UNIT: I

INTRODUCTION

PRINCIPLE OF WIRELESS SENSOR NETWORKS.

A Sensor is a device used to gather information about a physical pocess and translate into electrical signals that can be procesed, measured and analysed.

→ The physical proces canbe any real world information like temperature, pressure, light, sound motion position, flow, humidity, radiation etc.

→ A Sensor Network is a structure Consisting of Sensors Competationals units and Communications elements for the purpose of recording, observing and reacting to an event or a phenomenon.

→ The events can like physical world, an Industrial environment, a biological s/m while controlling or observing body can be a Consumer Applications. government, civil, military Or an Industrial entity.

→ Such Sensor Network can be Used for remote sensing medical telemetry, Surveillance, monitoring data collection etc.

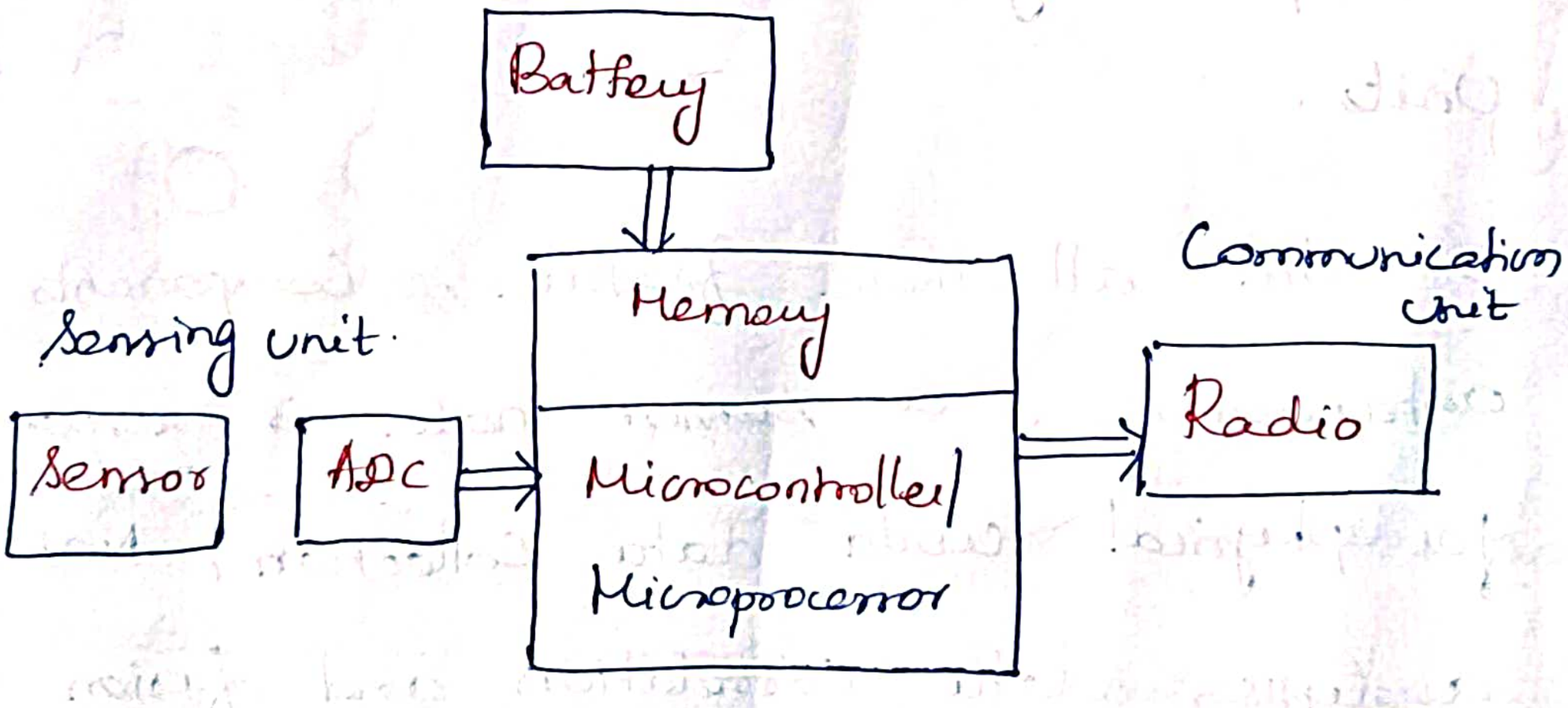
Wireless Sensor Network.

→ A typical Sensor Network consists of Sensor, Controller and a Communication s/m. If the Communication s/m is a Sensor Nw is implemented Using a wireless protocol then the Networks are known as Wireless Sensor Network.

A Sensor Node in a WSN consists of four basic Components

→ Power supply
→ Sensor
→ Processing Unit
→ Communication S/m.

Basic Components of WSN



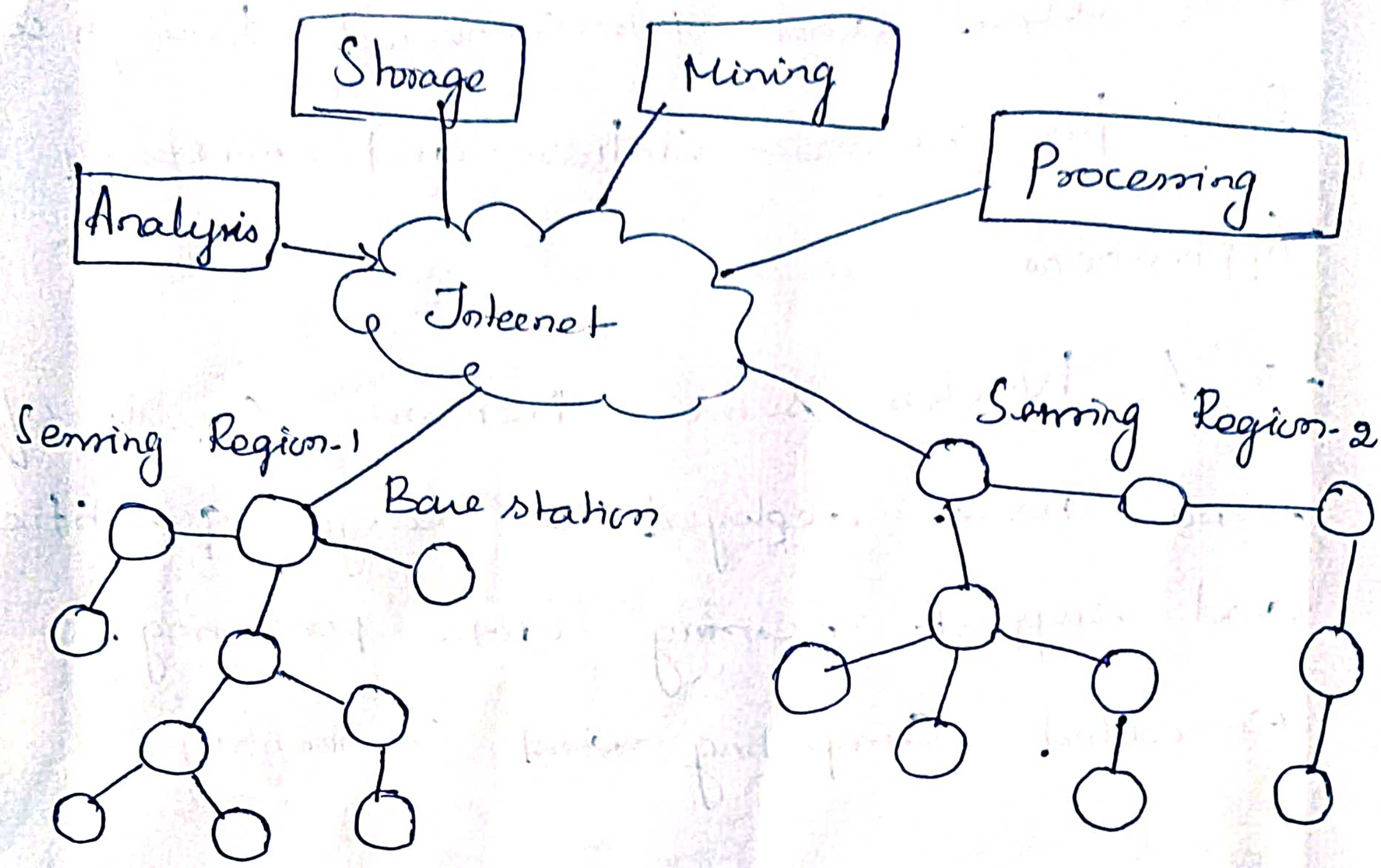
Elements of WSN

→ The sensor collects the analog data from the physical world and an ADC Converts this data to digital data.

→ The Main processing Unit a Microprocessor or a Microcontroller, Performs an intelligent data processing and manipulation, Communication System Consists of radio s/m, a Short range radio for data transmission and reception.

→ A Sensor Node Consists of not only sensing Component but also other important features like processing, Communication and storage Units.

→ With all these features, Components and enhancements, a sensor node is responsible for physical world data Collection, Network analysis, data Correlation and fusion of data from other sensor with its own data.



g

→ According to technologies. Wireless sensor Networks is an Important technology for the twenty first Century.

→ Recent developments in MEMs sensor (Micro Electro Mechanical S/m) and wireless Communication has enabled cheap, low power.

tiny and smart sensor deployed is a wide area and interconnected through wireless links for various civilian and military Applications.

→ A Wireless sensor Network consists of sensor node deployed in large quantities and support sensing data processing. embedded Computing and Connectivity.

Elements of WSN.

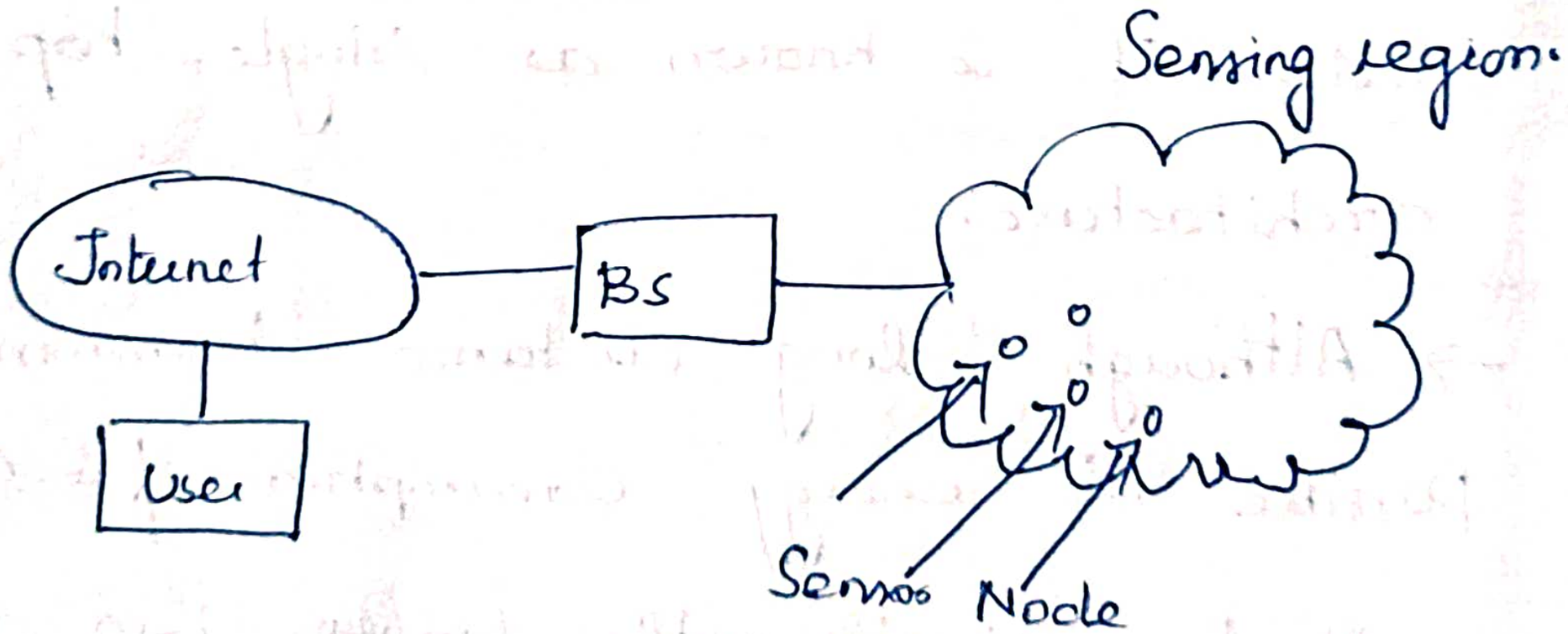
A typical WSN can be divided into two elements i They are.

→ Sensor Node

→ Network Architecture.

NETWORK ARCHITECTURE:

→ When a large number of sensor nodes are deployed in a large area to monitor a physical environment, the networking of these sensor node is equally important. A sensor node in a WSN not only communicates with other sensor node but also with a Base station using wireless communication.



→ The Base station sends commands to the sensor nodes and the sensor node perform the task by collaborating with each other.

The sensor node in turn send the data back to the base station. A base station also act as a gateway to other networks through the internet.

→ After receiving the data from the sensor node a base station performs simple data procesing and sends the updated information to the user using internet.

→ If each sensor node is connected to the base station, it is known as single-hop network architecture.

→ Although long distance transmission is possible the energy consumption for communication will be significantly higher than data collection and computation.

→ This can be implemented is two ways.

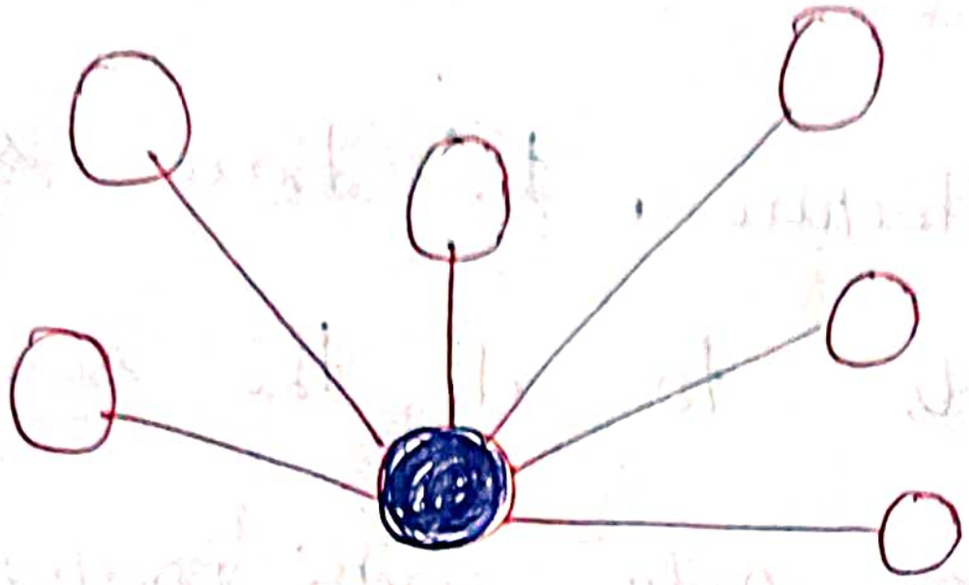
* Flat Network architecture
* Hierarchical Network Architecture.

→ In flat architecture, the base station sends commands to all the sensor nodes but the sensor node with matching query will respond using its peer nodes via a Multi hop path.

→ In hierarchical Architecture, a group of sensor nodes are formed as a cluster and the sensor node transmit data to Corresponding cluster heads.

→ The cluster heads can then relay the data to the base

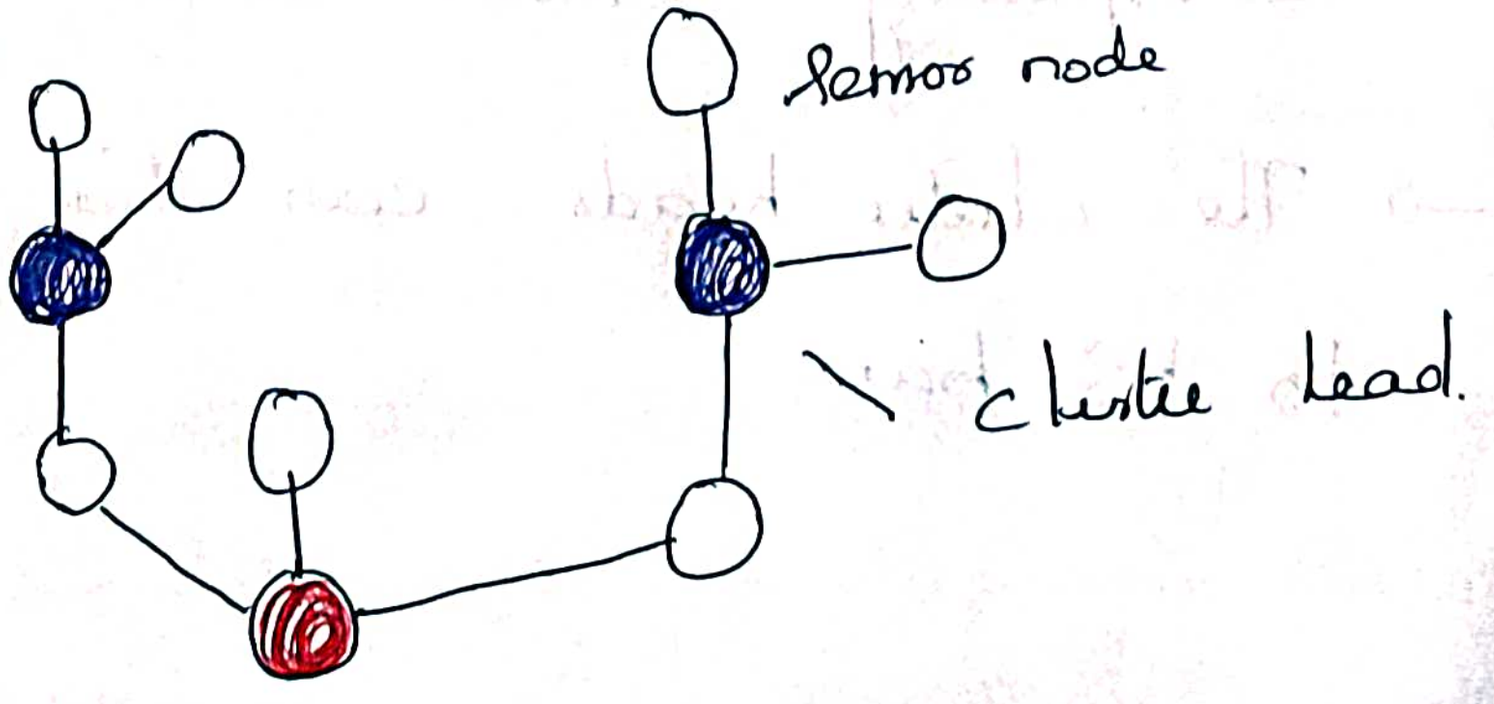
Single Hop Architecture



Base station

Multi-hop Architecture:

Multihop Architecture is usually used. Instead of one single link between the sensor node and the base station, the data is transmitted through one or more intermediate node.



CEC365

Challenges for WSN

Realizing the following characteristics is a major challenge of wireless sensor network

characteristics requirements

Type of Service:

* WSN is expected to provide meaningful information for a task
* WSN moving bits only a means to an end
* scoping of interaction to specific geographic region or to time interval is important
* " People want answers not numbers'

Quality of Service:

* Type of networks service is quality of that service
* QoS comes from multimedia type application like bounded delay or minimum B.W
* In some cases occasional delivery of a packet is enough in other cases high reliability exist
* In some cases delay is important when actuators are to be controlled
* Adapted Quality concepts like reliable detection of events or approximation of quality is important

fault tolerance:

→ when nodes run out of energy it might is damaged or wireless communication between two nodes is interrupted.

