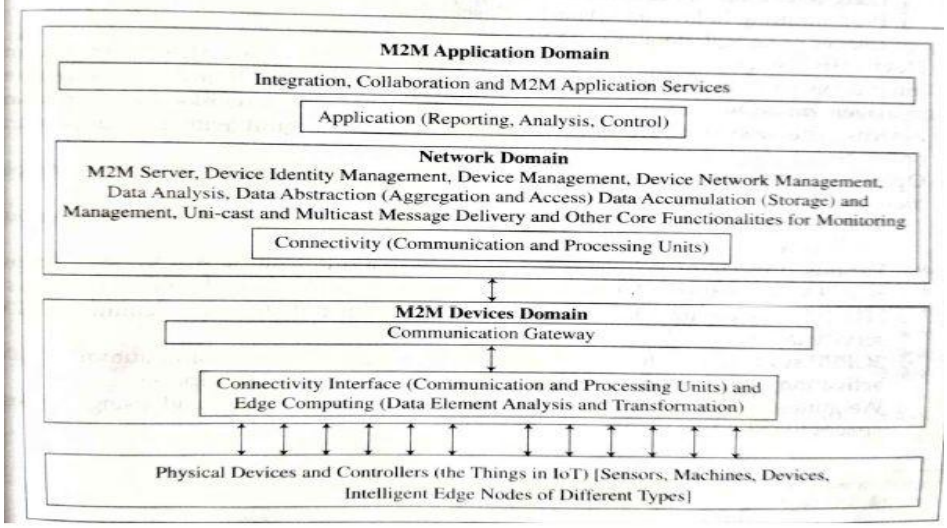
- It defines basic architectural building blocks and their integration capability into multi-tiered systems.
- The reference model defining relation-ships among various IoT verticals, for example, transportation and healthcare
- Gives a blueprint for data abstraction
- Recommends quality 'quadruple' trust
- "Protection, Security, Privacy, and Safety"
- Defines no new architecture and no reinvent but existing architectures congruent with it

----**Three architectural functionalities of M2M framework**

M2M refers to the process of communication of the physical devices or machines or smart devices with the other machines of same type without intervention of humans.
□Smart devices collect the data, monitor it, do some necessary computations, and perform communication to the remote devices using internet
□It also uses servers or cloud end applications, services and processes.
□M2M is used in many applications like home automation, industrial automation, smart cities, healthcare etc .
□For-example using robots interacting with machines at home.
□IT is similar to SCADA (Supervisory control and data acquisition system)

M2M architecture consists of three domains (Figure 1.9):

1. M2M device domain
2. M2M network domain
3. M2M application domain



IoT and M2M Comparison

M2M	IOT
Abbreviation for machine to machine	Abbreviation for internet of things
It is about direct machine to machine communication	It is about sensor automation and internet platform
It support point to point communication	It support cloud based communication
Device not necessary relay on internet	Device necessary relay on internet
It mostly based on hardware	It based on both hardware and software
Machine normally communicates with single machine at a time	Many user can access at a time over internet
It uses either proprietary or non IP based protocols	It uses IP based protocols
Its for only B2B business type	Its for B2B and B2C business type
Limited number of devices can be connected at a time	More number of devices can be connected at a time
It does not support open API's	It supports open API's
It is less scalable	It is more scalable

CoAP for IoT/M2M: (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
- COAP is used for CORE using ROLL network.
- Since COAP is a network oriented protocol therefore has similar features as http like coap://…….. http://…….
- http is based on TCP for PTP communication.
- CoAP is based on UDP for transmission over constrained network.

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 works with the UDP which is a unreliable protocol.
DTLS is application layer protocol which binds with UDP and provide

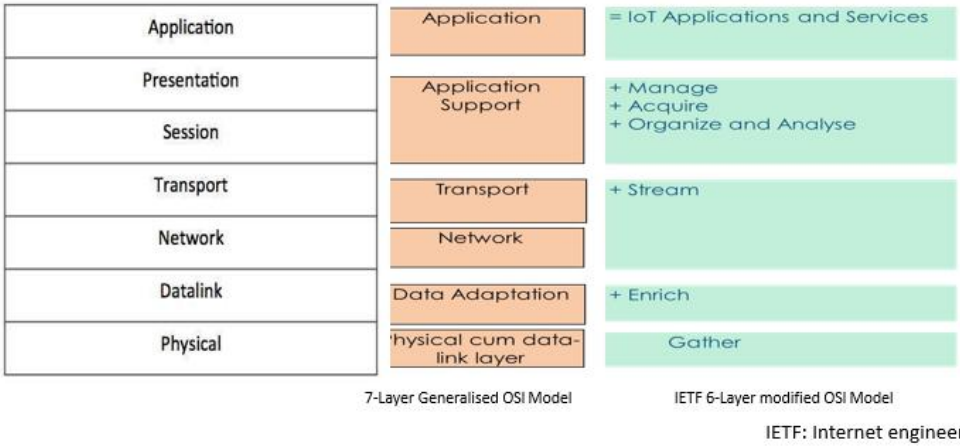
1) Security, integrity, authentication, confidentiality.
2) Packet retransmission
3) Assigning sequence number with handshake
4) Replay detection.

2 A Explain modified OSI model for the IoT/M2M with appropriate figures. [8]

B Explain MQTT protocol [8]

C Four layer architectural framework for smart city developed by CISCO [4]

Modified OSI model for the IOT/M2M Systems



IETF: Internet engineering task force

Physical 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.

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.

Gateway layer:

The main functions of this layer are:

1) Routing the data coming from sensors to the next upper management layer
2) This layer is responsible for managing the traffic between networks that uses the different protocols.
3) This layer is responsible for protocol translation and other interoperability tasks
4) The IOT gateway device is employed because some of the devices can't directly communicate to the front end applications as there is no network stack present for internet connectivity, hence gateways acts as proxy.

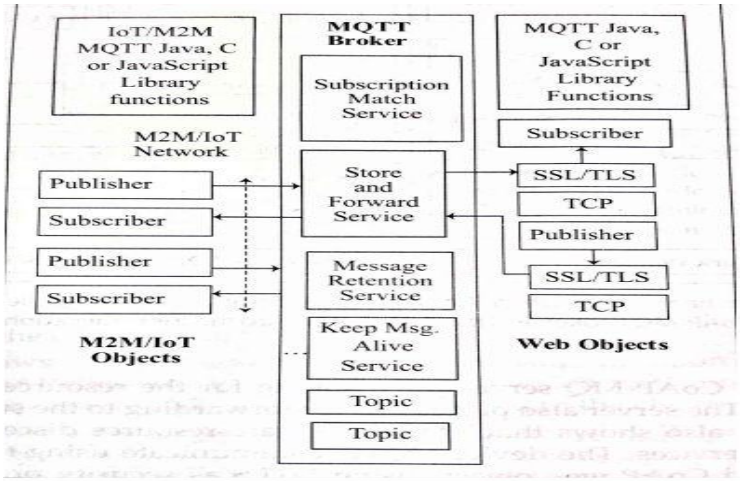
Application support layer/middleware layer:

□ 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.

MQTT: Message Queuing Telemetry Transport

- An **open source protocol for machine-to-machine (M2M)**/"Internet of Things" connectivity
- **(Telemetry dictionary meaning is measuring and sending values or messages to far off places by radio or other mechanism)**
- Created by IBM IN 1999, as a constrained environment protocol.
- Designed to **provide connectivity** (mostly embedded) **between applications** and middle-wares **(M2M/IOT objects)** on one side **and networks and communications (WEB Objects)** on the other side.