→ WSN as whole should tolerate such faults

→ To tolerate, redundant deployment is necessary

Life time:

→ nodes will rely on limited supply of energy (using batteries)
→ Replacing these sources is not practicable
→ lifetime is an important figure of merit
→ A supplement to energy supplies, a limited power source (solar cell) is available on sensor node
→ These does not ensure continuous operation but provide recharging of batteries.
→ Investing more energy can increase quality but decrease lifetime

Scalability:

→ WSN include large no. of nodes, the employed architecture and protocols must be able to scale these numbers

Wide range of densities:

→ In WSN, the no. of nodes per unit area density of network vary.
→ Different application have different node densities
→ The network should adapt to such variations

Programmability:

→ It is not only necessary for the nodes to process information but also to react flexibly on changes
→ so the nodes should be programmable and changeable during new operation and task

a fixed way is insufficient

Maintainability:

→ As both WSN and its environment change, the system has to adopt

→ It has to monitor the status to change operational parameters to choose different trade offs (eg. provide lower quality when resource is scarce)

→ The network should maintain itself and interact with external mechanism to ensure its required quality

Required Mechanism:

→ To realize these requirements innovative mechanism for a Communication n/w have to be found.

→ The Mechanism has to generalise wide range of applications some of the mechanisms of WSN are.

Multihop Wireless Communications:

→ In wireless, direct Communication between a sender and a receiver is faced with limitations.

→ For long distance high transmission power is required

→ The use of intermediate nodes reduce the total required power.

→ So in WSN multihop Communication is most necessary.

Energy Efficient Operation:

→ To support long lifetime, energy efficient Operation is a key technique.

→ Non homogenious energy Consumption forming hot spots is an issue.

Auto Configuration:

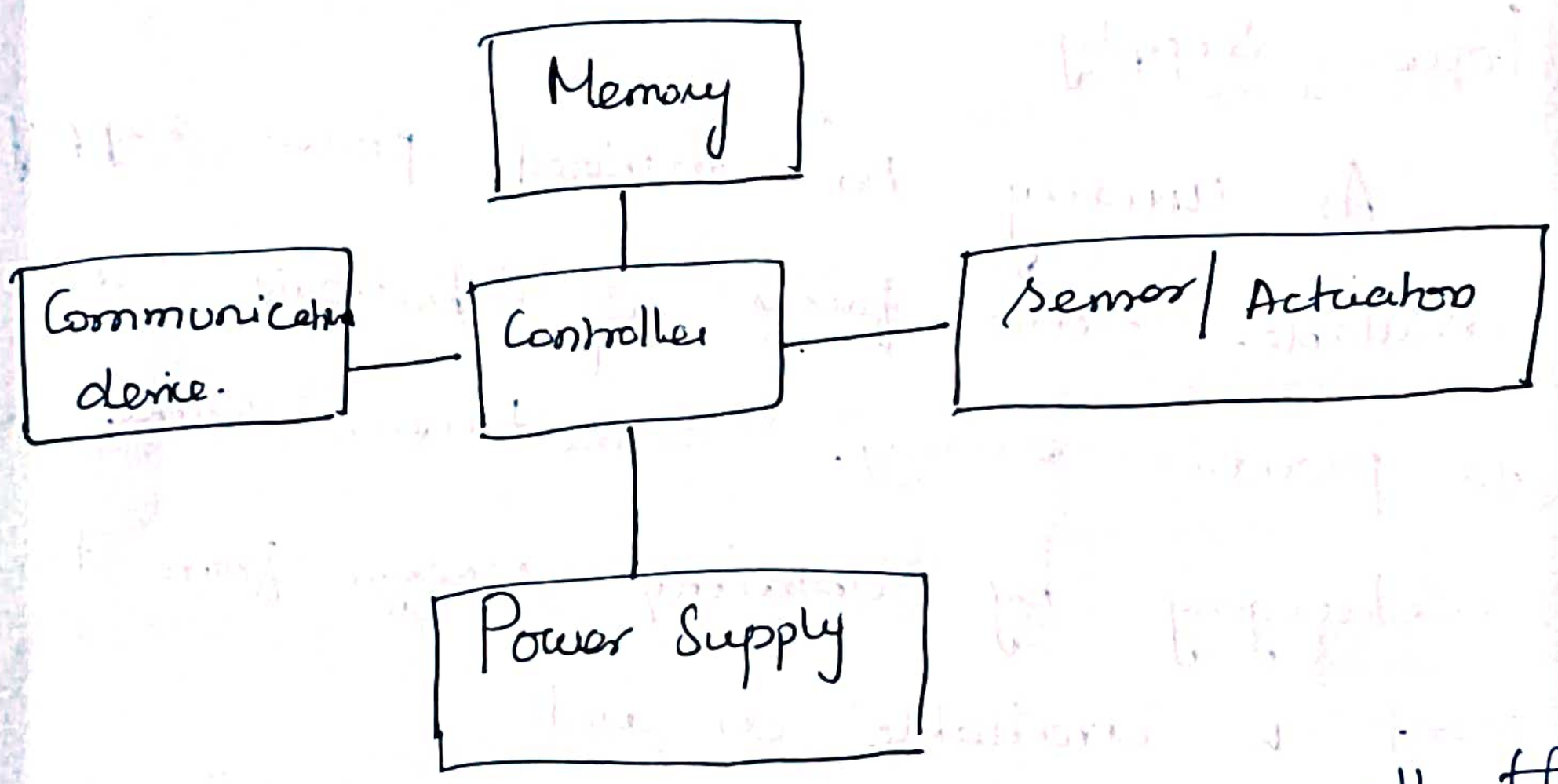
WSN Configure most of its

Single Node Architecture:

→ The principle task of a node is Competation, Storage. Communication and sensing / Actuation

→ The energy Consumption, energy gathering and saving is also an important task of node.

Major Components in Sensor Node:



Controller: A Controller to process all the relavent data, capable of executing arbitary Code:

Memory: Some memory 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, device that can observe or control physical Parameters of the environment.

Communication: Turning node into a network requires a device for sending and receiving information over a wireless channel.

Power Supply:

As usually no tethered power supply is available. Some forms of batteries are necessary to provide energy. Sometimes, some form of recharging by obtaining energy from the environment is available as well.

Ex: Solar cell.

Each of these components has to operate balancing the trade-off between as small energy consumption as possible on the one hand and the need to fulfill their task on other hand.

For example: both the communication device and the controller should be turned off as long as possible.

Ex: Use a preprogrammed timer to the reactivate after some time.

Alternatively, the sensor could be programmed to raise an interrupt if a given event occurs.

→ A temperature value increases / exceeds a given threshold or the Communication device detects an incoming transmission. Supposing such an event function requires Appropriate Interconnection between the individual Components. Moreover, both Control and data information has to be exchanged along these interconnections.

The Interconnection can be very simple. For example, a sensor could simply report an analog value to the controller.

→ It could be endowed with some intelligence of its own, preprocessing sensor data and only waking up the main Controller if an actual event has been detected.

For Ex: detecting a threshold crossing for a simple temp sensor.

Controller:

→ It is the core of wireless sensor node

→ It collects data from sensors, process data and decides when and where to send it, receive data from other sensor nodes

→ It has to execute various programs it is the central processing unit of the Node

→ Variety of processing tasks are performed on various Controller Architecture, representing

trade off between flexibility, Performance, energy efficiency and costs.

→ Simple processors like microcontroller are best suited due to their flexibility in connecting with other devices like sensors

→ Their low power consumption, and their instruction set amendble to time critical signal procesing and memory built in make them fase more flexible.

→ Micro Controllers are suitable for WSN as they enter the 'sleep mode' only when only part of Controllers are active.

→ Dsp are used for procesing large amtof Vectorial data in signal procesing Applications.

→ Sine in WSN the signal procesing task related to sensing of data is not ener Complicated and hence they are not Usually Used.

→ FPGA Cannot be reprogrammed at same frequency as micro controller.

→ ASIC provide the same function is potentially more costly hardware.

→ In WSN, bigger flexibility and simpler Usage make microcontroller the superior solution. Ex: μC

Ex:

Texas Instruments MSP 430 (16 bit) at 4 MHz freq.

SA – 1100 model → 32 bit RISC at 206 MHz

Atmel A-Tmega 128L – 8 bit processor

Used in Embedded Application.

Memory:

RAM is used to store intermediate Sensor readings Packets from other nodes

→ RAM is fast. but loses its content if power supply is interrupted

→ Program is stored in Read only Memory or Electrically Erasable Programmable Read only Memory (EEPROM)

→ Flash Memory also serve as intermediate storage of data if RAM is ~~if~~ insufficient or when power supply is shot down for RAM for some time.

→ Correct dimensioning of RAM is crucial with respect to cost and Power consumption.

Communication Device:

→ It is used to exchange data between individual nodes.

→ Wired Communication is frequently applied in many sensor network like Actings.

→ The Communication device for these networks are Custom off - the shelf Components

→ In Wireless Communication, Choice is made on transmission medium.

→ The medium are radio frequencies, optical Communication and Ultra sound.

→ Radio frequency (RF) based Communication best fits WSN.

→ It provides long range, high data rates and acceptable error rate and does not require the line of sight between sender and receiver.

Transceiver and Design Considerations:

→ Both transmitter and receiver are required in sensor node.

→ The main task is to convert a bit stream coming from microcontroller and convert them to radio waves.

→ The combined device to perform the two tasks in single entity is called transceivers.

→ Half duplex operation is performed.

→ low cost transceivers is commercially availdble for transmitting and receiving.

Transceiver task and Characteristics.

Service to upper layer:

→ A receiver has to offer certain services to upper layer i MAC.

→ The transceiver must provide an interface to allow MAC layer to intiate frame transmission., power consumption and energy efficiency.

→ Transceivers must be suitable between different states Ex: Active and sleeping

Carrier frequency and Multiple Channels:

Transceiver are available at deffarent carrier frequencies.

It must match application requirements and regulatory restriction.

State change times and energy:

A transceiver can operate in different modes sending or receiving, use different channels, or be in different power safe status.

Data rates:

Different data rates can be achieved by using different modulation or changing the symbol rate.

Modulation:

The transceiver support one or several of ON/OFF keying, ASK, FSK, or similar modulation.

Coding:

Transceiver allow various coding schemes

Transmission Power Control.

Range:

Range depends on Maximum transmitin power on ~~another~~ antenna characteristics on attenuation caused by environment, which depends on carrier frequency. It ranges from few meters and several hundreds of meters.

Blocking Performance:

It is achieved bit error rate of receiver in presence of interfeer

Out of band Emmon.

To limit the disturbance of other sys. the transmitter should produce. the transmision power a little outside the prescribed Band width.

Noise figure:

It is defined as the ratio of signal to noise Ratio (SNR) at the input of the element to SNR ratio at the elements Output.

$$NF = \frac{SNR_i}{SNR_o}$$

$$NF_{dB} = SNR_i\,dB - SNR_o\,dB.$$

Gain: It is the ratio of o/p signal Power to input signal power.

Powr efficiency:

It is defined as the ratio of radiated Power to overall power consumed by frontend.

Receiver Sensitivity:

It specifies the minimum signal Power at the receiver to achieve the prescribed error bit rate.

Carrier sense and RSSI:

The receiver must able to provide information whether channel or carrier is busy (another node is transmitting)

IEEE 802.15.4 has following modes

- The received energy is above threshold.
- A carrier has been detected
- Carrier detected and energy is present.

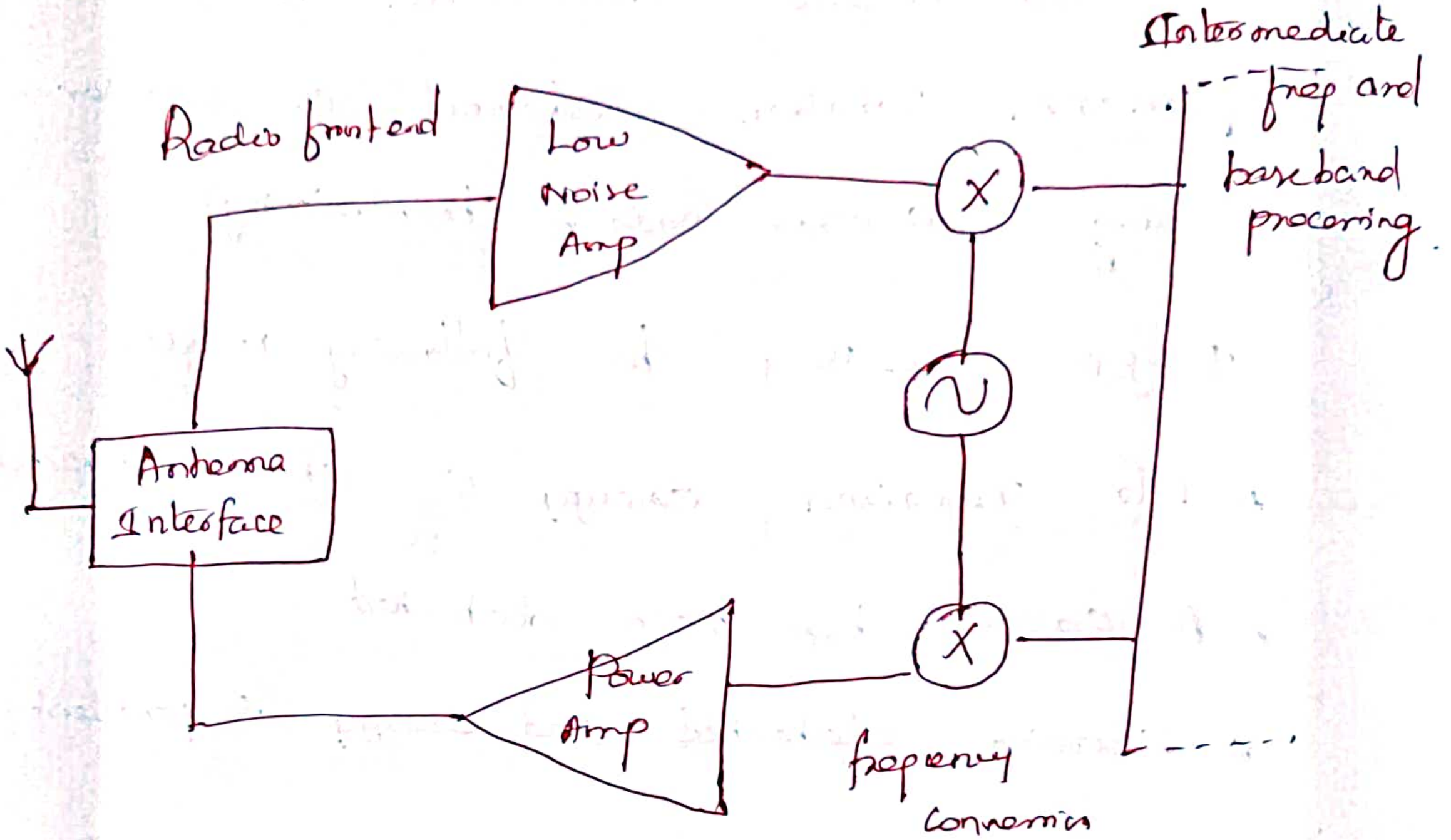
Frequency stability:

It denotes the degree of variation from nominal centre frequencies when environmental conditions of oscillator, like temperature or pressure change.

Voltage Range:

The transceiver should operate over a range of supply voltage.

Transceiver Structure:



→ The radio frequency front end performs analysis of signal processing.

→ The baseband processor performs all signal processing is digital and Communicates with sensor nodes.

Power Amplifier (PA)

It accepts the unconverted signal from IF or baseband port and amplifies them

for transmission over antenna.

Low Noise Amplifier (LNA)

It amplifies the incoming signal suitable for further processing without reducing SNR.

Local oscillator (or) VCO and Mixer.

They are used for frequency conversion from RF spectrum to Intermediate frequencies to baseband.

Transmit

The transmit part of transceiver is active and the antenna radiates power.

Receive:

The receiver parts is active.

Idle:

A transceiver is ready to receive, but not currently receiving to be in idle state.

→ Many parts are active and others are switched off.

Sleep:

In sleep state, significant parts of transceiver are switched OFF.

→ In complete power down includes complete Initialisation of Configuration of reader.

→ In lighter sleep modes, the clock driven parts is throttle down and operational state is remembered.

Physical Layer And Transceiver design Consideration in WSN's.

Some of the most crucial influencing PHY design in Wireless Sensor networks are

→ Low power Consumption

→ As one consequence : Small transmit Power and thus a small transmission range.

→ As a further consequence : low duty cycle Most hardware should be switched off or operated in a low power standby mode most of the time.

→ Comparably low data rates, on the order of tens to hundreds kilobits per second.

→ Low Implementation complexity and costs.

→ low degree of mobility

→ A small form factor for the overall node.

Energy Usage Profile

→ The choice of a small transmit power leads to an energy consumption profile different from other wireless device like cell phones.

→ First the radiated energy is small, typically on the order of 0dB. On the other hand, the Overall transceiver consumes much more energy than is actually radiated.

→ For a radiated power of 0dBm, the transmitter Uses actually 32 mW. Whereas the receiver Uses even more, 38 mw.

→ For the mica motes, 21 mw are consumed in transmit mode and 15 mw in receive mode. These numbers coincide well with the observation that many practical transmitter designs have efficiencies below 10% at low radiated power.

→ A second key observation is that for small transmit powers the transmit and receive modes consume more or less the same power, it is even possible that reception require more power than transmission, depending on the transceiver architecture, the idle mode's power consumption can be less or is the same range as the receive power.

→ To reduce Average power consumption is a low traffic wireless sensor network, keeping the transceiver in idle mode all the time would consume significant Amounts of energy.

→ Therefore, it is important to put the transceiver into sleep state instead of just idling.

→ It is also important to explicitly include the received power in to energy dissipation models. Since the traditional assumptions that receive energy is negligible is no longer true.

→ However, there is the problem of the start up energy / start up time, which a transceiver has to spend upon waking up from sleep mode.

→ A third key observation is the relative costs of communications versus computation in a sensor node. clearly a comparision of these costs depends for the communication part on the BER requirements, range, transciver type.

Choice of Modulation Scheme:

A crucial point is the choice of modulation scheme, several factors have to be balanced here: the required and desirable data rate and symbol rate, the implementation complexity the rationship between radiated power and target BER and the expected channel characteristics.

To Maximize the time a transceiver can spend in sleep mode, the transmit times should be minimized. The higher the data rate offered by the transciever / modulation the smaller the time needed to transmit a given amount of data and, consequently, the smaller the energy consumption.

NETWORK ARCHITECTURE

→ A Sensor Node can gather information from other sensor node.

Two types.

1. Layered Architecture
2. clustered Architecture.

Layered Architecture

→ It consists of single powerful base station

→ It uses military sensor based infra structure

→ Base station acts as fixed point to a wired Network.

→ The small sensor nodes from a wireless backbone

→ It Uses Unified Network Protocol framework (UNPF)

Operation 1 : Network initialisation and Maintenance

→ The protocol organizes the sensor node into different layer.

→ Base station Communicate will all nodes using one hop.

→ BS broadcasts its identifies to sensor nodes using CDMA.

→ Each node sends its ID at lowest Power level.

→ The layer one node form layer two with nodes which are one hop away from layer one node.

Operation 2: MAC:

The Distributed TDMA Receiver Oriented channel assignment MAC protocol is used.

→ Base station assigns a reception channel to each node and channels are reused to avoid collisions.

→ The node schedules transmission slots for all its neighbours and broadcasts the schedule so that collision free transmission is possible. This in turn saves energy as the nodes are turned off when they don't send/ receive information.

→ Main functions are

- Channel Allocation → Assign reception channel to the nodes
- Channel scheduling → Shares reception channel among the neighbours.

→ 'DTRoC' uses Suitable Channel Allocation Algorithm to avoid problems of Hidden and Exposed terminal.

Routing Protocol.

→ The data transfer from the sensor nodes are transmitted to base station by multihop data forwarding.

→ The node is selected to forward the packet by considering the remaining energy of the nodes so that higher network lifetime is established.

→ Only one node of next layer maintains the routing table and the existing adhoc routing protocol can be simplified for the layered architecture.

UNPF – R

UNPF-R is the modified form of UNPF Protocol. The sensor nodes can adapt to their transmission range to optimize the network performance.

→ If the transmission range is small, it leads to new partitioning, while the large transmission range reduces the frequency reuse.

* optimal range is determined by a centralized control algorithm which evaluates the objective function periodically. The objective function is

$$f(R) = \frac{\varepsilon \times d}{n/N}$$

* $R \rightarrow$ transmission Range

$N \rightarrow$ total number of sensor in the sim

$n \rightarrow$ number of nodes in layer 1

$\varepsilon \rightarrow$ Energy consumption per packet

$d \rightarrow$ Average packet delay.

→ A New transmission range R' is selected by BS as follows.

→ If no packet is received by the base station from any sensor node for particular

time interval. then the transmission range is increased by Δr with probability $(1 - 0.5 \frac{n}{N})$ else the transmission range is decreased by Δr with probability $0.5 \times \frac{n}{N}$.

→ The objective function is reevaluated with new transmission range such that if $f(R') < f(R)$, the transmission Range 'R' is adopted. Else 'R' is modified to R' with probability

$$e^{\frac{f(R) - f(R') \cdot (n/N)}{T}}$$

T → temp. Parameter

Advantages of UNPF - R

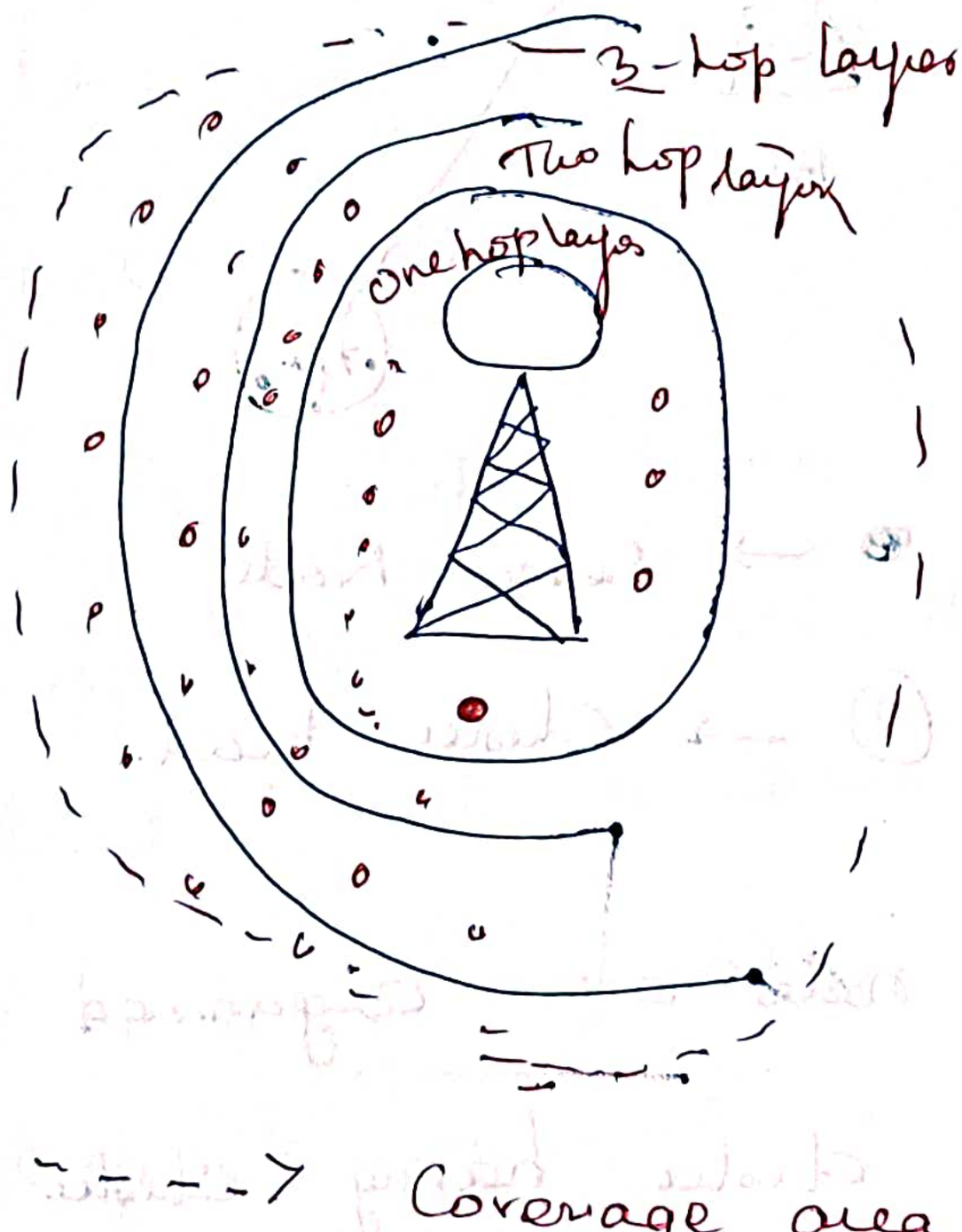
* Minimum Energy * delay metric ie delay is minimum with reduced energy consumption.

Maximum Number of nodes can be connected to base station.

Advantages of Layered Architecture.

* Each node is involved to minimum distance

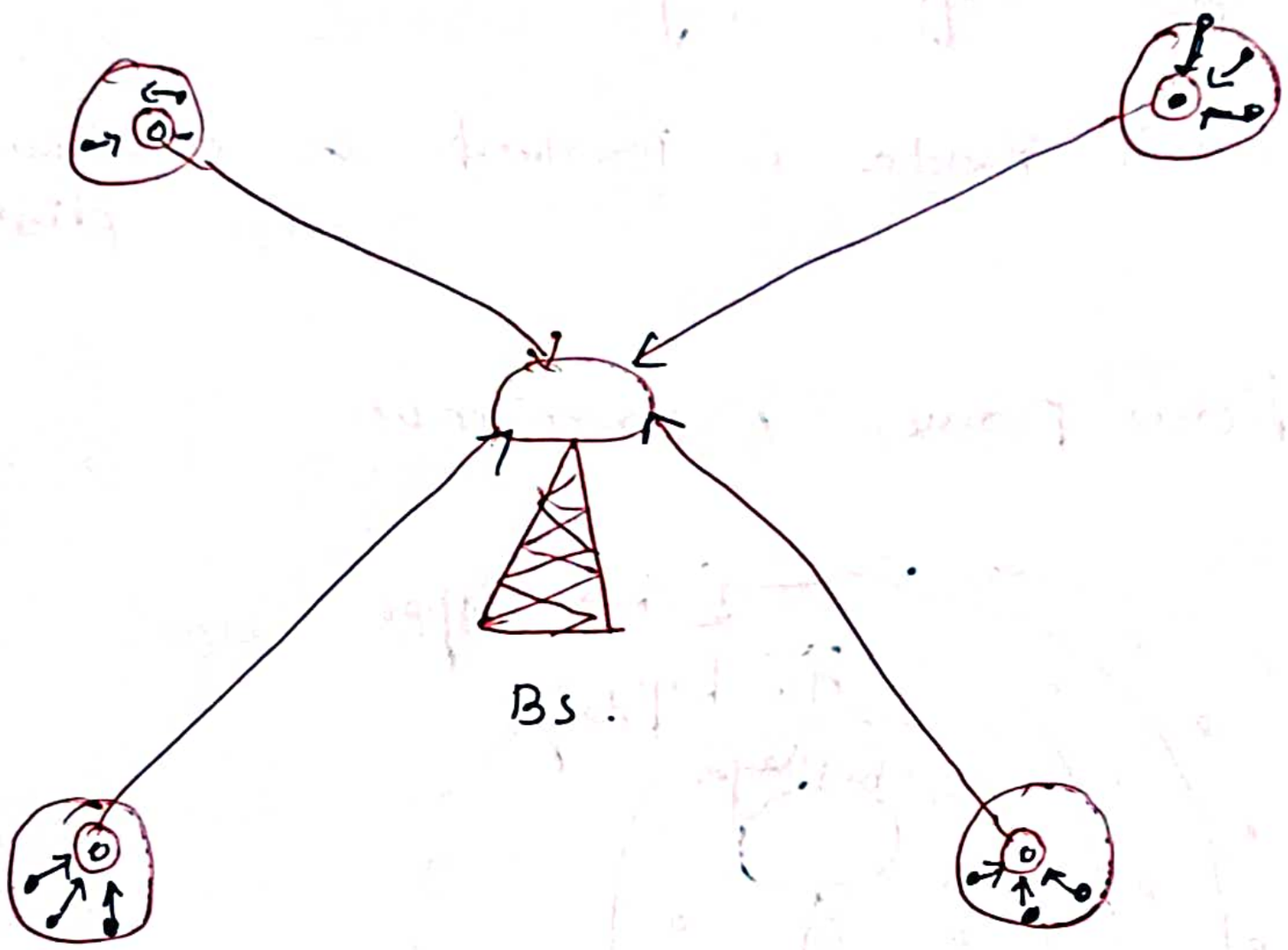
* Low power transmission.



- - - -> Coverage area

◉ → Sensor node.

Clustered Architecture.



● → Sensor Node

Ⓒ → Cluster head.

The sensor nodes are organized into cluster which each cluster having cluster head.

The nodes in each cluster exchange message with their respective cluster heads,

The nodes in each cluster exchange message with their respective heads. Which intern send message to a base station connected to a wired N/w.

→ Any Message can reach bs in maximum two hops and the clustering can be extended hierarchically to greater depths.

* Due to data fusion, clustered architecture is useful for sensor Networks. The data gathered by all members of the cluster can be fused at the cluster lead and the resulting information is sent to the base station.

→ The self Organizing sensor networks are used such that cluster formation and selection of cluster leads are automated and distributed.

This process is achieved through n/w layer protocols such as Low Energy Adaptive Clustering Hierarchy (LEACH)

GATEWAY CONCEPTS

→ Need For Gateways

→ For practical deployment, a sensor Network only concerned with itself is insufficient

→ The N/w should interact with other information devices Ex: temp sensor.

→ To exhibit robustness

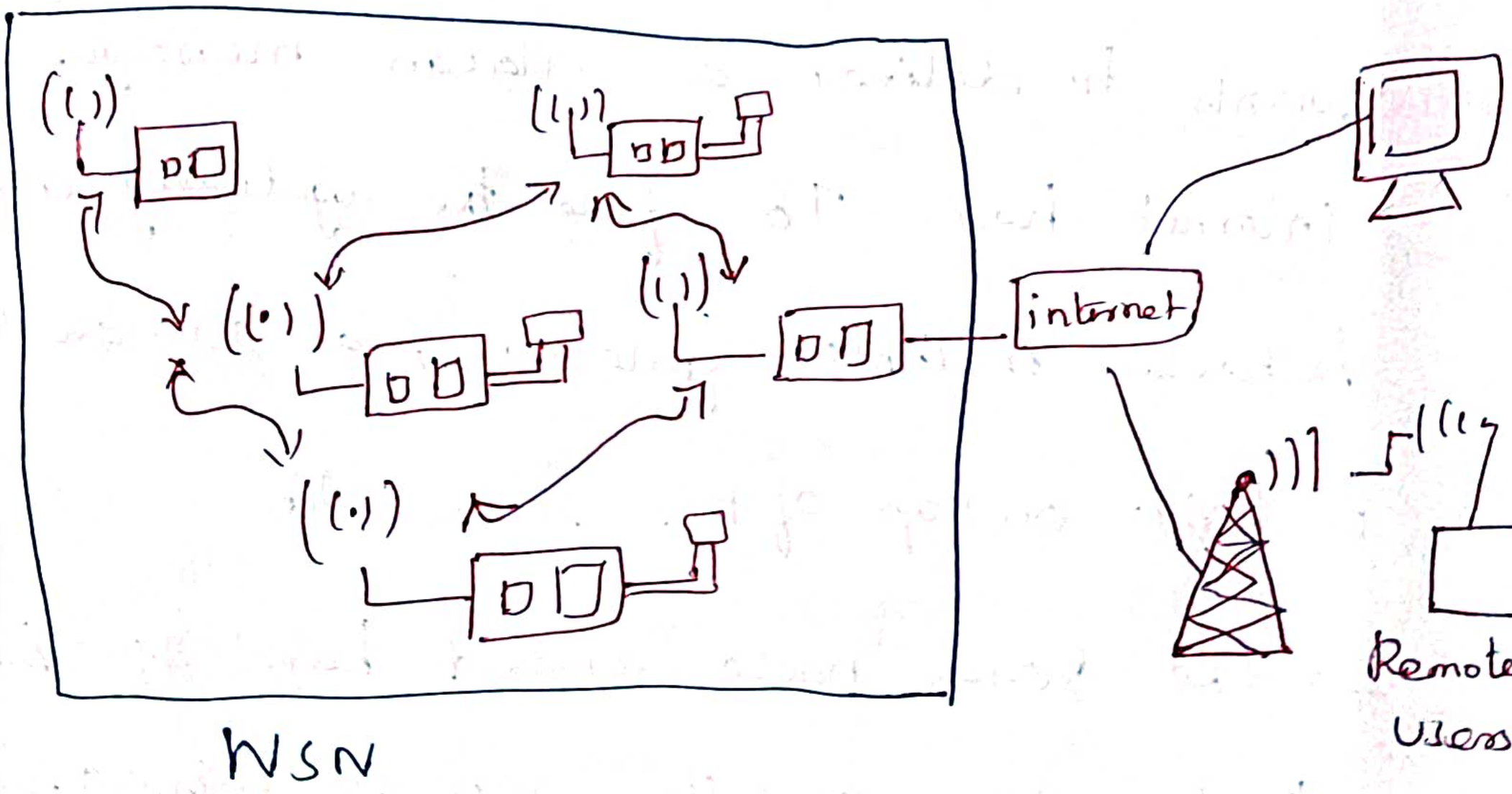
→ The nodes should not fail due to limited number of nodes run out of energy or environment changes or Link breaks.

→ The failure should be compensated by finding other ~~not~~ routes.

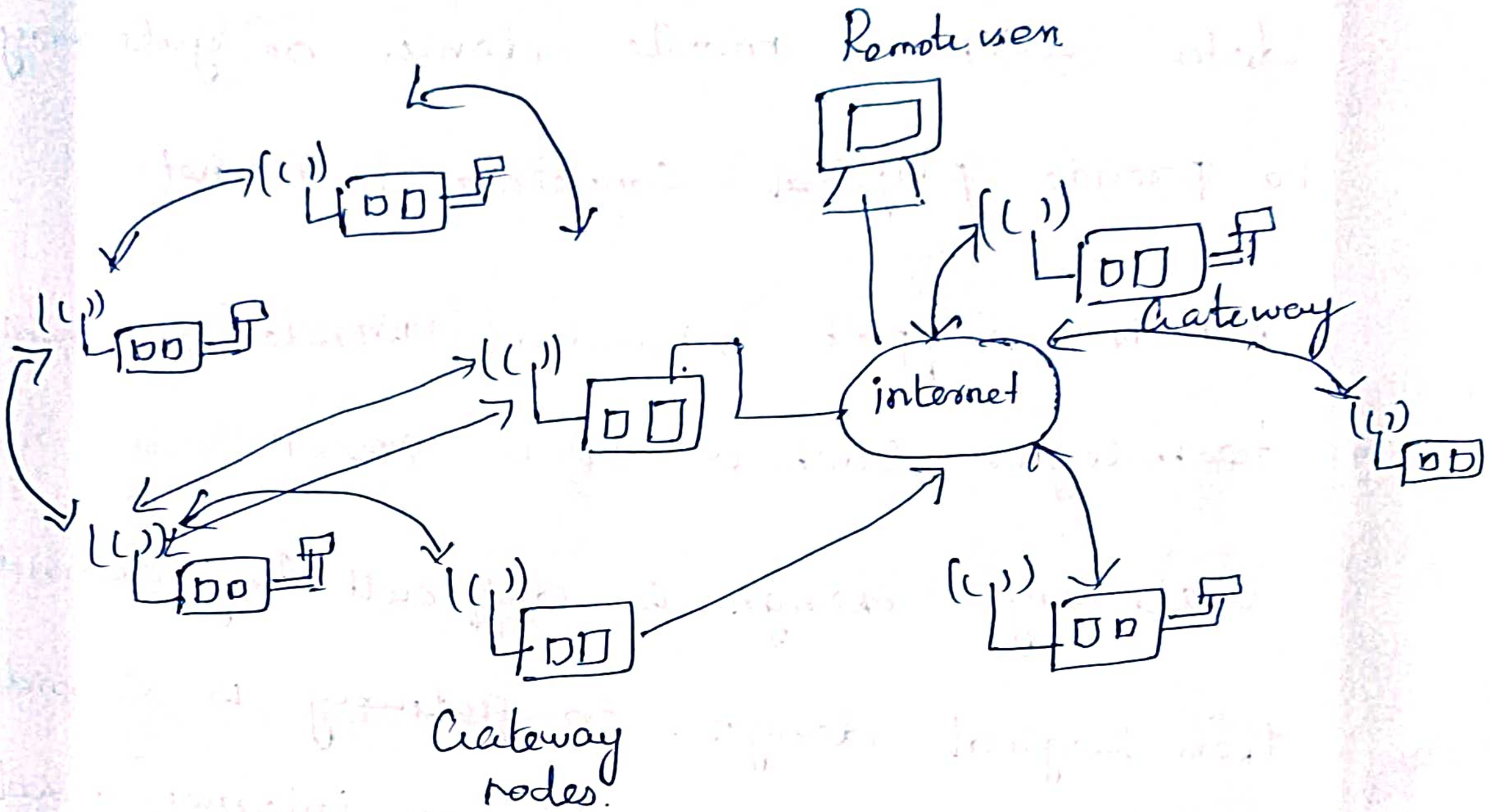
→ The WSN should be able to exchange data with a mobile device or gateway to provide physical connection to Internet.

A WSN support standard wireless communication technologies such as IEEE 802.11

Gateway design is difficult by considering their logical design. So gateway is considered as a simple router between internet and Sensor Networks.



WSN to Internet Communication.



→ Assume a node in WSN initiates WSN to internet Communication Ex: a sensor nodes wants to deliver a alarm message to some internet host. To find the gateway and choose between multiple gateways, an IP overlay N/w is built on top of the sensor N/w.

→ The sensor node should have IP Address Port numbers, and its own packets. The gateway

Should extract this information and translate it into Ip Packets.