Publisher:

- They are the **light weight sensors also called as clients**
- These **clients first make connections to the Broker and then publish a message to the broker.**
- **The message include the topic**. The topic is the routing information for the broker.

Broker:

- Perform **store and forward** operation
- **Receives the topics** from **publishers**
- **Each client** that wants to receive messages **first subscribes to a certain topic** and then **the broker delivers all messages with the matching topic to the client**.

Therefore the clients don't have to know each other. They only communicate over the topic.

Subscribers:

- They are the clients that require the information from publishers

Four layer architectural framework for smart city developed by CISCO

Layer 1 consists of sensors, sensor networks and device networks in parking spaces, hospitals, streets, vehicles, banks, water supply, roads, bridges and rail roads. Bluetooth, Zigbee, NFC, WiFi and other short range communications protocols are used at the bottom layer.

Layer 2 captures data at distributed computing points where data is processed, stored and analyzed.

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

Layer 4 consists of new innovative applications, such as waste containers monitoring, WSNs for power loss monitoring, bike sharing management, smart parking, smart surveillance and much more.

Four Layer architecture framework developed by CISCO

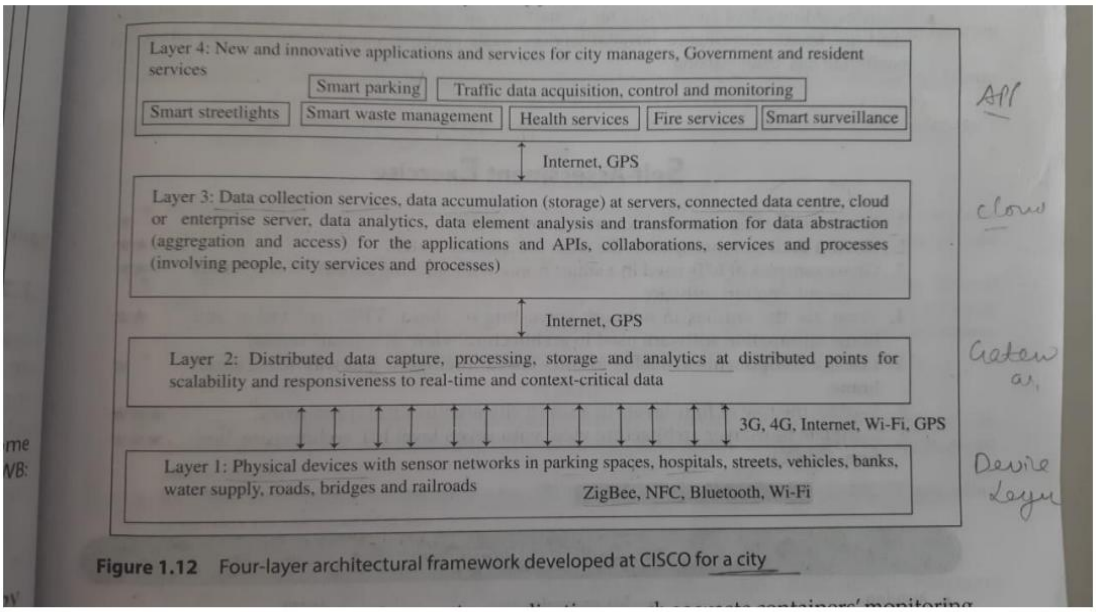


Figure 1.12 Four-layer architectural framework developed at CISCO for a city

3

a. Explain Cloud service model and development model with examples

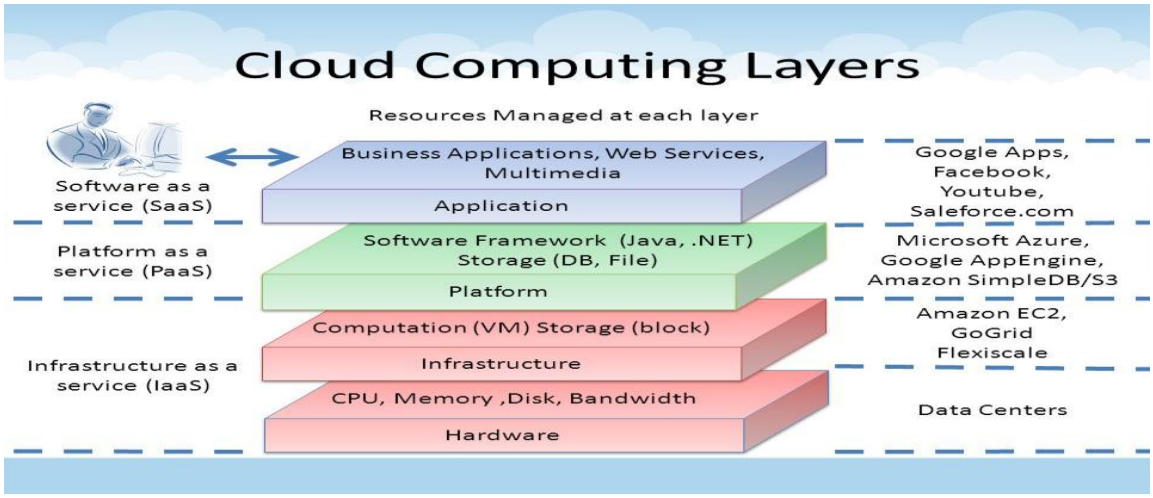
b. Explain IPV4 and addressing IPV4

c. Explain HTTPs Protocol

a. 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.
--It's called cloud computing because it does not require a user to be in a specific location to gain access to it. This type of system allows the users to store files and applications on remote servers, and then access It via the internet any time.

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)**



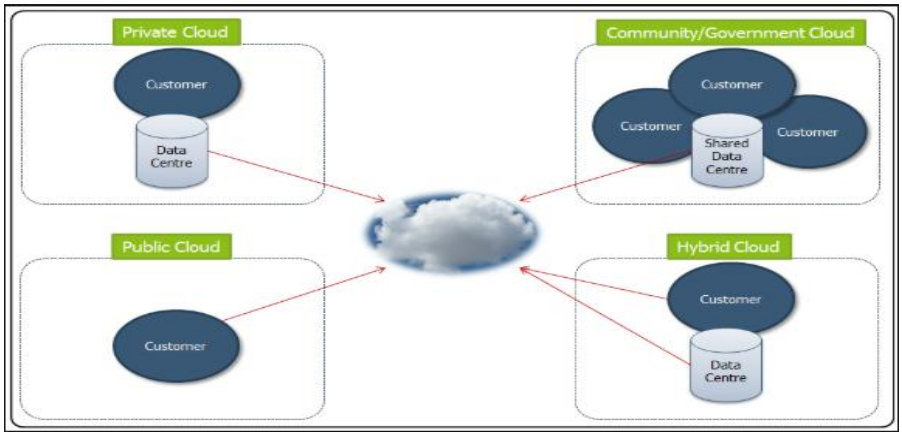
Deployment model

PUBLIC CLOUD :
The Public Cloud allows **systems and services to be easily accessible to the general public.** Public cloud may be less secure because of its openness, e.g., e-mail. Amazon Elastic Compute **Cloud** (EC2), IBM's Blue **Cloud**, Sun **Cloud**, Google AppEngine and Windows Azure Services Platform