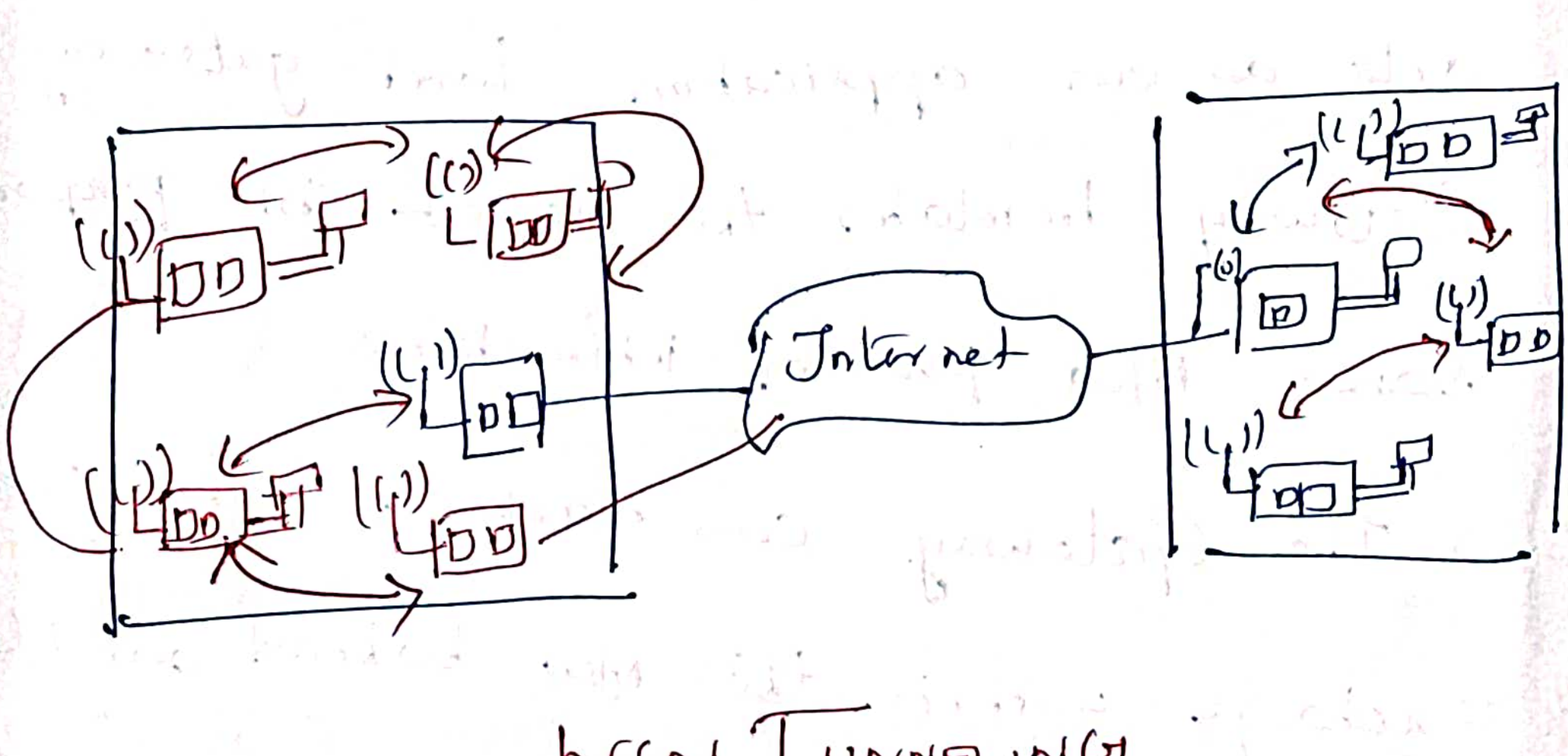
Internet to WSN Communication:

A terminal far away request to Communicate with any sensor node require gateway node The requesting terminal send a properly formatted request to this gateway which acts as an application level gateway. the Gateway translates this request into proper Intra Sensor N/w protocol interaction.

→ the Gateway can make a data centric exchange within the N/w behind an identify Centric exchange used is the Internet. Web service protocol explicity describe Services and the way that can be accessed.

WSN Tunneling:

Virtual WSN is built transperently tunneling all protocols message blw two nlw using internet as a transport N/w. Gateway act as simple extensions of one WSN to another WSN.



WSN TUNNELING

IEEE 802.15.4 MAC Protocol:

IEEE 802.15.4 is a standard which specifies the physical layer and media access control for Low rate wireless personal area Networks.

→ This is the basis for the zigbee, MIWI, wireless HART and thread specifications

→ IEEE 802.15.4 offers the lower networks layer of a wireless personal Area n/w. which focuses on Low cost, low speed Communication bhw devices.

Features:

Suites for real time Applns

Collision avoidance through CSMA/CA

Secure Communication

Power Management functions such as link quality and energy detection.

Characteristics..

→ Fully handshake protocol for transfer reliability.

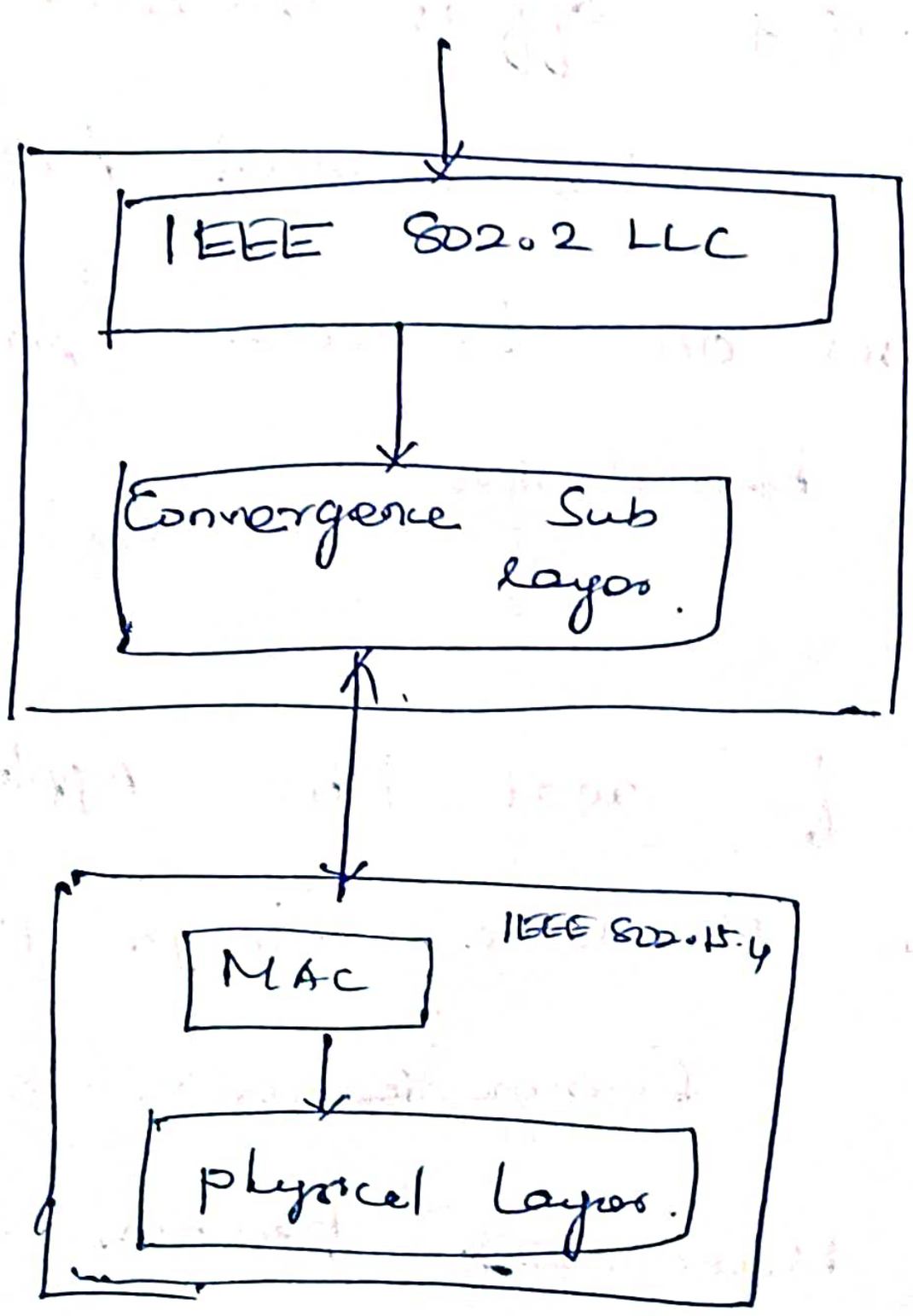
→ Low power consumption

→ Three frequency band operation.

→ Support low latency devices

→ Dynamic device Addressing.

Architecture.



Physical layer

* It is the initial layer of OSI model.
* The physical layer provides the data transmission service as well as the interface to the physical layer management entity which store the database of information related to personal area Nlws.
* The PHY Manages the physical RF transceiver and performs channel selection and energy and signal Management function.

→ It operates on one of the three possible freq bands:

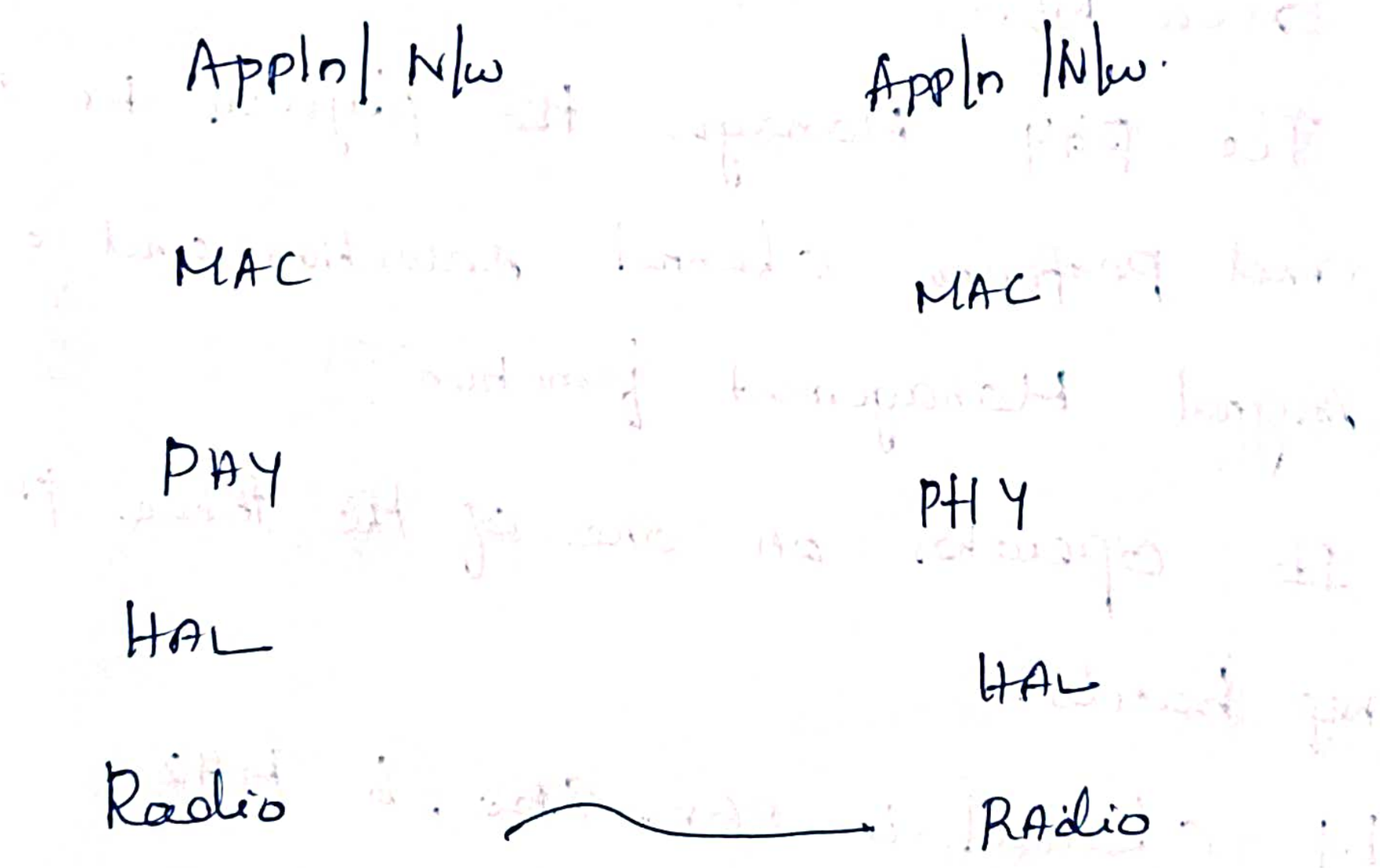
1. 1 channel in 868.868.6 MHz
2. 10 channels in 902 - 928 MHz.
3. 16 channels in 2400 - 2483.5 MHz.

MAC LAYER

→ The MAC enables transmission of MAC frames through the use of Physical channel

→ In Addition to data service, it also offers a Management interface and manages access to the physical channel and N/w beaconing

→ It also controls frame validation, guarantees time slots and handles node associations



IEEE 802. 15.4 device

→ Full Function device

→ Restricted Function device

* Ans the International Standards Organization Created a Communication N/w model, the IEEE standards modified the OSI model to suit its requirements

Appln / Network layer

→ The Appln / Nlw layer of the Software solution are necessary for final Software are necessary for final software solution.

→ It Consists of the highest level of the Software Control hierarchy and direct the MAC layers to perform lower level functions.

MAC Layer

→ Generate network beacons for the co-ordinating device

→ Synchronize the beacons

→ Supports PAN Associations & disassociation

→ Supports device security

→ Employ the CSMA/CA Mechanism for channel access

→ Handle and maintains the GTS Mechanism

→ provide a reliable link btw two peer MAC entities.

physical layer.

→ Activation & deactivation of the radio transceiver

→ Channel frequency selection

→ Data transmission & reception

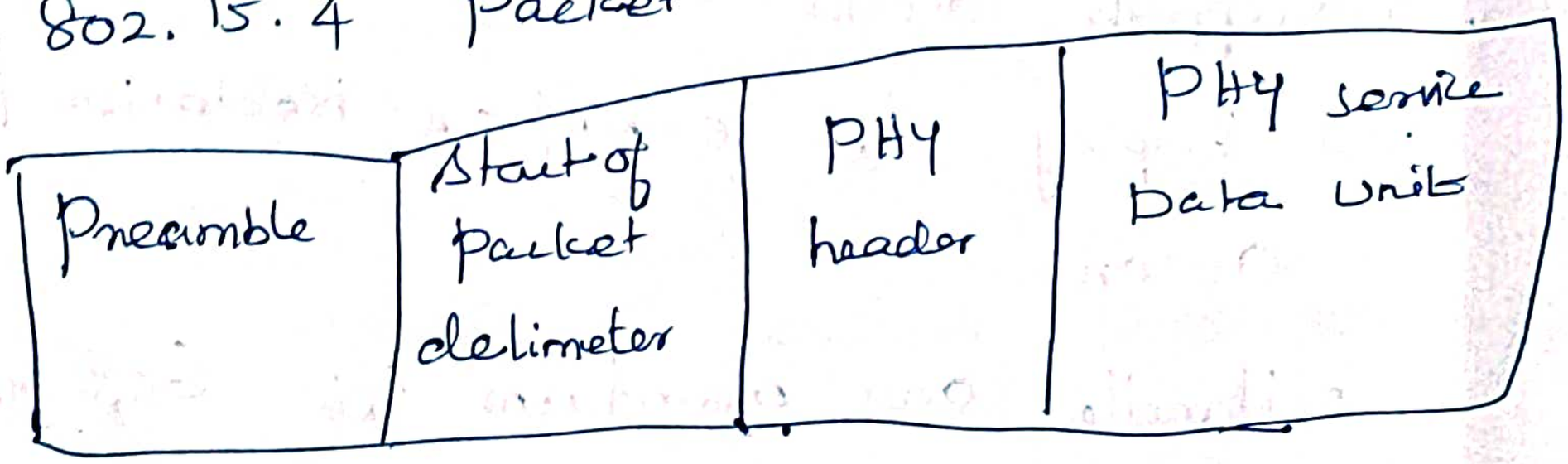
→ CSMA

HAL (Hardware Abstraction Layer)

The HAL layer is specific to a particular radio and interfaces the common PHY layer to the physical radio.

HAL module can be carried to meet the requirement of various radio.

802.15.4 Packet Structure:



Preamble (32 bits) used for synchronization

Start of packet delimiter (8 bits)

PHY header (8 bits) - specifies the length of PSDU

PSDU (0 to 1016 bits) - data bits.

The beacon mode is used with star-type N/ws that are configured with the n/w Management node, referred to also as the "PAN Co-ordinator" at the core.

* The PAN Co-ordinator sends off a beacons signals a fixed intervals, while other nodes Synchronize with this beacon signal and perform communication during allotted periods.

* Since only the node that has been singularly assigned by the Co-ordinator gains exclusive use of the channel, it becomes possible to conduct communication in which no collisions.

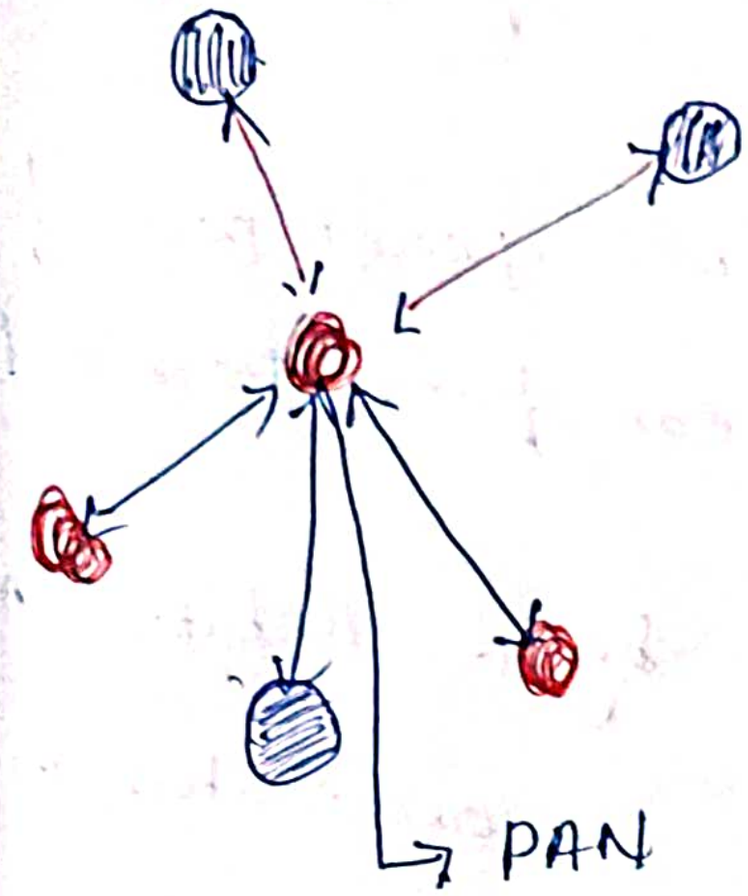
can occur. This mode, therefore, is used for communications requiring low delay levels.

→ The Non-beacon mode: On the other hand is a mode wherein the channel is accessed constantly with the CSMA/CA.

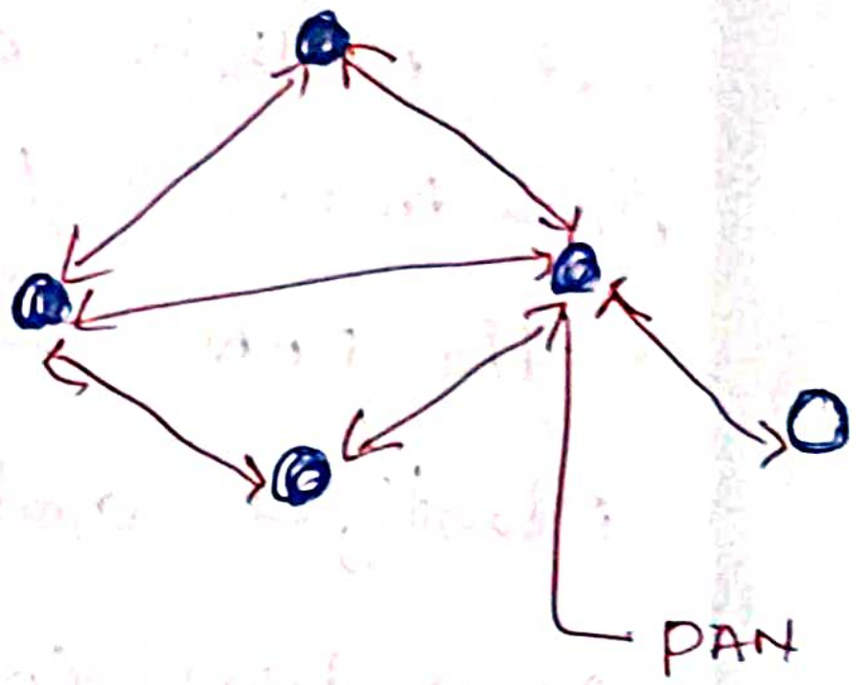
→ When this mode is used in a mesh link with which direct communication are conducted with peripheral nodes. Each node can directly communicate with any other node at any time. While each node can also be ready to receive data addressed to that node, at all times.

→ This need for each node to constantly remain in a reception standby condition make it impossible to conserve energy by conducting intermittent operations as is the case of the beacon mode.

Star topology



Peer to peer topology



— Full function device

— Reduce function device

↔ → Communication flow.

→ A full function device (FFD) can operate in three different roles: it can be a PAN Co-ordinator, a simple Co-ordinator (or) a device.

→ A Reduced function device can operate only as a device.

→ A device must be associated to a Co-ordinator node (which must be a FFD) and

Communicate only with this way forming a star

N/w:

→ Co-ordinator can operate in peer to peer fashion and multiple co-ordinators can form personal Area Network (PAN)

→ The PAN is identified by a 16 bit PAN identifier and one of its Co-ordinators is designated as a PAN Co-ordinator.

The Co-ordinator handle the following tasks.

→ It manages a list of associated devices. Devices are required to explicity associate and disassociate with a Co-ordinator using certain signalling packets.

→ It allocates short Addresses to it devices. All IEEE 802.15.4 nodes have a 64 bit device address. When a device associates with a Co-ordinator, it may request assignment of a 16 bit short address to the subsequently is all Communications between device and Co-ordinators.

→ The assigned address is included in the association response packet issued by the co-ordinator.

→ In a beaconed mode of IEEE 802.15.4, it transmit regularly frame beacon packets announcing the PAN identifier, a list of out standing frames and other parameters. furthermore, the co-ordinator can accept and process requests to reserve fixed time slots to nodes and the allocation are indicated in the beacon.

→ It exchanges data packets with devices and with peer Co-ordinators.

Advantages:

→ Low data transmission rate is the range 250kb

→ Energy Conservation

→ Long battery life

→ Simple and easy to implement

→ Handle upto 64,000 device simultaneously

Dis Advantages.

→ Limited data transfer ability.

ZIGBEE PRO Features:

Zigbee PRO: Published in 2007.

Stochastic Addressing:

A device is assigned a random Address and announced, Mechanism for address Conflict resolution. Parents don't need to maintain assigned address table.

Link Management:

Each node maintains quality of links to neighbors. Link quality is used as link cost in routing.

Frequency Agility:

Nodes experience interference report to Channel Manager (eg: trust centre) which then select another channel.

→ Multicast, Many to one Routing

Asymmetric link: Each node has different transmit Power and sensitivity, Paths may

asymmetric.

Fragmentation and Reassembly:

Zigbee Overview:

Industrial Monitoring and control Applications requiring requiring small amounts of data, turned off most of the time (<1% duty cycle)

Ex: Wireless light Switches, meter reading, Patient monitoring

→ Ultra-low power, low-data rate, multi-year battery life.

→ Power Management to ensure low power Consumption.

→ Less Complex: 32kB protocol stack Vs 250kB. for - Blue tooth.

→ Range: 1 to 100m, upto 65000 nodes.

→ Tri band:-

16 Channels at 250 kbps in 2.4 GHz. ISM.

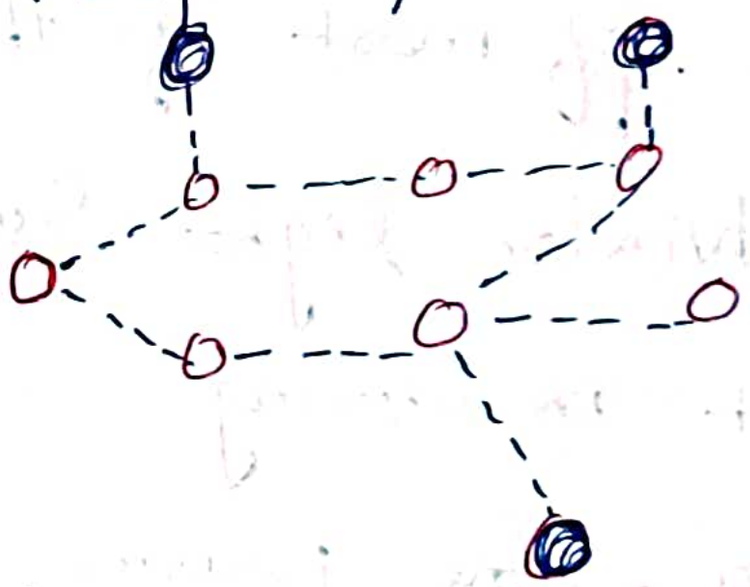
10 Channels at 40 kb/s in 915 MHz ISM band.

→ Multi hop Routing — Message to non adjacent nodes.

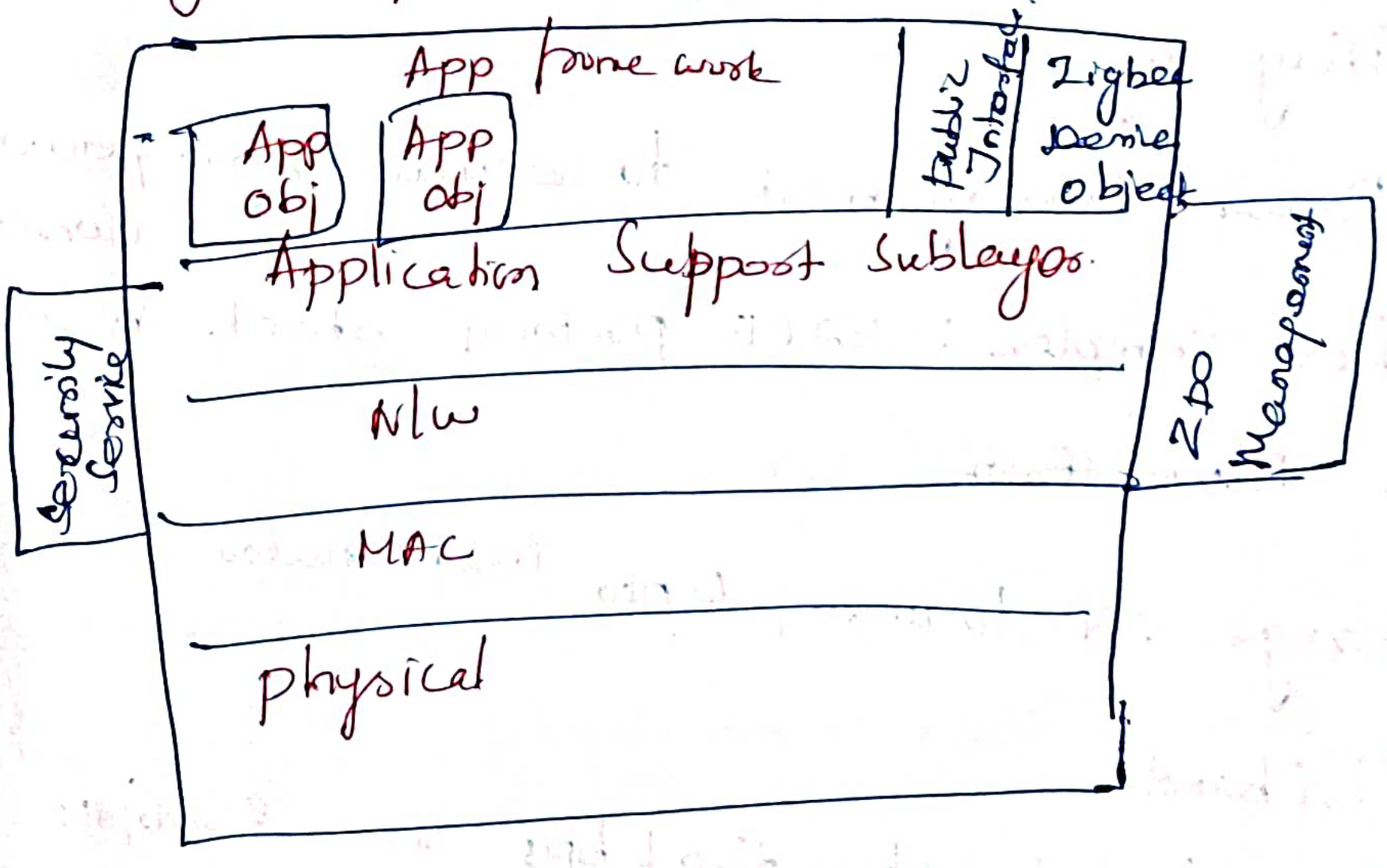
Adhoc Topology → No fixed topology! Nodes discover each other.

Mesh Routing → End nodes helps route messages for others.

Mesh Topology → Loops possible.



Zigbee protocol Architecture:



Zigbee Application layer:

Appln layer consists of Appln Objects (aka end points) and Zigbee device objects (ZDOs)

256 End point Addresses

→ 240 Appln objects : Addresses EP1 through EP240.

→ ZDO is EP0

→ End points 241 - 254 are reserved.

→ EP 255 is broadcast.

→ Each End point has one Appln profile.

Ex: light on/off

→ Zigbee forum has defined a number of profile Users can develop other profile.

→ Attributes : Each profile requires a number of data items. Each data item is called as 'attribute' and is assigned an 16 bit 'attribute ID' by zigbee forum.

Clusters: A collection of attributes and commands on them. Each cluster is represented by a 16 bit ID. Commands could be read/write requests or read/write response.

Cluster Library: A collection of cluster. Zigbee forum has defined a number of cluster libraries Ex: General cluster library contains ON/OFF level control, alarms, etc.

Binding: Process of establishing a logical relationship (parent, child...)

ZDO

→ Uses device and service discovery commands to discover details about other devices.

→ Uses binding commands to bind and unbind end points

→ Uses N/w Management Commands for N/w discover, route discovery, link quality indication join/leave requests.

Zigbee types

Co-Ordinator : Select channels, starts the Nw. assign short addresses to other nodes, transfer Packets to/from other nodes.

Router.

Transfer Packets to/from other nodes.

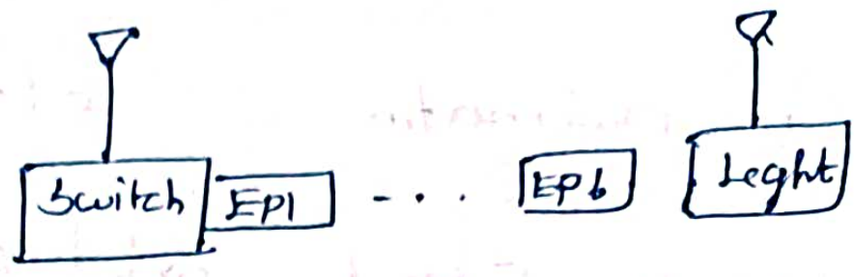
Full function device: Capable of being Co-ordinator or router.

Reduced - Function Device → Not capable of being a Co-ordinator (or) a router.

⟹ Leaf node.

Zigbee Gateway → Connects to other networks eg: WiFi.

Zigbee Protocol Architecture:



Appln objects

Ex: Remote Control Applns.

End - Node : End device.

Each node can have up to 250 Appln objects.

Zigbee Device Objects (ZDO)

Control and Management of application Objects. Initialize Co-ordinator, Security service device and service discovery.

Appln Support layer (APS): ~~service~~ Serves Appln Objects.

Network layer:

Route Discovery, neighbour discovery, ZDO Management, Security service.

Bluetooth

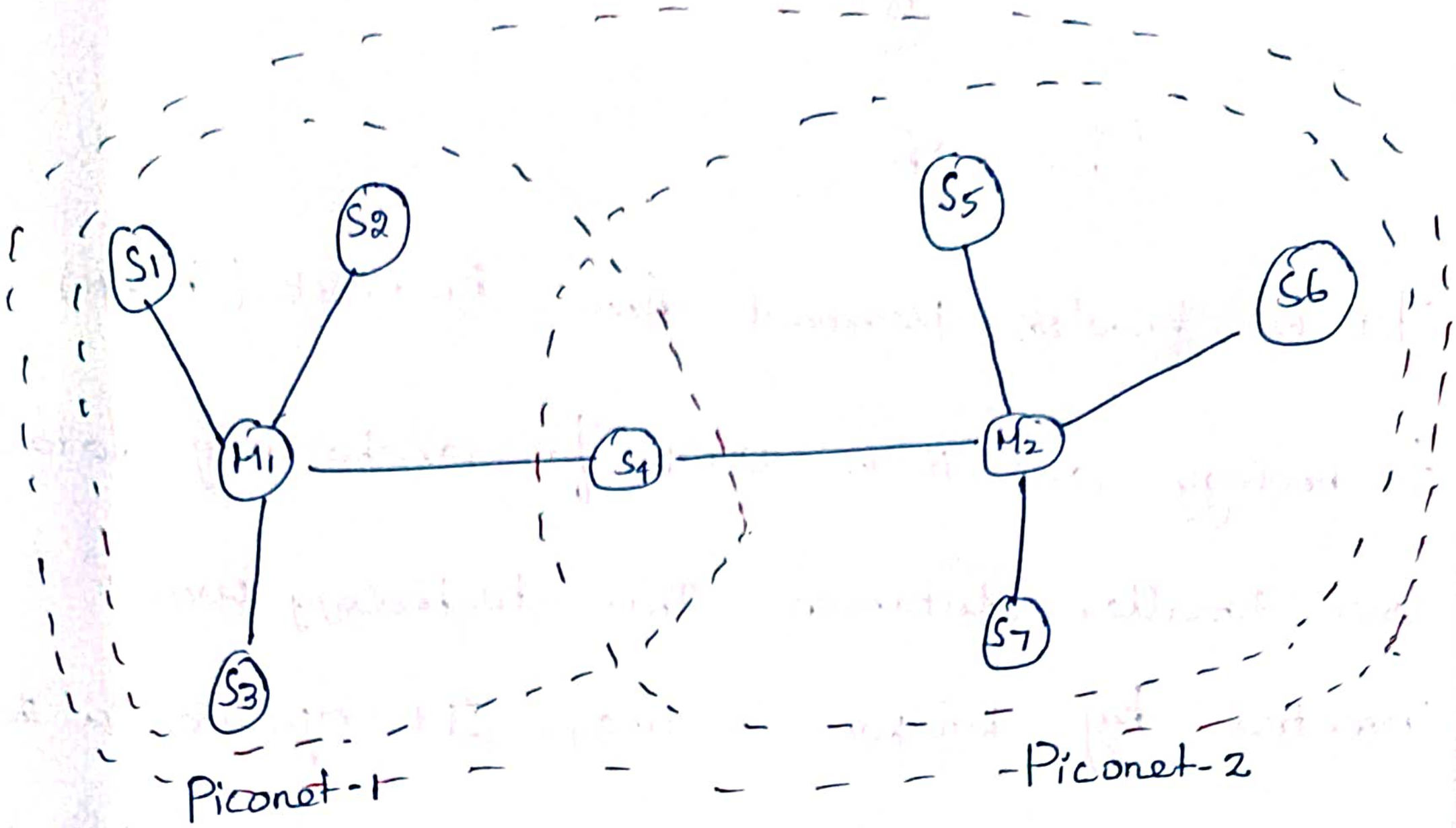
It is Wireless Personal Area Network (WPAN) technology and it is used for exchanging data over smaller distances. This technology was invented by Ericson in 1994. It operates in the Unlicensed, Industrial, Scientific and medical (ISM) band at 2.4 GHz to 2.485 GHz. Maximum devices that can be connected at the same time are 7.

→ Bluetooth ranges upto 10 meters.

→ It provides data rates upto 1Mbps or 3Mbps. depending upon the version.

→ The spreading technique which it uses is FHSS.

→ A Bluetooth Network is called piconet. and a collection of interconnected piconets is call Scatternet.



Scatternet

The features of Piconets are follows.

→ Within a piconet, the timing of various devices and the frequency hopping sequence of individual device is determined by the clock and unique 48-bit Address of Master.

→ Each device can communicate simultaneously with upto seven other devices within a single piconet.

→ each device can communicate with several piconets simultaneously.

→ piconets are established dynamically and automatically as bluetooth enabled devices enter and leave piconets.

There are no direct connection between the slaves and all connections are essentially master-to-slave or slave to Master.

→ slaves are allowed to transmit once there have been polled by the master.

→ Transmission starts in the slave-to-master time slot simultaneously immediately following a polling Packet from the master.

→ A device can be member of two or more piconets, jumping from one piconet to another by adjusting the transmission regime-timing and frequency hopping sequence dictated by the master device of the second piconet.

→ It can be slave in one piconet and Master in another. It however cannot be a master in more than once piconet.

→ Devices resident in adjacent piconets provide a bridge to support inner piconet connections, allowing assemblies of linked piconets to form

physically extensible communication infrastructure known as scatternet

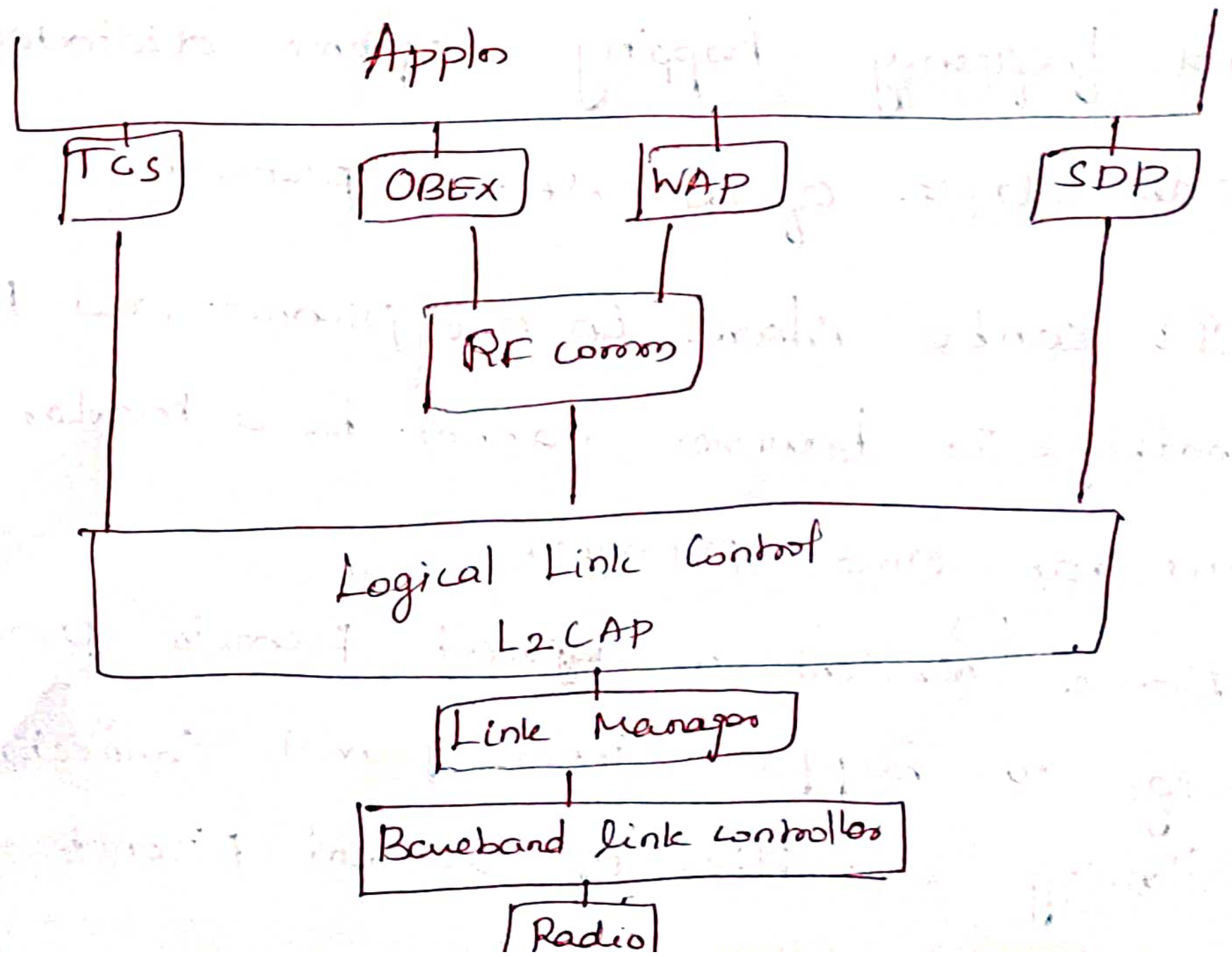
Spectrum

Bluetooth technology operates in the unlicensed and industrial, scientific and medical band at 2.4 to 2.485 GHz

Data rate

Bluetooth supports 1 Mbps data rate for version 1.2 and 3 Mbps data rate for version 2.0 Combined with error data rate.