PRIVATE CLOUD :
The Private Cloud **allows systems and services to be accessible within an organization**. It offers increased security because of its private nature for example Oracle, IBM, Redhat

COMMUNIY CLOUD :
The **Community Cloud** allows **systems and services to be accessible by group of organizations.**

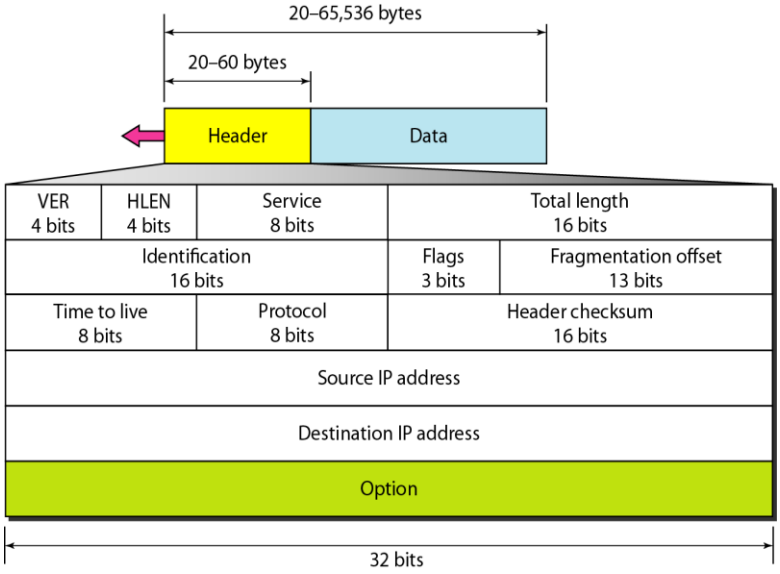
HYBRID CLOUD :
The **Hybrid Cloud** is **mixture of public and private cloud.**



IPV4 and Addressing

- ➔ 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**

Datagram format



IPv4 address:

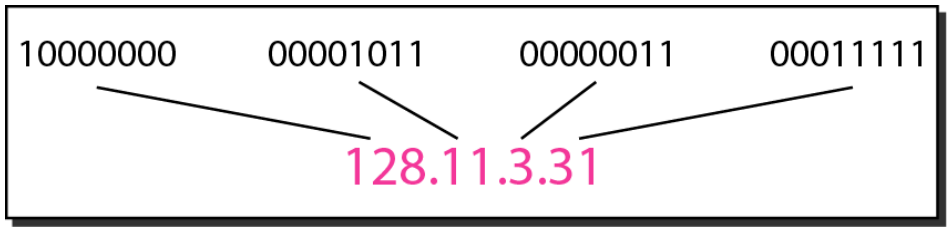
- *Binary notation for an IPv4 address:*
- *Dotted-decimal notation:*

- *Binary notation for an IPv4 address:*

The IPv4 address is written in 32 bits or 4 bytes.

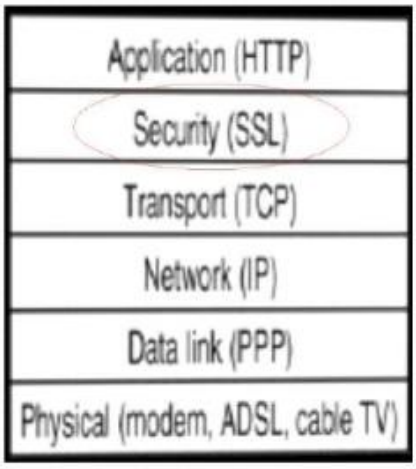
- *Dotted-decimal notation:*

The internet address is written in decimal form with decimal dots seperating the bytes



HTTPS

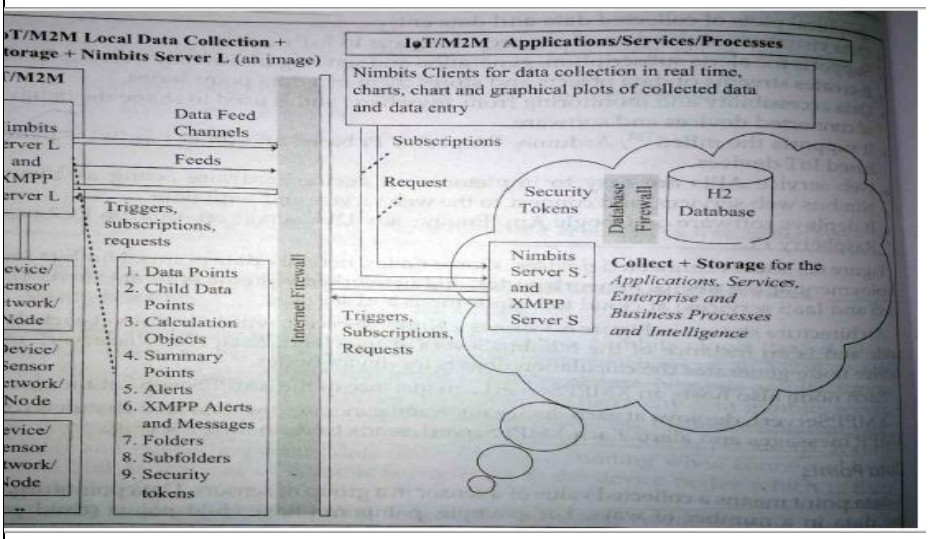
- HTTPS stands for Hypertext Transfer Protocol over Secure Socket Layer, or HTTP over SSL.
- SSL acts like a sub layer under regular HTTP application layering.
- HTTPS encrypts an HTTP message prior to transmission and decrypts a message upon arrival.



4. a. Explain IoT services using Nimbits server. [6]
b. Explain 6LoWPAN .[6]
c. Cloud service model -Repeated

Nimbits server

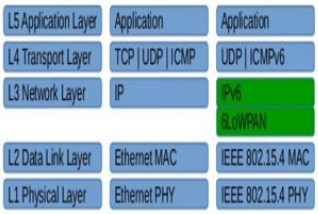
- Nimbits is a platform as a service (PaaS) used to develop software and hardware solutions that seamlessly connect to the cloud and each other.
- Nimbits server runs on powerful cloud platforms like Google App Engine to the smallest Raspberry Pi device.
- Nimbits server is a web portal and API designed to
- Provides time-stamping or geo-stamping on incoming data.
- Store and process that time and location stamped data over cloud (pushing the data over cloud and store them in a data point)
- Provide filtering to incoming data from noise, add important changes to it and then generate trigger events and alerts based on rules and then sending them in real time over internet.
- It provides rule engine for connecting sensors, persons and software to cloud.