Bluetooth protocol stack.



① Radio (RF) layer

It performs modulation/ demodulation of the data into RF signals

It defines the physical characteristics of bluetooth transceiver.

It defines two types of physical link.

Connection - less and Connection oriented.

② Baseband Link layer

It performs the connection establishment within a piconet.

③ Link Manager protocol layer

It performs the Management of the already established links. It also includes authentication and encryption processes.

④ Logical link control and Adaption protocol layer

It is also known as the heart of the bluetooth protocol stack. It allows the communication b/w upper and lower layers of the bluetooth protocol stack. It packages the data packets received from upper layer into the form expected by lower layers. It also performs the segmentation and Multiplexing.

SDP layer

It is short for Servke Discorery Protocol

It allows to discover the servies arcailable on another bluehooth enabled device.

RF Comm layer:

It is short for Radio Frontend Component. It provides serial interface with WAP and OBEX.

OBEX

It is short for Object Exchange. It is a common protocol to exchange objects blw 2 devices.

WAP (Wireless Appln Acces protocol)

↳ It is used for Internet Acces.

TCS (Telephony Contol protocol)

↳ provides telephony service.

Appln layer

↳ It enables the user to interact with the appln

Advantage	DIS Advantage
Low Cost, Easy to use	It can be hacked, less secure
Penetrate through walls.	Slow data transfer range 3Mbps
It is used for voice and data transfer.	Small range: 10 meters.

UNIT: II - MAC & Routing Protocols

MAC protocols - fundamentals - low duty cycle protocols and wakeup concepts, Contention and schedule based protocols SMAC, BMAC, TRAMA, Routing protocols Requirements, classifications - SPIN, Directed Diffusion, COUGAR, ACQUIRE, LEACH, PEGASIS.

Introduction:

Medium Access Control (MAC) protocols is the first protocol layer above the physical layer and consequently MAC protocols are heavily influenced by its properties.

The fundamental task of any MAC protocol is to regulate the access of a number of nodes to a shared medium in such a way that certain application dependent

Performance requirements are satisfied. Some of the traditional performance criteria are delay, throughput, and fairness. Whereas in WSNs the issue of energy conservation becomes important.

→ Within the OSI reference Model, the MAC is considered as a part of the data link layer (DLL), but there is a clear division of work between MAC and the remaining part of the DLL.

→ The MAC protocol determines for a node the points in time when it accesses the medium to try to transmit a data / control or management packet to another node, or to a set of nodes (multicast, broadcast)

Addresses / Names are always tied to a representation which has a certain length when considered as a string of bits.

An opposed to other type of Networks, representation size is the critical issue in wireless sensor Network, since the addresses are present in almost any packet. However coordination among nodes is needed to assign reasonably short addresses. A second key aspect is Content based addressing, where not nodes or network interfaces but data is addressed.

Content based addressing can be integrated with data Centric routing and is also a key enabler of in Network processing.

MAC protocols for Wireless Sensor Networks

Fundamentals of (Wireless) MAC protocols:

MAC protocol used in Wireless sensor Nw inherit many of the problems and approaches already existing in the general field.

Requirements of and design of for Wireless MAC protocols:

→ The Main issues in designing MAC protocols for sensor Network * Bandwidth * efficiency.

Bandwidth must be Utilized in efficient manner.

minimal control Overhead

BW = ratio of BW used for actual data transmission to the total available B.W.

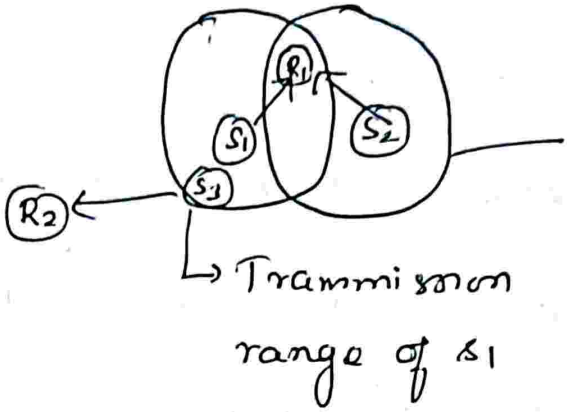
Quality of Service support:

* Essential for supporting time Critical traffic sessions.
* They have resource reservation mechanism that takes into considerations the nature of wireless channel and the mobility of nodes.

Synchronization.

* MAC protocol must consider synchronization between nodes in the network.
* Synchronization is very important for BW reservation by Nodes
* Exchange of control packet may be required for achieving time synchronization among nodes.

Hidden and exposed terminal problems.



⟶ Packet transmission

- - - - - → Transmission data that is not permitted.

The hidden terminal problem refers to the collision of packet at a receiving node due to the simultaneous transmission of those node that are not within the direct transmission range of the sender but are within the transmission range of the receiver.

* Collision occurs when both nodes transmit packet at the same time without knowing about the transmission of each other.

S_1 and S_2 are hidden from each other and they transmit simultaneously to R_1 which leads to collision.

→ The exposed terminal problem refers to the inability of a node, which is blocked due to transmission by a nearby transmission node to transmit to another node.

→ If S_1 is already transmitting to R_1, then S_3 cannot interfere with on going transmission and it cannot transmit to R_2.

→ The Hidden and exposed terminal problem reduce the throughput of a network when traffic load is High.

Error - prone shared broadcast channel.

* When a node is receiving data, no other node in its neighbourood should transmit.
* A node should get access to the shared medium only when its transmission

do not affect any going session.

MAC protocol should grant channel access to nodes in such a manner that collisions are minimized.

- Protocol should ensure fair B.W allocation.

Distributed Nature / lack of Central Co-ordination.

Do not have Centralized Co-ordinators.

Nodes must be scheduled in a distributed fashion for gaining access to the channel

MAC protocol must make sure that additional overhead in terms of BW Consumption incurred due to this control information is not very high.

Mobility of nodes:

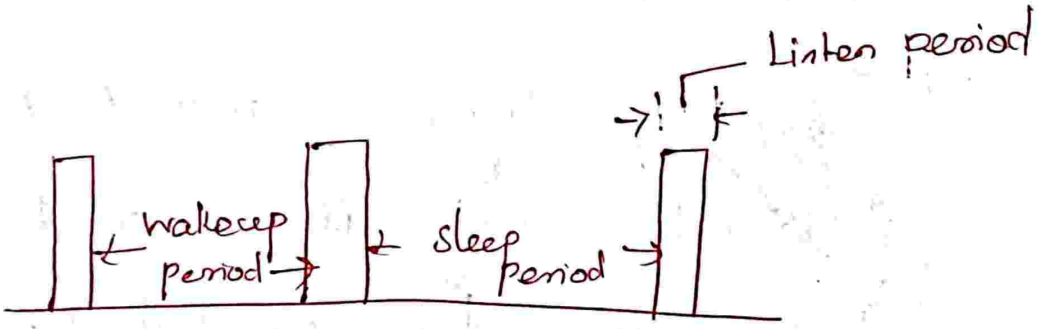
Nodes are mobile most of the time.

The protocol design must take this mobility factor into consideration. So that

the performance of the system is not affected due to node mobility.

Low duty cycle protocols and Wakeup concepts:

Low duty cycle protocols try to avoid spending (much) time in the idle state and to reduce the communication activities of a sensor node to a minimum. In an Ideal case, the sleep state is left



Periodic Wakeup scheme.

Only when a node is about to transmit or receive packets. A concept for achieving this, the Wakeup radio, is

discussed. However, such a system has not been built yet, and thus, there is significant interest to find alternative approaches.

In general protocol, a periodic wakeup scheme is used. Such scheme exist in different flavours. One is the cycled receiver Approach. In this Approach, nodes spend most of their time in the sleep mode and wakeup periodically to receive packets from other nodes.

specifically, a node A listens onto the channel during its listen period and goes back into sleep mode. When no other node take opportunity to direct a packet A. A potential transmitter B must acquire knowledge about A's listen periods to send its packets at the right time.

This task corresponds to a rendezvous.

⇒ Node A only receive the packets during its listen period. If node A itself wants to transmit packets, it must acquire the target listen period. A whole cycle consisting of sleep period and listen period is also called a wakeup period.

→ The ratio of the listen period length to the wakeup period length is also called a node's duty cycle

⇒ By choosing a small duty cycle, the transceiver is in sleep mode most of the time, avoiding idle listening and conserving energy.

By choosing a small duty cycle, the traffic directed from neighbouring nodes to a given node concentrates on a small time window (the listen period) and in heavy load situations significant competition can occur.

→ choosing a long sleep period induces a significant per-hop latency. Since a prospective transmitter node has to wait an average of half a sleep period before the receiver can accept packets.

→ In the multihop case, the per hop latencies add up and create significant end to end latencies.

Sleep phases should not be too short lest the start up costs outweigh the benefits. In other protocols like, for example S-MAC, there is also a periodic wakeup but nodes can both transmit and receive during their wakeup phases.

→ When nodes have their wakeup phases at the same time, there is no necessity for a node wanting to transmit a packe to be awake outside these phases to rendezous its receiver.

—

S - MAC

The S - MAC (Sensor MAC) protocol provides mechanisms to circumvent idle listening, collisions, and overhearing. As opposed to STEM, it does not require two different channels. S-MAC adopts a periodic wakeup scheme. that is, each node alternate between a fixed length listen period and a fixed length sleep period according to its schedule. Compare as opposed to STEM, the listen period of S-MAC can be used to receive and transmit packets.

S-MAC attempts to co-ordinate the schedule of neighbouring nodes such that their listen period start at the same time. A node x's listen period is subdivided into three different phases.

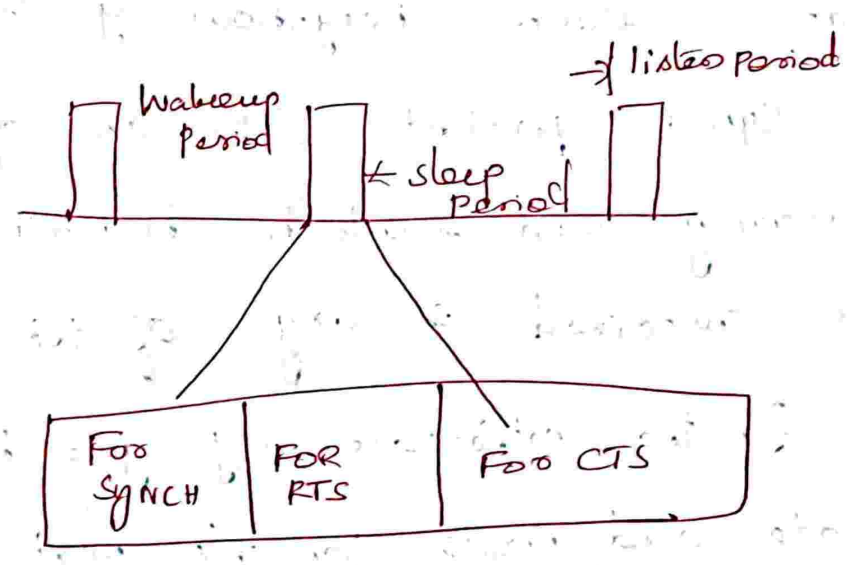
In the first phase (SYNCH), node 'x' accepts SYNCH packets from its neighbours. In there packets, the neighbour describe their own schedule and x stores their schedule. Node x's SYNCH phase is subdivided into time slots and x's neighbors contest according to a CSMA scheme with additional backoff that is each neighbour 'y' wishing to transmit a SYNCH packet picks one of the time slots randomly and starts to transmit if no signal was received in any of the previous slots.

→ In the otherwise, y goes back into sleep mode and waits for 'x's next wake up.

→ In the other direction, since x known as neighbour y's schedule, x can wake at appropriate times and send its own SYNCH Packet to y (broadcast mode)

→ If it is not required that 'x' broadcast its schedule in every of y's wakeup periods

→ However, for reasons of time synchronization and to allow new nodes to learn their local Network topology. 'x' should send SYNCH packet periodically. Thus according period is called Synchronization Period.



S-MAC Principle.

In the second phase (RTS), 'x' listens for RTS packet from neighbouring nodes. In S-MAC, the RTS/CTS handshake described is used to reduce collisions of data packets due to hidden terminal situations. Again interested neighbour contend in this phase according to a CSMA scheme will additional back-off.

→ In the third phase (CTS), node x transmits a CTS packet if an RTS packet was received in the previous phase. After this, the packet exchange continues, extending into x's nominal sleep time.

→ In General, when competing for the medium the nodes use the RTS/CTS Handshake including the virtual carrier sense mechanism, whereby a node maintains a NAV variable.

→ The NAV mechanism can be readily used to switch off the node during ongoing transmissions to avoid overlearing. When transmitting is a broadcast mode (for example SYNCH packets), the RTS and CTS packets are dropped and the node use CSMA with back off.

→ If we can arrange that the schedules of node x and its neighbours are synchronized node x and all its neighbour wake up at the same time and 'x' can reach all of them with a single SYNCH packet.

→ S-MAC protocol allows neighboring nodes to agree on the same schedule and to create virtual clusters-

→ The clustering structure refers solely to the exchange of schedules.

the transfer of data packet is not influenced by virtual clustering.

The S-MAC protocol proceeds as follows to form the virtual clusters.

A node x, newly switched 'on', listens for a time of at least the synchronization period. If 'x' receives any SYNCH Packet from a neighbour, it adopts the announced schedule and broadcast it in One of the neighbour's next listen periods.

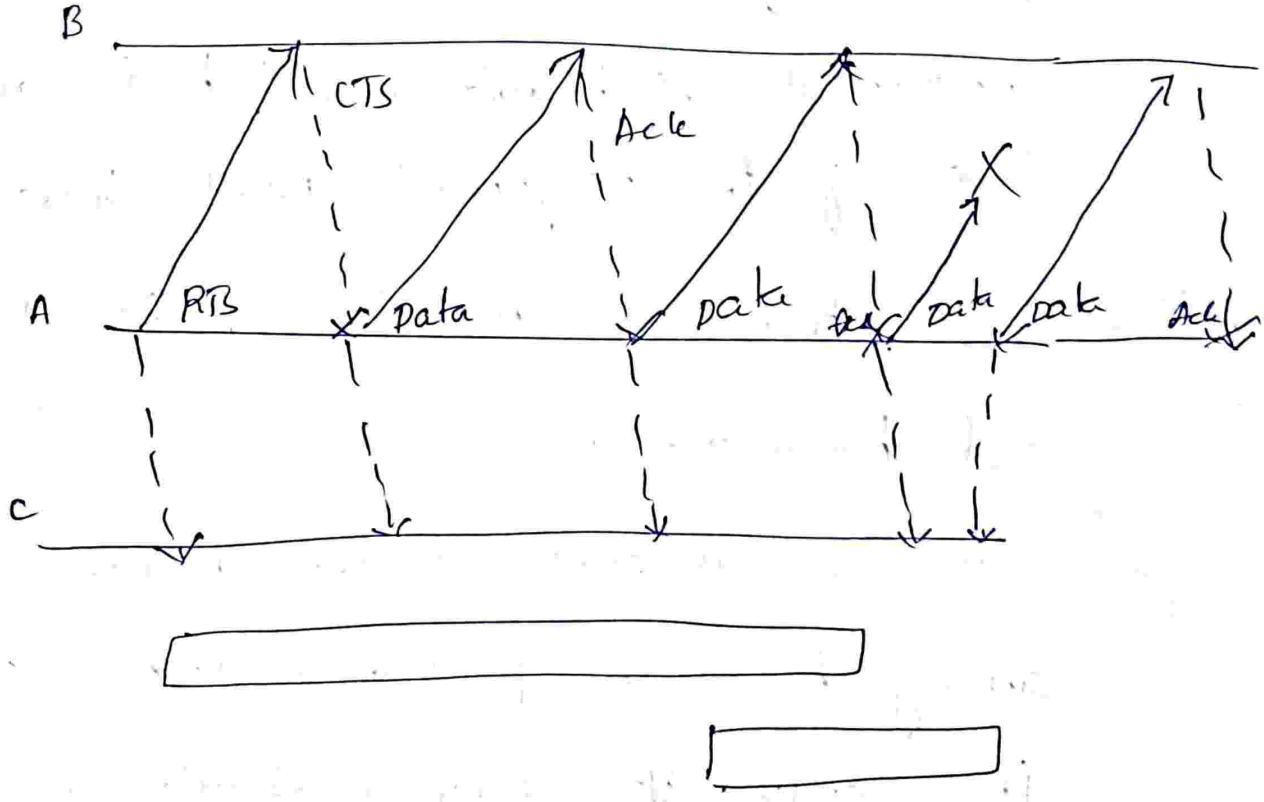
→ In other case, node 'x' picks a schedule and broadcast it.

→ If 'x' receives another node's schedule during broadcast packet contention Period, it drops its own schedule and follows the other one.

→ It might also happen that a node 'x'

receives a different schedule after it already has chosen one.

→ The periodic wakeup scheme adopted by S-MAC allows node to spend much time in the sleep mode, but there is also a price to pay in terms of latency.



S-MAC also adopts a message passing approach, where a message is larger data item meaningful to the application. In network processing usually requires the aggregating node to receive a message completely.

S-MAC includes a fragmentation scheme working as follows: A series of fragmentation is transmitted with only one RTS/CTS Exchange between the transmitting node A and receiving node 'B'. After each fragment, B has to answer with an acknowledgement packet. All the Packets (data, Ack, RTS, CTS) have a duration field and a neighbouring node 'i' is required to set its NAV field accordingly. In S-MAC the duration field of all packets carries the remaining length of the whole transaction.

including - all

B - MAC

B - MAC : Berkeley Media Access Control for Low Power wireless Sensor Networks.