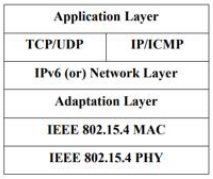


- The Nimbits serverL is deployed at each device node.
- The Nimbits servers first store, then filter and clean the data from noise, add some important changes, provides time-stamping or geo-stamping to it and send events, alerts (like email alerts or push notifications) by using rules an calculations called as **data feeds channels** (A data feed contains latest updates of current information like events, alerts etc) using XMPP messages.
- These data feeds (notifications) are sent over data feed channel using XMPP (Extensible Messaging and Presence Protocol) messages
- This pushing of the **alerts and messages down quickly or repeatedly is called as "Jabbing".**
- Hence server relay that data up to a website or to a some mobile device

6LoWPAN

- **6LoWPAN** is an acronym of *IPv6 over Low -Power Wireless Personal Area Networks (WPAN).*
- 6LoWPAN concept originated from the idea that *"even the smallest and the low-power devices (operating in LR-WPAN) with limited processing capabilities can participate in the Internet of Things*
- 6LoWPAN sits between network layer and data link layer hence called as **adaptation-layer protocol** for the IEEE 802.15.4 network devices.
- Features of 6LoWPAN are **header compression, fragmentation and reassembly**.
- The devices are the **WPAN** nodes having low power and low speed and forms a mesh network.





Protocol Stack of 6LoWPAN Architecture

5

a. Explain security in IOT and security models [8]
b. Explain IoT Security Tomography and layered attack model [8]
c. Write note on Aurdino Programming [4]

a. 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:

1) Identity management
2) Authentication
3) Authorization
4) Key exchange and management
5) Trust and reputation

Identity Management:

- 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.
- Devices must establish their identity before they can access gateways and upstream services and apps.

Secure Authentication/Authorization

- IoT devices should be authenticated with strong usernames IDs and password before being allowed tocommunicate with other IoT devices on the network.
- Authentication token/session should always be unique to each user along with user id, app id and deviceid.
- System's credentials, application, device and server should be authenticated as spoofed devices couldtransmit malicious data to other devices and can implement a denial- of-service attack on the IoT network
- Public key cryptography also known as asymmetric encryption keeps the data safe and secure duringtransmissions.
- Certification Authority (CA): This is effectively done by a issuing a digital certificate to confirm theauthenticity of the device, firmware / software updates, and facilitate encrypted communications

Key exchange and management

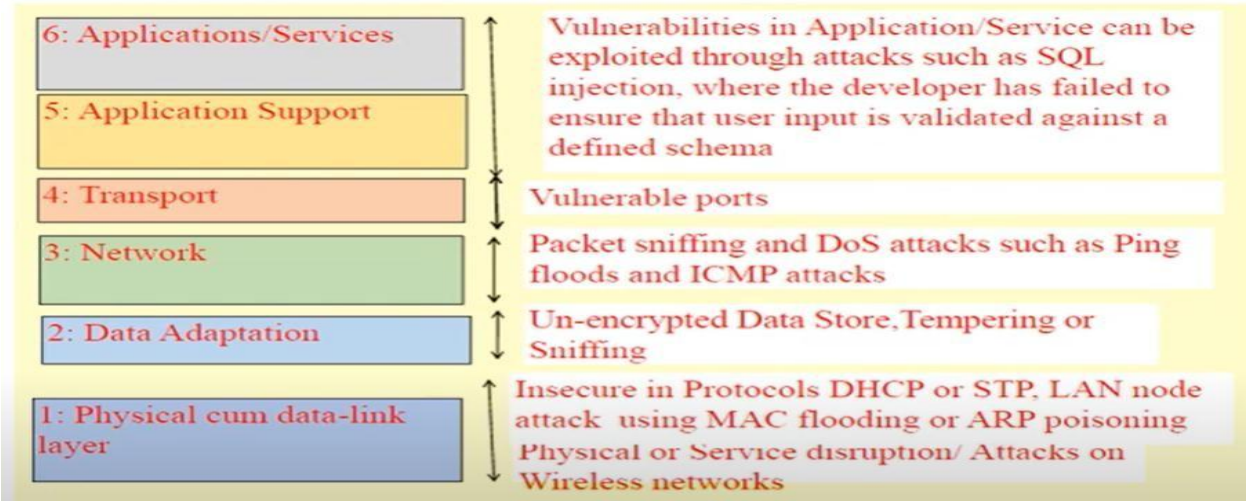
Key exchange (also key establishment) is a method in cryptography by which cryptographic keys are exchanged between two parties, allowing use of a cryptographic algorithm**.**

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.
The Trusted IoT Master:
A trusted master must provide a secure communication with dependent sensor devices, and issue firmware/software updates to those devices and ensures that the code is authentic, unmodified and non-malicious

b. Security tomography
It enables finding the attack vulnerable sections/subsections on observation for behaviors using finitenumber of objects or threats in a complex set of subsystems

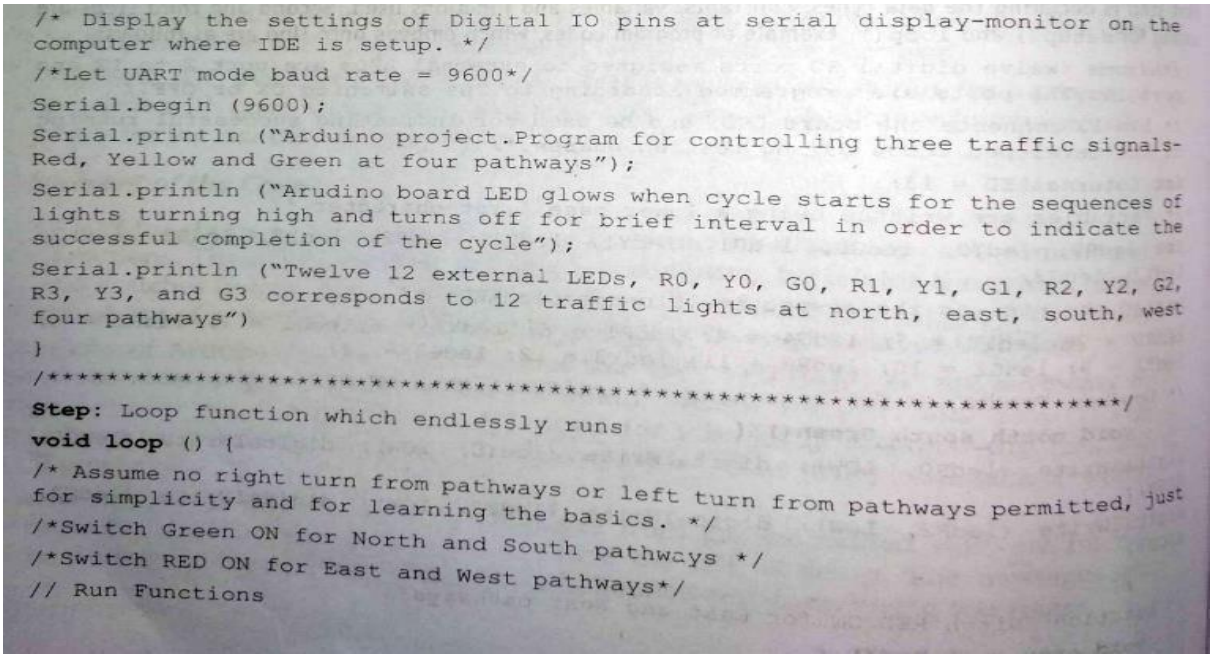
LAYERED ATTACKER MODEL:



Ardunio Programming:

```
testing phases.*/                                    successful running
int internalLED = 13;
/* Variables are written using a lower case first character */
int ledR0, ledY0, ledG0, ledR1, ledY1, ledG1, ledR2, ledY2, ledG2, ledR3,
ledY3, ledG3;
Assign the pins to the respectively connected LEDs */
ledR0 = 2; ledY0 = 3; ledG4 = 4; ledR1 = 5; ledY1 = 6;ledG1 = 7; ledR2 = 8;
ledY2 = 9; ledG2 = 10; ledR3 = 11; ledY3 = 12; ledG3 = 14;
/* Declare Functions for sequences of traffic lights ON-OFF as follows: */
   void north_south_Green() {
digitalWrite (ledR0, LOW); digitalWrite (ledY0, LOW); digitalWrite (ledG0,
HIGH);
digitalWrite (ledR2, LOW); digitalWrite (ledY2, LOW); digitalWrite (ledG2,
HIGH);
   };
/* Function Switch RED ON for East and West pathways*/
   void east_west_Red() {
```

```
digitalWrite (ledR1, HIGH); digitalWrite (ledY1, LOW);digitalWrite (ledG1,
LOW);
digitalWrite (ledR3, HIGH); digitalWrite (ledY3, LOW); digitalWrite (ledG3,
LOW);
      };
/*************************************************************/
void setup ( ) {
/* GPIO pins 2 to 12 and 14 are thus assigned port numbers corresponding to 12
external LEDs, R0, Y0, G0, R1, Y1, G1, R2, Y2, G2, R3, Y3, and G3.*/
/* Assign mode of each pin as output */
pinmode (ledR0, OUTPUT); // Constants are written in Upper Cases
pinmode (ledY0, OUTPUT);
.
pinmode (ledY3, OUTPUT);
pinmode (ledG3, OUTPUT);
```

```
/* Display the settings of Digital IO pins at serial display-monitor on the
computer where IDE is setup. */
/*Let UART mode baud rate = 9600*/
Serial.begin (9600);
Serial.println ("Arduino project.Program for controlling three traffic signals-
Red, Yellow and Green at four pathways");
Serial.println ("Arudino board LED glows when cycle starts for the sequences of
lights turning high and turns off for brief interval in order to indicate the
successful completion of the cycle");
Serial.println ("Twelve 12 external LEDs, R0, Y0, G0, R1, Y1, G1, R2, Y2, G2,
R3, Y3, and G3 corresponds to 12 traffic lights at north, east, south, west
four pathways")
}
/*************************************************************************/
Step: Loop function which endlessly runs
void loop () {
/* Assume no right turn from pathways or left turn from pathways permitted, just
for simplicity and for learning the basics. */
/*Switch Green ON for North and South pathways */
/*Switch RED ON for East and West pathways*/
// Run Functions
```

6 Explain how the data is read from sensors and devices [10]

1) Sensors senses the analog data and send the analog data to 10 bit ADC (Analog to digitalconverter).
2) ADC send the 10 bit parallel data to PISO converter.
3) Parallel IN serial out (PISO) converter converts parallel data to serial data.
4) This serial output connects to SPI (Serial Peripheral Interface) Pin of Arduino board.
5) For example let sensor senses the temperature of 100 degree.
6) ADC converts 100 degree to binary 1111111111,which is send to PISO then to SPI of arduino

7 Explain PAMAS Protocol with necessary figures.

PAMAS: Power Aware Multi-Access with Signaling

a) Signaling channel/control channel: For sending RTS/CTS messages and busy tone messages and itenables the nodes to determine that for how long they can power off themselves.

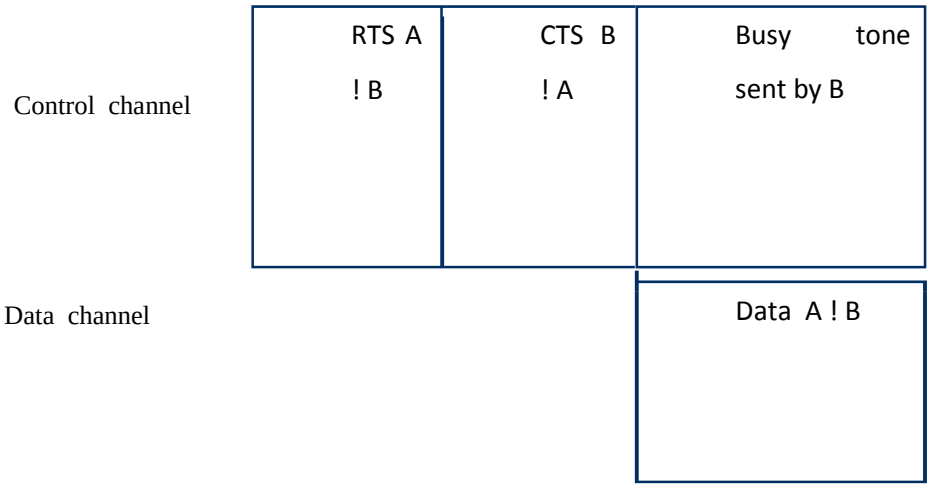
b) Data channel: Actual data transmission happens.

Procedure

a) Node A does not sense channel and transmits RTS on control channel. If node B receives RTS, it sendsCTS on control channel if it can receive the data and does not know about ongoing transmissions

b) B then sends busy tone as it starts to receive data.

c) If node A fails to receive the CTS signal then it enters backoff scheme using binary exponential method.



PAMAS conserves the battery power by selectively powering off the nodes that are neither receiving nortransmitting and during overhearing.

Conditions that force nodes to turn off the power:

Case1) The node has no packet to transmit,

Case2) The node has packet to transmit

Case1) The node has no packet to transmit:

- Here the node should go to sleeping mode, turn off the power and wake up exactly when outgoing transmission ends so that it can receive the packet.
- *Use a probing protocol on control channel*

❑ Probing protocol defines the length of outgoing transmission (i.e. length of ongoing packet).

❑ *Use a probing protocol on control channel*

❑ Probing protocol defines the length of outgoing transmission (i.e. length of ongoing packet).

❑ Here the node sends a *t_probe (l/2,l)* packet where *'l' is the length of packet.*

- ❑ Any transmitter node who finishes the transmission with in this time interval ***'(l/2,l)'*** answers with ***t_probe_response (t)*** packet indicating the time ***'t'*** where transmission ends.
- ❑ ***If the node manages to receives this response packet it knows where exactly this transmission ends and can wake up fast***

Case2) The node wakes up during ongoing transmission and has packet to transmit.

- ❑ Here the node has to take care of ongoing transmission as well as ongoing reception.
- ❑ **It uses a probing protocol to define the time for next wakeup.**
- ❑ **Probing protocol that defines the length of ongoing packets**

6 a. Explain about the security and threat analysis in IoT/M2M using neat figure. **(08 Marks)**

b. Explain layered attacker model with possible attacks and suggest the steps for mitigating attacks. **(08 Marks)**

c. Explain how data is read from sensors and devices. **(04 Marks)**

a)

Ans: * All the systems require secure communication to the cloud to ensure personal data cannot be accessed or modified.

* The security Model for IOT consists of communication security that focuses mostly on the confidentiality and integrity protection of interacting entities and functional components.
* IoT reference architecture is a guide for one or more concrete architects.
* IoT reference architecture is a set of 3 architectural views:

1. Functional View: The functional view is from F.Carrez and Co-workers, which gives the description of what the system does, and its main functions.
2. Information View: It gives the description of the data and information that the system handles.
3. Deployment and Operational View: It gives the description of the components of the system such as devices, network routers, servers, etc.

Security is one of the functional group of the functional view.

FG consists of security functions between the application and device.