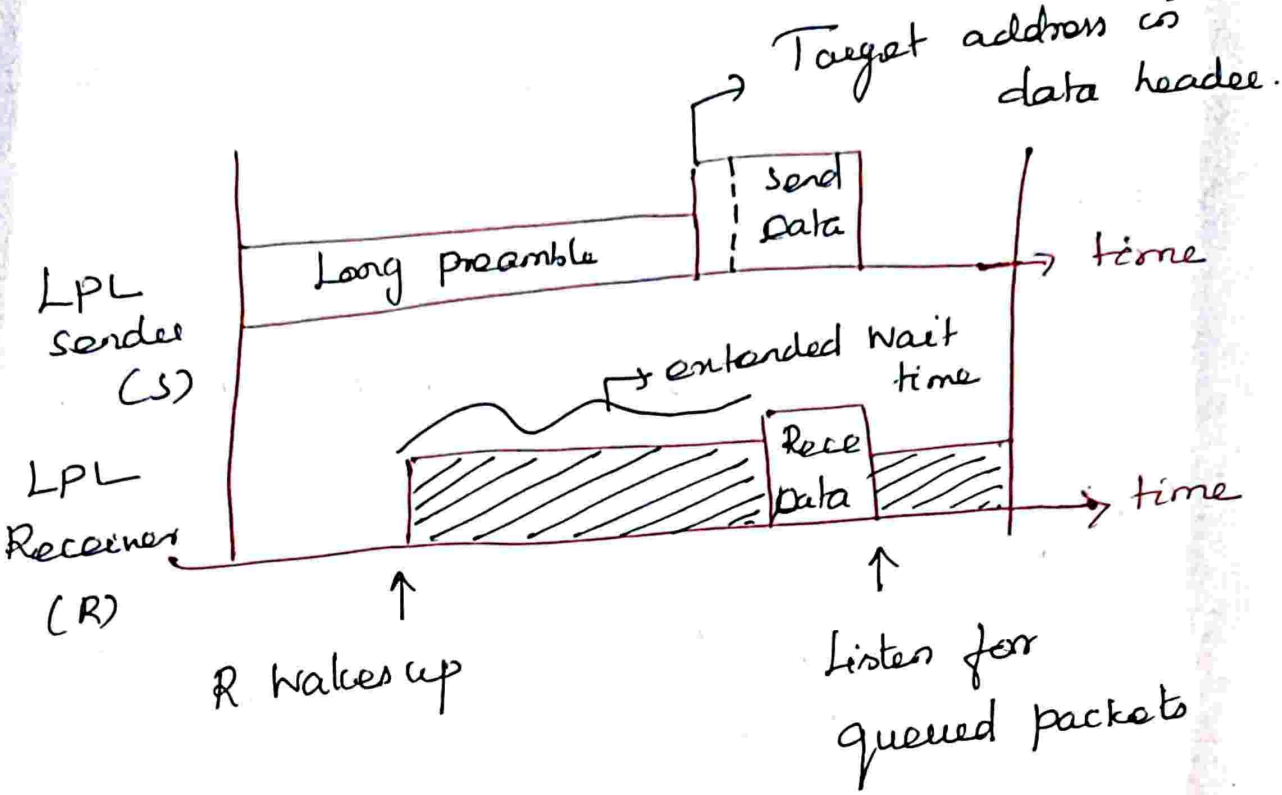
→ Versatile low power Media Access for Wireless Sensor Nlw. Topology stwe.

Key Ideas:

→ Low Power Listening (LPL) for Low Power Communication.

→ Clear channel assessment (CCA) and Packet backoff for channel arbitration.

→ Link layer Acknowledge for reliability



Low power listening:

→ Extended preamble and preamble sampling

→ each time a node wakes up, it turns on its radio to check for activity. If a preamble signal is detected, it will stay awake for the time required to receive the incoming packet.

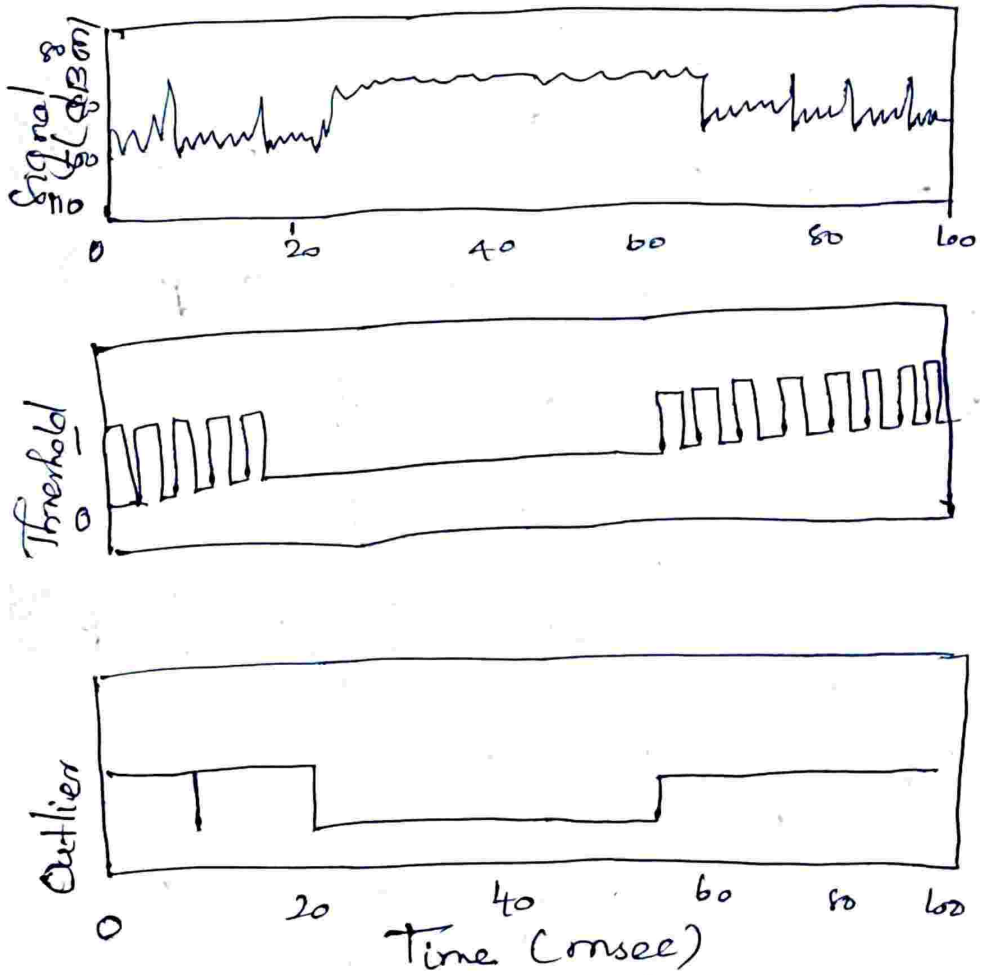
→ The preamble length is matched to the interval that the channel is check.

Ex: the preamble must be atleast 100 msec If the channel is check every 100 msec.

Clear channel Assessment (CCA)

→ filtering to remove issue.

→ Thresholding to estimate Channel status.



TRAMA

Traffic Adaptive Medium Access protocol

→ The traffic Adaptive medium Access protocol presented by Rajendran. Creates schedules allowing Nodes to access a single channel in a collision-free manner.

→ The schedules are constructed in a distributed manner and on an on-demand basis.

→ The protocol assumes that all nodes are time synchronized and divides time into random access Periods. and scheduled-access Periods

→ Random Access Period followed by a scheduled access period is called a cycle

→ The nodes broadcast their neighbourhood information and, by capturing the respective packets from their neighbour, can learn about their own two hop - neighborhood.

→ Furthermore, they broadcast their schedule information. that is they periodically provide their neighbor with an updated list of receivers for the packets currently in a nodes queue.

→ On the basis of this information, the nodes execute a distributing scheduling algorithm to determine for each time slot of the scheduled access period the transmitting and receiving nodes and the nodes that can go into sleep mode.

→ The protocol itself consists of three different components

→ The neighborhood protocol

→ The schedule exchange protocol

→ adaptive election algorithm.

→ The Neighborhood protocol is executed solely in the random Access phase, which is subdivided in to small time slots. A node picks randomly a number of time slots and transmits small control packets in there without doing any carrier sensing.

→ There packet indicates the nodes identification are included that belong to new neighbors information. that is only those neighbor identifications are included that belong to new neighbor or neighbor that were missing during the last cycle.

When node does not transmit, it listens to pick up neighbor control packets.

The length of the random access phase should be chosen such that a node receive its neighbor packets with sufficiently high probability to ensure consistent topology information...

→ It depends thus on the node degree, All nodes' transceivers must be active during the random Access period.

→ By the scheduling exchange protocol. a node transmits its current transmission schedule and also picks up its neighbor Schedule.

→ This information is used to actually allocate slots to transmitter & receivers.

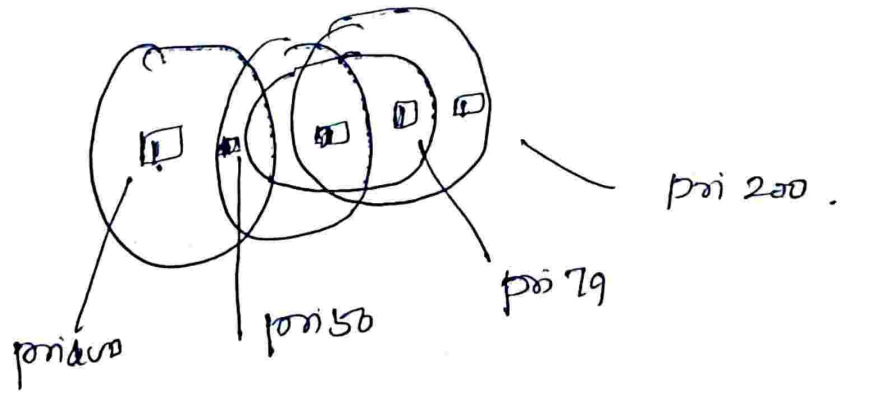
(15).

→ How does the node know which color

All nodes possess a global hash function h. and a node with identification 'x' Computes for time slot occuring at 't' the following priority value 'p'

$$p = h(x \oplus t)$$

Where $x \oplus t$ → concatenation of x's node identification with current time t.



The TRAMA protocol needs significants Computation and memory in dense sensor network. Since the two-hop neighborhood of a node tends to be large in this case

→ Therefore TRAMA is a feasible soln only if the sensor nodes have sufficient resources.

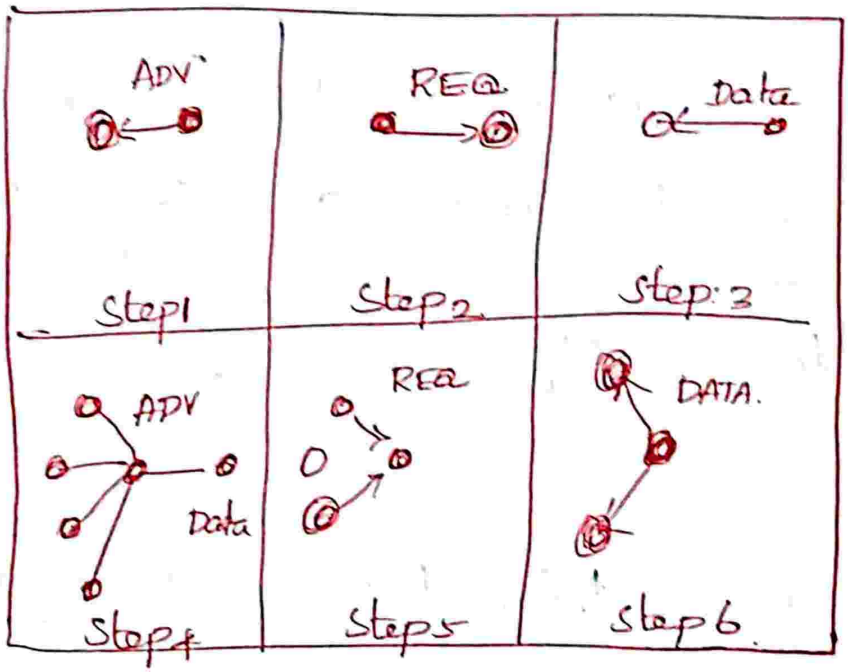
SPIN: (Sensor Protocols for Information via Negotiation)

Sensor Protocols for Information via Negotiation use negotiation and resource adaption to address the disadvantage of basic flooding.

SPIN uses data centric routing, nodes are advertising their data and they will send the data after receiving a replay from interested nodes.

SPIN uses three types of messages ADV, REQ, and DATA. The sensor node that has collected some data sends and ADV message containing meta data describing the actual data. If some of nodes neighbours

is interested in the data, the neighbour sends a REQ message back. After receiving the REQ message, the sensor node sends the actual data. The neighbour also sends ADV message forward to its neighbors. Thus data is disseminated through the network.



→ In fig. node A advertises its data using ADV message. its neighbor node B replies with a REQ message and

thus node A sends actual data to the B.

Node B also forwards ADV message to its neighbor. Improved version of SPIN, SPIN-2 uses an energy or resource threshold to reduce participation of nodes.

Thus, only those nodes that have sufficient amount of resource participate in ADV-REQ-DATA exchange.

SPIN is more efficient than flooding since the negotiation reduces the explosion and overlap.

Resource Adaptation in SPIN-2 prolongs the lifetime of the network. Sensor node with low resources do not have to

Participate in ADV – REQ – DATA exchange and as a result they can collect data for a longer time.

Low energy Adaptive clustering Hierarchy (LEACH)

→ Randomly select sensor node as Cluster Heads, so that High energy dissipation in Communicating with the sink is spread to all sensor nodes in the sensor n/w.

Set up phase.

- each sensor node chooses a random number between '0' and '1'
- If the random number is less than the threshold T(n), the sensor node is cluster head.

$$T(n) = \begin{cases} \dfrac{p}{1 - p[r \bmod (1/p)]} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases}$$

$p \rightarrow$ the desired percentage to become cluster Head.

r, the current round.

$G \rightarrow$ the set of nodes that have not been selected as a cluster-head in the last $1/p$ rounds.

Set up phase:

* The cluster head advertise to all sensor nodes in the network.
* The sensor node inform the appropriate cluster heads that they will be a member of the cluster (based on signal strength)
* Afterwards, the cluster-heads assign the time at which the sensor nodes

can send data to the cluster heads based on the TDMA Approach.

Steady phase:

The sensor nodes can begin sensing and transmitting data to the cluster-heads The cluster heads also aggregate data from the nodes in their cluster before sending there data to the sink.

After a certain period of time spent on the steady phase, the n/w.

→ goes into the setup phase again

→ enters into another round of selecting the cluster heads.

Note: LEACH utilizes randomized rotation of Local cluster head (CHs) to evenly distribute the energy load among sensors.

[Leach makes use of a TDMA/CDMA scheme to reduce inter and intra cluster collisions.

PEGIASIS

Data Gathering

The objective of the data gathering problem is to transmit the sensed data from each sensor node to Base station.

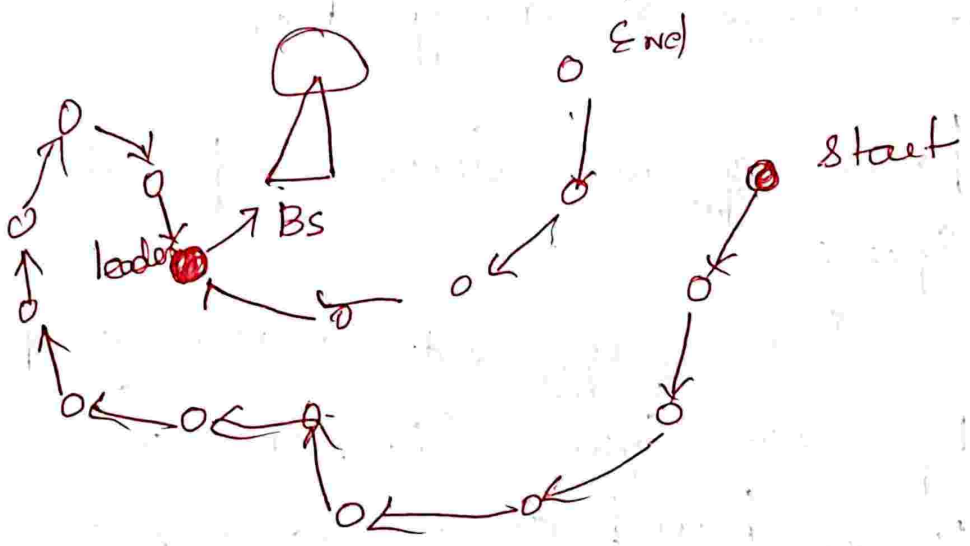
One round is defined as the Base station collecting data from all the sensor nodes once. This scheme performs poorly with respect to the energy x delay Metric.

Power efficient Gathering for sensor Information System.

→ Power efficient Gathering for sensor Information System (PEGIASIS) is a data gathering protocol based on the assumption that all sensor nodes know the location of every other node that is, the topology information is available to all nodes.

The Goal of PEGASIS

* Minimize the distance over which each node transmits.
* Minimize the broadcasting overhead.
* Minimize the number of message that needs to be sent to the base station.
* Distribute the energy consumption equally across all nodes.



Binary scheme.

This is also a chain based scheme like PEGASIS, which classifies nodes in to different levels.

* All nodes which receive message at one level move to the next.

* The number of nodes is halved from one level to the next.

* The number of nodes is halved from one level to the next. For instance, Consider a Network with eight nodes. labeled S_0 to S_7 This scheme is possible when nodes communicate using CDMA, so that transmissions of each level can take place simultaneously.

Step:1 $S_0 \rightarrow S_1$ $S_2 \rightarrow S_3$ $S_4 \rightarrow S_5$ $S_6 \rightarrow S_7$

Step:2 $S_1 \longrightarrow S_3$ $S_5 \longrightarrow S_7$

Step:3 $S_3 \longrightarrow S_7$

Step:4 $S_7 \longrightarrow$ sees. Base station.

Chain based three level scheme:

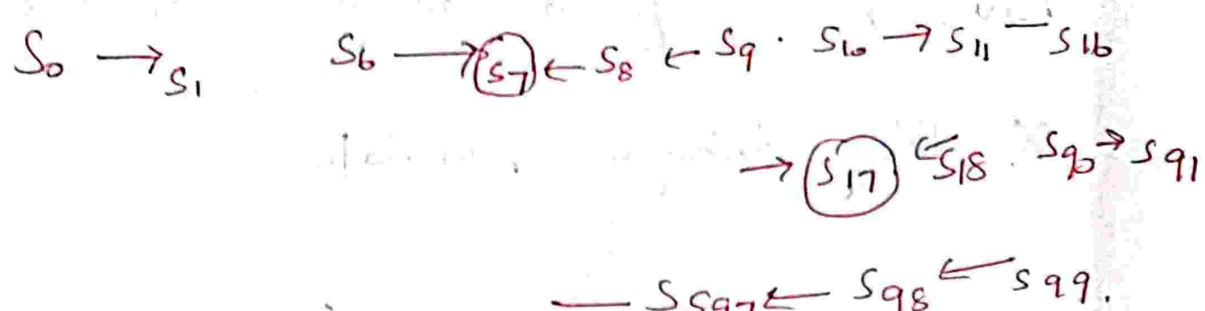
For non CDMA sensor node, a binary sch eme is not applicable. The chain based three lev l scheme addresses this sti situation, where again a chain is constructed as in PEGASIS.

The chain is divided into a number of groups to space out simultaneous transmission in order to minimize interference.

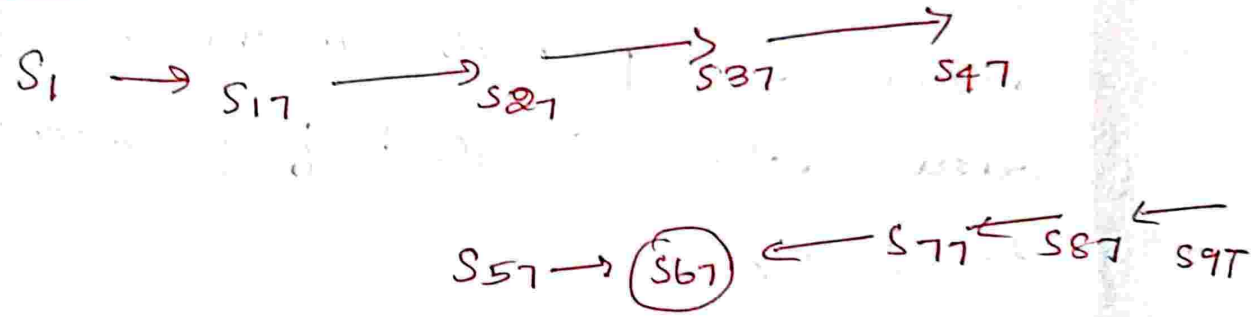
One node out of each group aggregates data from all group members and rises to the next level. The index of this leader node is decided a priori.

In the second level, all nodes are divided into two groups, and the third level consists of a message exchange between one node from each group of the second level.

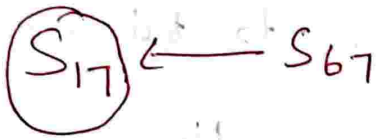
Step: 1



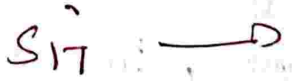
Step: 2



Step: 3



Step: 4



○ → Group leader.