Security FG contains five sets of functional

a)

components which are required for ensuring security and privacy.

* The 5 functional components :

1) Identity management (IdM): It must be able to identify devices, sensors, monitors, and manage their access to sensitive and non-sensitive data.

2) Authentication: Devices must be secured with the strong passwords for the authentication. Third party software tools can be used that make devices more secure.

3) Authorisation: It is the process of granting permissions to an authenticated identity.

4) Key exchange and management: The main challenge of it is to exchange information so that no one else but only the authorized persons can obtain a copy of it.

5) Trust and reputation: Trust is the subjective probability by which an individual, A expects that another individual, B performs a given action on which its welfare depends. Reputation of an entity is an expectation of its behavior based on other entities. Observation or the objective information about the entity's past behaviour within a specific context at a given time.

→ A threat analysis tool first generates the threats & analyses the system for threats.

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

→ Stride means a regular or steady course, pace or striding means, passing over or across in one long step.

→ Stride means taking a long step for dainty little steps.

b) **Layered attacker model**

Layered attacker model gives possible attacks on the layers.

Data between the things and application/service communicate through six layers and their sub layers. The layered attacker model shows the vulnerabilities at each layer, so that necessary solutions can be given for each layer attack.

For example, using a temper resistant router, packet filtering and controlling routing messages and packets data between layers 3 and 4 through a Firewall reduce the risks

Layer 1 Attacks Solution

1. Depends on the devices used
2. For example, link level provisioning of security
3. Uses—BT LE link level AES-CCM 128 authenticated encryption algorithm for confidentiality and authentication, and
4. ZigBee at link level security using AES-CCM-128.

Layer 2 Attacks Solution

1. Programming the network switches to prevent internal node attacks during use of DHCP or Spanning Tree Protocol (STP)
2. Additional controls:
1. ARP inspection,
2. Disabling unused ports and

Enforcing effective security on VLAN's (Virtual LAN) to prevent VLAN hopping.

4. Provisions for MAS for security, root key data store, and devices and data authentication in LWM2M OMA specification for device gateway to Internet

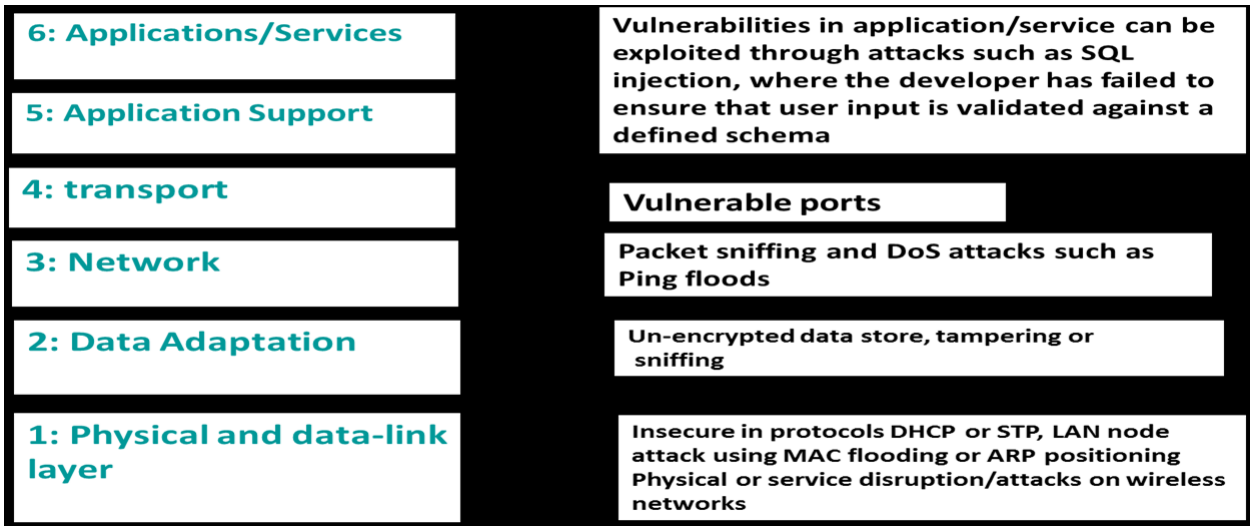


Figure: Layered attacker model and possible attacks using IETF six-layer modified model for IoT/M2M

Layer 3 Attacks Solution

1. Use of temper resistant router
2. Use of packet filtering
3. A firewall for controlling routing messages and packets data between layers 3 and 4 for reducing the risks.

Layer 4 Attacks Solution

1. Port scanning method to Identify the vulnerable port
2. Effective firewall configuring and opening of network ports and locking down ports only to those required
3. DTLS between layers 5 and 4
4. The DTLS three types of security services: integrity, authentication and confidentiality.
5. Inclusion of SASL (Simple Authentication and Security Layer) for security when using the XMPP protocol.

Layer 5 and 6 Attacks Solution

1. Results of poor coding practices of Application programmer
2. Un-encrypted Data Store, Tempering or Sniffing
3. Use HTTPS communication link for Web applications/s

6c. Describe how data is read from sensors and devices.

Internet of Things (IoT) is an ecosystem of connected physical objects that are accessible through the internet. The 'thing' in IoT could be a person with a heart monitor or an automobile with built-in-sensors, i.e. objects that have been assigned an IP address and have the ability to collect and transfer data over a network without manual assistance or intervention. The devices are connected to a server via internet. The server hosts the application logic that collects, analyses, stores and deletes data that is sent by the devices. The controller usually collects the data and periodically uploads to a server where there is a database that organizes the data based on value. This data is further used for analytics. Some example of controller used to read the data are like: Raspberry Pi and Arduino.

Raspberry pi is a credit-card-size single board 32 bit computer. The original purpose of Raspberry Pi was to promote teaching basic computer science concepts in schools, however because of its low cost and multi functions, it has brought many inventive approaches to computing. Raspberry Pi comes with a New Out of Box System (NOOBS) system and an application store (The Pi Store) website for users to exchange apps.

Arduino board is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It is a single-board 8 bit microcontroller, intended to make the application of interactive objects or environments more accessible. It can be purchased pre-assembled or as do-it-yourself kits. Arduino programs are written in C or C++. The Arduino integrated development environment (IDE) is a cross-platform application that comes with a software library called "Wiring" which provides many common functions for developers.

Module-4

7 a. Write a short note on operational states of a sensor node with different power consumptions with figure. **(10 Marks)**

b. Write a detailed note on Optimization goals and figure of merit for wireless sensor networks. **(10 Marks)**

a) **Transceivers operational states:**

Transceivers can be put into different operational states, typically:

Transmit

Receive

Idle – ready to receive, but not doing so some functions in hardware can be switched off, reducing energy consumption a little

Sleep – significant parts of the transceiver are switched off not able to immediately receive something Recovery time and startup energy to leave sleep state can be significant

- **ENERGY CONSUMPTION OF SENSOR NODES:** **.....cntd**
- **Microcontroller energy consumption:** For a controller, typical states are "active", "idle", and "sleep".
  - The energy saving in microcontroller is denoted given by

$$E_{saved} = (t_{event} - t_1)P_{active} - (\tau_{down}(P_{active} + P_{sleep})/2 + (t_{event} - t_1 - \tau_{down})P_{sleep})$$

  - The energy overhead is denoted by $E_{overhead} = \tau_{Up}(P_{active} + P_{sleep})/2$

Examples:

Intel StrongARM:

- ✓ In *normal mode* the power consumption is up to 400 mW.
- ✓ In *idle mode* the power consumption is up to 100 mW.
- ✓ In *sleep mode* the power consumption is up to 50 µW.

Texas Instruments MSP 430:

- ✓ In fully operational mode it consumes 1.2 mW
- ✓ In the deepest sleep mode(LPM4) only consumes 0.3 µW.
- ✓ In the next 3 higher modes consumes only 6 µW.

Atmel ATmega

- ✓ In Atmel ATmega power consumption varies between 6 mW and 15 mW in idle and active modes and is about 75 µW in power-down modes.

Note: Power is energy divided by time. Often units of J/s (joules/second). Gives as Watts.

$$E_{saved} = (t_{event} - t_1) P_{active} - (\tau_{down}(P_{active} + P_{sleep})/2 + (t_{event} - t_1 - \tau_{down}) P_{sleep}). \quad (2.1)$$

Once the event to be processed occurs, however, an additional overhead of

$$E_{overhead} = \tau_{up}(P_{active} + P_{sleep})/2, \quad (2.2)$$

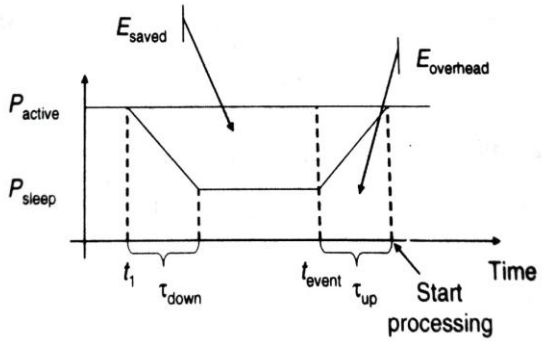


Figure 2.5 Energy savings and overheads for sleep modes

b) **Optimization goals and figures of merit:**

For all WSN scenarios and application types have to face the challenges such as
How to optimize a network and How to compare these solutions?
How to decide which approach is better?
How to turn relatively inaccurate optimization goals into measurable figures of merit?
For all the above questions the general answer is obtained from
Quality of service
Energy efficiency
Scalability
Robustness

Quality of service:

WSNs differ from other conventional communication networks in the type of service they offer. These networks essentially only move bits from one place to another.

Event detection/reporting probability

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

Missing reports -In applications that require periodic reporting, the probability of undelivered reports should be small

Approximation accuracy- For function approximation applications, the average/maximum absolute or relative error with respect to the actual function.

Tracking accuracy Tracking applications must not miss an object to be tracked, the reported position should be as close to the real position as possible, and the error should be small.

Scalability:

The ability to maintain performance characteristics irrespective of the size of the network is referred to as scalability.

With WSN potentially consisting of thousands of nodes, scalability is an obviously essential requirement

The need for extreme scalability has direct consequences for the protocol design

Often, a penalty in performance or complexity has to be paid for small networks

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

If possible, these failures have to be compensated by finding other routes.

8 a. Write a note on embedded operating system suitable for WSN and explain about different programming paradigms. **(10 Marks)**

b. Explain the single node architecture with necessary hardware components. **(10 Marks)**

a) **Embedded operating systems**:

An operating system (OS) is system software that manages computer hardware and software resources i.e acts as an intermediary between programs and the computer hardware.

An embedded system is some combination of computer hardware and software, either fixed in capability or programmable, that is specifically designed for a particular function.

Embedded operating systems (EOS) are designed to be used in embedded computer systems. EOS are able to operate with a limited number of resources. They are very compact and extremely efficient by design

TinyOS:

TinyOS is an open-source, flexible and Application-Specific Operating System for wireless sensor networks.

WSN consists of a large number of tiny and low-power nodes, each of which executes simultaneous and reactive programs that must work with strict memory and power constraints. TinyOS meets these challenges.

Salient features of TinyOS are

- Has Event-based concurrency model
- Component-based architecture.
- TinyOS's component library includes network protocols, distributed services, sensor drivers, and data acquisition tools.
- TinyOS's event-driven execution model

Programming paradigms and application programming interfaces:

Concurrent Programming:

Concurrent processing is a computing model in which multiple processors execute instructions simultaneously for better performance. It is said to be synonymous with parallel processing.

Tasks are broken down into subtasks that are then assigned to separate processors to perform simultaneously.

Process-based concurrency:

It is concurrent (parallel) execution of multiple processes on a single CPU.

Fault-tolerance and scalability is the main advantages of using processes.

It has advantage compared with thread that if they can crash and we can retrieve process perfectly by just restarting them. But if thread crashes, it may crash the entire process.

Event-based programming:

In Event-driven programming the flow of the program is determined by events such as user actions (mouse clicks, key presses), sensor outputs, or messages from other programs/threads.

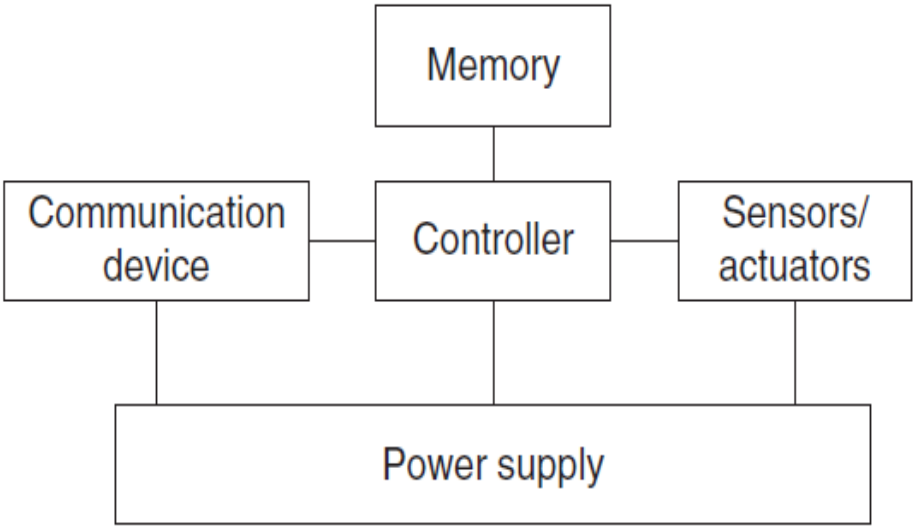
Event-driven programming is the leading paradigm used in Graphical User Interfaces (GUI-type of user interface that allows users to interact with electronic devices through graphical icons).

b) **SINGLE-NODE ARCHITECTURE**

HARDWARE COMPONENTS:

Choosing the hardware components for a wireless sensor node has to consider size, costs, and energy consumption of the nodes.

A basic sensor node contains five main components such as Controller, Memory, Sensors and Actuators, Communication devices and Power supply Unit.



Sensor node Hardware components

Controller: The controller is the core of a wireless sensor node, it process all the relevant data, capable of executing arbitrary code.

It collects data from the sensors, processes this data, decides when and where to send it, similarly receives data from other sensor nodes and decides on the actuator's behavior.

Memory:

Memory is required to store programs and intermediate data; usually, different types of memory are used for programs and data.

Sensors and actuators:

The actual interface to the physical world: The devices that can observe or control physical parameters of the environment.

Communication Device:

To turn nodes into a network a device is required for sending and receiving information over a wireless channel.

Power supply:

As usually no tied power supply is available, some form of batteries are necessary to provide energy.

Sometimes, some form of recharging by obtaining energy from the environment is available as well (e.g. solar cells).

There are essentially two features:

1. Storing energy 2. Energy scavenging

Module-5

9	a.	Explain the crucial points influencing the physical layer of WSN.	(08 Marks)
	b.	Explain Mediation Device Protocol with advantages and disadvantages.	(06 Marks)
	c.	Explain the CSMA protocol with proper flow diagram.	(06 Marks)

a) Some of the most crucial points influencing PHY design in WSNs are:
Low power consumption;
Consequence 1: small transmit power and thus a small transmission range;
Consequence 2: low duty cycle; most hardware should be switched off or operated in a low power standby mode most of the time;
Low data rates (tens to hundreds kb/s);
Low implementation complexity and costs;
Low degree of mobility;
A small form factor for the overall node;
Low cost;

b) mediation device (MD) Protocol uses a mediation device which is available all the time
Allows each node to go to sleep mode periodically and to wake up only for short times to receive packets.No global time reference , node does not take care of neighbour schedule.
Useful where MD is unconstrained.
Protocol
Sender B sends RTS to MD
MD stores this information
Receiver C sends query to MD
MD tells reciever C when to wake up
C sends CTS to B (now in sync)

B sends data
C acknowledges
C returns to old timing
It Avoid useless listening on the channel for messages
Advantages-

Does not require time synchronisation between the nodes only the MD has to learn the periods of the nodes
Most of the power burdon is shifted to MD,other devices can be in sleep mode most of the time and have to spend energy only for periodic becons.
Synchronisation work is done by MD,very low duty cycles can be supported.
Disadvantages:
The becons of different nodes may collide repeatedly when nodes have overlaping wakeup periods.
MD has to be energy independent or unconstrained.
Solution: Distributed Mediation Device Protocol
Nodes randomly wake up and serve as mediation device

c)

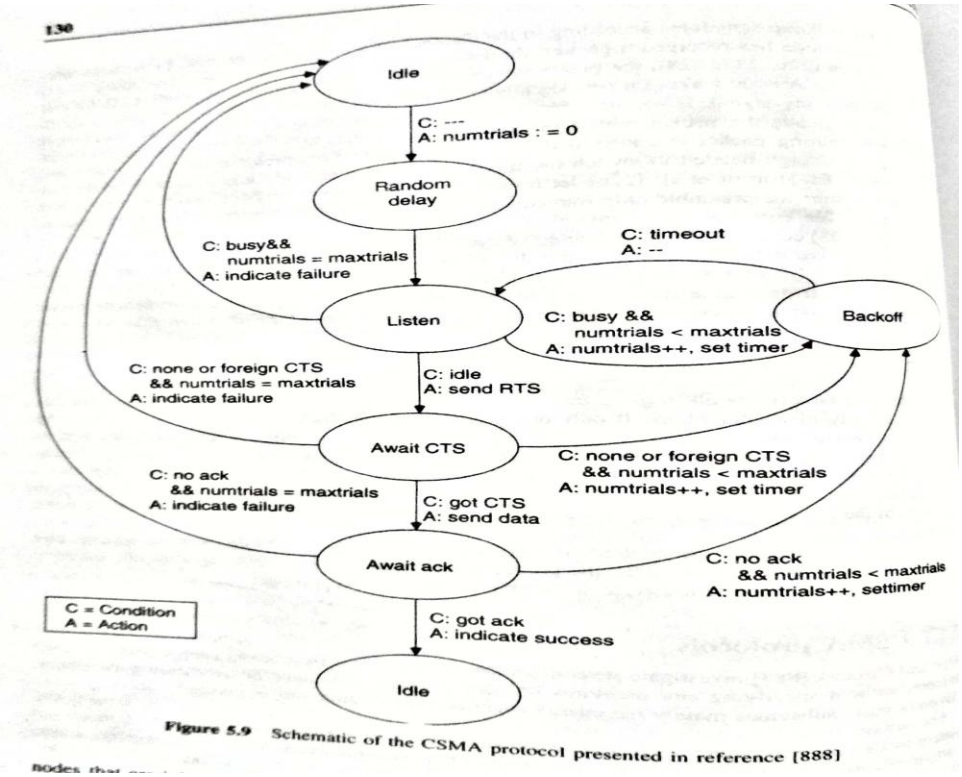


Figure 5.9 Schematic of the CSMA protocol presented in reference [888]

10 a. Explain the S-MAC protocol and explain how it handles the major sources of energy inefficiency in WSN. **(08 Marks)**

b. What is geographical routing and explain about Greedy Perimeter Stateless Routing for wireless networks with proper figure. **(08 Marks)**

c. Explain Leach protocol with necessary figure. **(04 Marks)**

* * * * *

a)

S-MAC protocol was suggested by Ye, Heidemann and Estrin

Tradeoffs: Between latency, fairness and energy efficiency.

Tries to reduce wastage of energy from all four sources of energy inefficiency

Collision – by using RTS and CTS

Overhearing – by switching the radio off when transmission is not meant for that node

Control Overhead – by message passing

Idle listening – by periodic listen and sleep

b) Routing tables contain information to which next hop a packet should be forwarded

Explicitly constructed. Alternative: Implicitly *infer* this information from physical placement of nodes. Position of current node, current neighbors, destination known – send to a neighbor in the right direction as next hop.

Geographic routing

Options Send to any node in a given area – ***geocasting***

Use position information to aid in routing – ***position-based routing***

Might need a ***location service*** to map node ID to node position

The **Greedy Perimeter Stateless Routing in Wireless Networks** is a routing protocol for mobile ad-hoc networks. It was developed by B. Karp. It uses a greedy algorithm to do the routing and orbits around a perimeter. GPSR is a geo routing method, which means that data packages are not sent to a special receiver but to coordinates. The packages should be relayed to the node that's geographically closest to the coordinates. This assumes that every node knows its own position.

c) LEACH-Low Energy Adaptive Clustering Hierarchy

Assumes dense sensor network of homogeneous, energy constrained nodes, which shall report their data to sink node

TDMA based MAC protocol, integrated with clustering and simple “routing” protocol

Partitions the nodes into clusters and in each cluster a dedicated node, the cluster head is responsible for creating and maintaining the TDMA schedule.

Other nodes are member nodes. To all member nodes , TDMA slots are assigned, which can be used to exchange data between the member and cluster head; there is no peer-to-peer communication. When the members don't have their time slot they can sleep. The cluster head aggregates the data of its members and transmits it to the sink node or to other nodes for further relaying. As the sink is often far away, the cluster head must spend significant energy for transmission. For member it is cheaper to reach CH than to transmit directly.CH role is energy consuming as it is always switched on and is responsible for the long range transmission.CH Role is given on rotation basis to all the nodes. Nodes decide independently whether to become a CH hence no signaling traffic required for CH election. Signaling traffic is required to associate the nodes to CH. Decision depends on when it was CH last time if it's been long it is more likely to become CH. Protocol is round based –all nodes decide simultaneously whether to become CH.Subsequently associate themselves to CH.Depending upon the signal strength.CDMA code is broadcasted by CH after the formation of cluster to avoid a situation where a border node belonging to cluster A distorts transmissions directed to cluster B