Finally, the leader transmits a single msg to the Base station.

→ The N/w has 100 nodes. and the group size is 10 for the first level.

and five for the second level. Three level have been found to give the optimal energy X delay through simulations.

Data Centric Routing:

Data centric protocols differs from traditional address centric protocols in the manner that the data is sent from source sensor one to the sink.

In Address centric protocols, each source sensor that has the appropriate data responds by sending its data to the sink independently of all other sensors. However, in data centric protocol, when the same sensors send their data to the sink, intermediate sensor can perform some form of aggregation on the data originating from multiple source sensors and send the aggregated data toward the sink.

⟹ This process can result in energy saving because of less transmission required to send the data from the sources to the sink.

COUGAR

A data centric protocol that views the Network as a huge distributed database s/m. The main idea is to use declarative queries in order to abstract query processing from the Network layer functions such as selection of relavent sensor etc. and utilize in-N/w data aggregation to save energy. The Abstraction is supported through a new query layer between the N/w and Appln layers.

COUGAR proposes architecture for the sensor database s/m where sensor nodes

Select a leader node to perform aggregation and transmit the data to the Gateway.

ACQUIRE (Active Query Forwarding in Sensor Networks)

ACQUIRE is another data centric querying Mechanism used for querying named data. It provides superior query optimization to answer specific types of queries, called One-Shot Complex queries for replicated data. ACQUIRE query (ie Interest for named data) consists of several sub queries for which several simple response are provided by several relevant sensor. Each sub-query is answered based on the currently stored data at its relevant sensor.

ACQUIRE allows a sensor to inject an active query in a network following either a random or a specified trajectory. Until the query get answered by some

Sensors on the path using a localized update mechanism. Unlike other query techniques ACQUIRE allows the queries to inject a complex query into the network to be forwarded stepwise through a sequence of sensors.

UNIT: III

6LOWPAN

Introduction:

The IPV6 Low power Wireless personal Area N/w standards allows IPV6 to be used over 802.15.4 Wireless Networks.

- 6Lowpan is often used for Wireless sensor Network.

→ There are huge range of applications which could benefit from a Wireless Embedded Internet Approach.

→ Today there applications are implemented using a wide range of proprietary technologies which are difficult to integrate into larger n/w and with Internet based services.

→ The benefits using Internet protocols in there applications, and then integrating them with the Internet of Things.

- IP based device can be connected easily to other IP Networks without the need for translation gateways or proxies.

* IP Networks allow the use of existing network infrastructure.

→ Direct Communication with traditional IP NWs require many Internet protocol, often requiring an operating system to deal with the complexity and maintainability. IP are demanding for embedded devices for the following reasons.

Security: IPV6 Includes optional optional support for IP security authentication and encryption and web services typically make use of secure sockets or transport layer security mechanisms. These technique may be too complex, especially for simple embedded devices.

Web services: Internet services today rely on web services, mainly using the transmission control protocol (TCP), HTTP, SOAP and XML with complex transaction patterns.

Management: Management with the simple network Management protocol (SNMP) and web services is often inefficient and complex.

frame size: Current IP require links with sufficient frame length of 1280 bytes

Applicatoon of 6LOWPAN:

* Embeddded devices need to Communicate with internet based Services.
* Low power heto heterogeneous networks need to be tied togather.
* The network needs to be open, reusable and evolable for new uses and servies and scalability is needed across large network infrastructure with mobility.

Example

* Home and building Automation
* Healthcare Automation and logistics
* Personal health and fitness
* Industrial Automation
* Smart metering and smart grid infrastructure.
* Asset Management and logistics.
* Vehicular Automation.

6LoWPAN → facility Management. Which is the Management of large facilities using a Combination of building automation, asset Management and other embedded systems.

* Door access control
* Smart metering.
* Building Automation
* Tracking
* Energy reduction
* Maintenance.

6 LowPan

Protocol stack:

A simple IPV6 protocol stack with 6Lowpan is almost identical to a Normal IP stack with following difference.

* 6Lowpan stack implementation is embedded device often implement the Lowpan Adaption layer together with IPV6. thus they can alternatively be shown together as part of the Nw layer.
* The Most common transport protocol used with 6Lowpan is the User datagram protocol which can be compressed using the Lowpan format.

→ Tcp is not used in 6Lowpan.

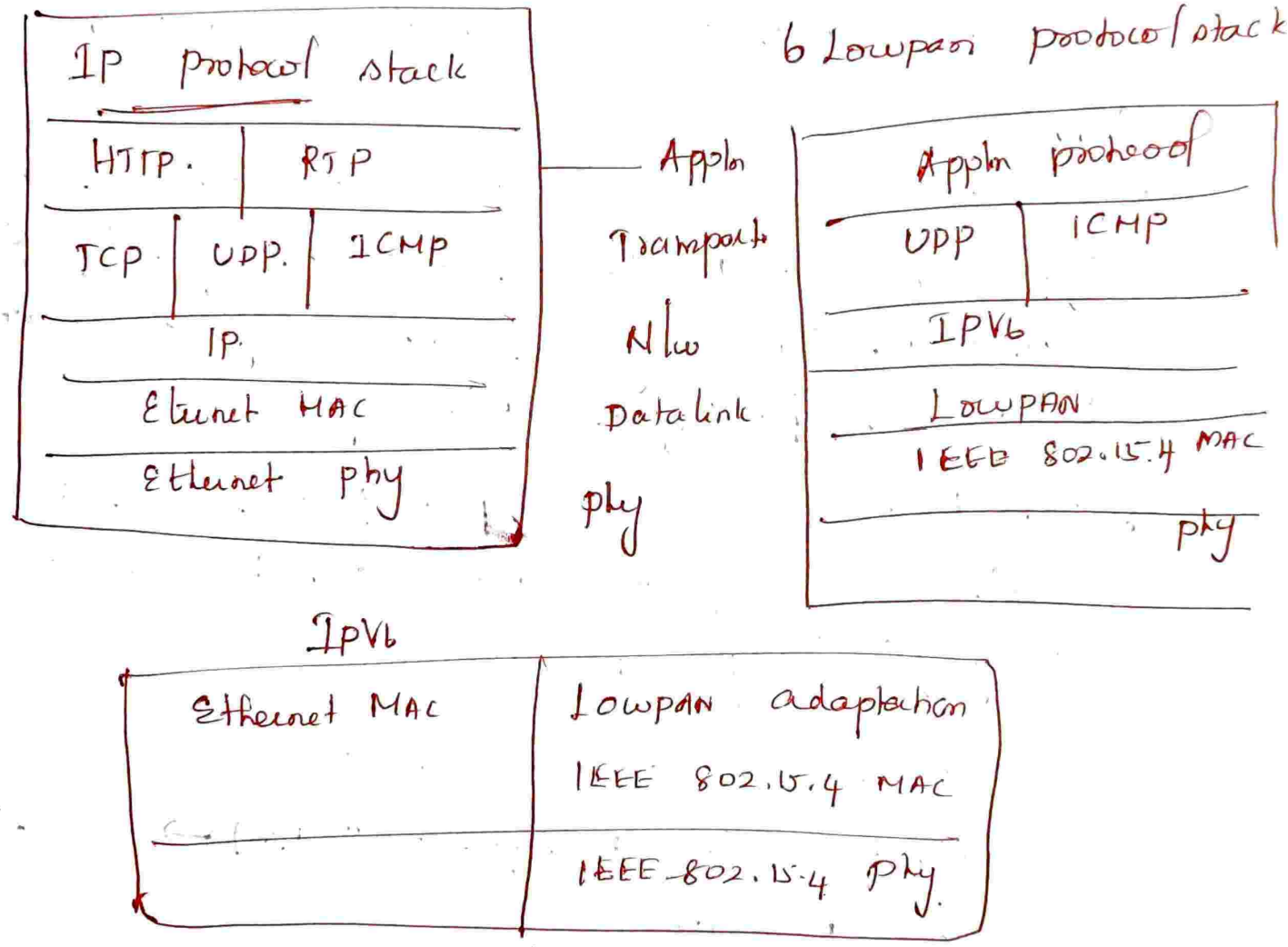
→ Internet control Message V6 (ICMV6) → used for control Messaging.

→ Ex: ICMP echo, ICMP destination Unreachable & Neighbour discovery.

→ Appln protocol are often Appln specific and in binary format.

→ 6Lowpan - - refered as edge router.
This transformation is transparent, efficient, Stateless

Lowpan Adaptation is an edge router typically performed as part of the 6 Lowpan Nlw Interface and is usually transparent to the IPV6 protocol stack itself.



nk layers for 6LowPAN

→ Basic requirements for a link layer to support 6LowPAN are framing, Unicast transmission and addressing

→ Addressing is required for to differentiate b/w nodes on a link. and to form IPV6 address which are then elided by 6LowPAN Compressions.

→ It is highly recommended that a link supports Unique addresses by default (EUI-64) to allow for stateless Auto configuration.

→ Multi access link should provide a broadcast services. Multicast service is required by std IPV6. but not by 6LowPAN. IPV6 requires a Maximum transmission unit - 1280 bytes for fragmentation purpose

→ A link should provide payload size at least 30 bytes in length to be useful.

→ UDP & ICMP include 16 bit checksum.
↳ for error checking.

→ IP Section may not always be practical for 6LowPAN. It is highly recommended that links include strong encryption and Authentication.

Addressing

→ IPv6 addresses are typically formed automatically from the prefix of the LoWPAN and the link layer address of wireless Interface.

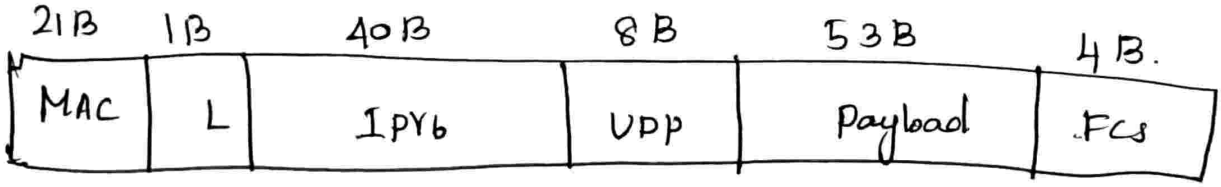
→ Low power wireless technologies support link layer addressing, a direct Mapping b/w link layer address and the IPv6 address is used for achieving compress

→ IPv6 addresses are 128 bits in length. and 64 bit prefix part and 64 bit interface identifi

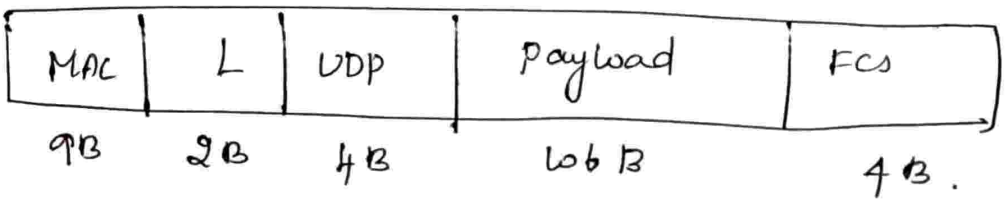
→ IID has a direct Mapping to the link layer addr therefore avoiding need for address resolution.

→ IPv6 prefix is acquired through Neighbour discovery Router Advertisment Message as on Normal IPv6 link.

Header format



Full UDP/IPV6 (64 bit Addressing).



Routing:

IP mobility solution consider technique for preserving the IP address of a node as it moves from one point of attachment to another, along with methods for forwarding traffic to a node while roaming and performing route optimization.

→ IP routing on the otherhand deals with maintaining routing tables on IP routers which indicate which next-hop forwarding decision should be made for the destination of an IP packet.

→ Routing is challenging for 6LOWPAN, with low power and lossy radio links, battery powered nodes, multihop mesh topologies and frequent topology change due to mobility.

→ 6LOWPAN uses Routing Over low power and lossy (ROLL) N/w.

→ As IP N/w are packet switched, as opposed to circuit switched, forwarding decisions are made

hop by hop, based on the destination add in a packet.

→ Therefore reaching a destination node in a N/w from source to destination address is a packet. node require building a path from source to destination node in a N/w route tables on nodes along the path. IP addresses are structured, and this structure is used to group addresses together under a single route entry.

→ LOWPAN Routers typically perform forwarding on a single wireless interface. ie. they receive a packet on their wireless interface from one node and then forward it to the next-hop destination using the same interface. This is an important difference to how forwarding on IP routers normally works. Where packets generally are forwarded b/w interface.

→ The reason for this model is that typically not all nodes in a LOWPAN are reachable in a single wireless transmission.

→ IP forwarding is used to provide full connectivity over multiple hops within the same link

→ A LoWPAN has a flat Address space as all nodes in a LoWPAN share the same IPv6 prefix.

→ This is due to the way 6LoWPAN compression is achieved using the fact that all nodes in the N/w know common information to elide (or) compress fields. Therefore 6LoWPAN routing tables only contain entries to destination addresses in the LoWPAN along with default routes.

→ LoWPAN are stub N/ws. and are not meant to be transit N/ws b/w two different subnets. This simplifies the requirements for LoWPAN Routers.

Distance Vector Routing

Using this approach, each link (node) is assigned a cost using appropriate route metrics. When sending a packet from node A to node B, the path with the lowest cost is chosen.

→ The routing table of each router keeps soft state route entries

for the destination it knows about, with the associated path cost.

→ Routing information is updated either proactively (periodically) or reactively (on-demand) depending on the routing algorithms.

→ Distance Vector Algorithms are commonly applied to 6LowPAN.

Link state Routing

In this Approach, each node acquires complete information about the entire N/w. called a graph

→ each node floods the n/w with information about its link information to nearby destination.

After receiving the link state reports from sufficient nodes, each node then calculate a tree with shortest path from itself to each destination

→ The tree is used either to maintain the routing table in each node for hop by hop forwarding (or) to include a source route in the header of IP packet.

Proactive Routing:

Algorithms using a proactive approach build up routing information on each node before the routes are needed. Thus they proactively prepared for the data traffic by learning routes to all possible or likely destinations.

→ Most protocols that are used in inter-domain (or) intra domain IP routing use a proactive approach as topologies are stable.

→ Advantage:

Routes are immediately available when needed but this comes at the cost of increased signalling overhead especially with frequent topology changes and increased state for routers.

Reactive Routing

Mobility

LoWPAN Nodes, for example in asset Management, often tend to be mobile. In some cases, such as with body area N/ws. the network itself may even be mobile

→ All of these uses of the wireless networks require us to deal with node mobility between edge routers in the same LoWPAN. between LoWPAN and possibly b/w network domains.

→ As the same time, active data flows may be in progress in application servers may need to know how to reach tracked devices.

Mobility types:

Roaming:

A process in which a mobile node moves from one network to another typically with no existing packet stream.

Handover

A process in which a mobile node disconnects from its existing point of attachment and attaches itself to a new point of attachment. Handovers may include operations at specific link layers as well as at the IP layer in order for the mobile node to be able to communicate again.

→ one or more application packet stream typically accompany the mobile node as it undergoes handover.

Physical Movement:

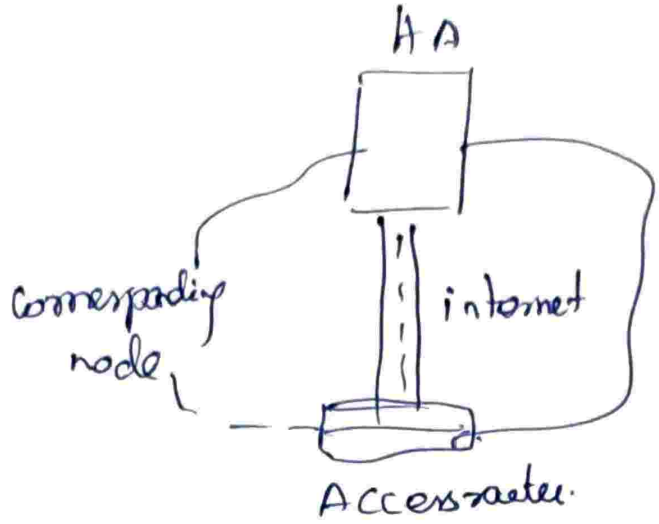
The most evident reason for mobility is when nodes in a network physically move in relation to each other. Which changes the wireless connectivity between pair of nodes.

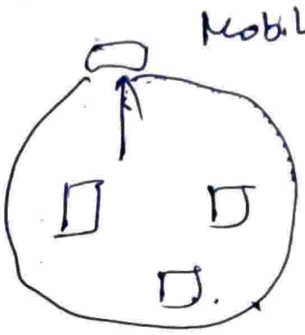
Radio channel

Changes in the environment cause change in radio propagation called fading.

1, 10, 13, 15, 17
18, 20, 22, 29, 41
48, 53
301.

NEMO → N/w mobility.





This changes often require topology change even without physical movement.

Network Performance:

Packet loss and delay on wireless network may be caused by poor signal strength, collisions, overloaded channel capacity or node congestion.

→ High packet loss may cause a node to change its point of attachment.

HEADER COMPRESSION

→ An Important characteristics of 6LOWPAN is the limited payload size of packet. provided by IEEE 802.15.4 about half of which would be consumed by the size of an IPv6 header already while larger packet can be sent using fragmentation and reassembly.

→ Header compression can be performed end to end but is then limited to comprising the headers that are within the payload of the IP header, as the routers on the way between compressor and decompressor still have to see full IP header.

→ The Largest header is many IPv6 header stacks is the IP header itself. This is not very efficient.

→ Instead of most header compression scheme

operates hop-by-hop is as part of the adaptation layer.

→ 6LowPAN challange.

Header size calculation.

→ Ipv6 header is 40 octets, UDP header is 8 octets

→ 802.15.4 MAC header can be upto 25 octets (null security) or 25+21 = 46 octets

→ with the 802.15.4 frame size of 127 octets

- 127 - 25 - 40 = 57 octets (null security)
- 127 - 46 - 40 = 33 octets (AES-CCM-128)

Of space left for Application data

Ipv6 Requirements

Ipv6 require that links support an MTU of 1280 octets

→ Link layer fragmentation / reassembly is needed.

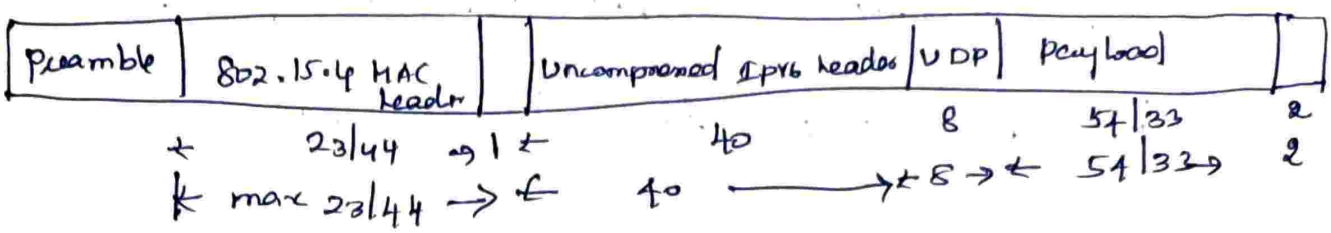
6LoWPAN dispatch codes:

All LoWPAN encapsulated datagrams are prefixed by an encapsulation header stack.

→ each header in the stack starts with header type field followed by zero or more header fields

Bit Pattern	Shortcode	Description
00 xxxxx	NALP	Not A LoWPAN packet
01 000001	IPV6	Uncompressed IPV6 address
01 000010	LoWPAN-HC1	HC1 Compressed IPV6 address
01 010000	LoWPAN-BC0	BC0 Broadcast header
01 111111	ESC	Additional Dispatch octet
10 xxxx	MESH	Mesh routing header.
11 000xxx	FRAG1	Fragmentation header (first)
11 100xxx	FRAGN	Fragmentation header (subseq)

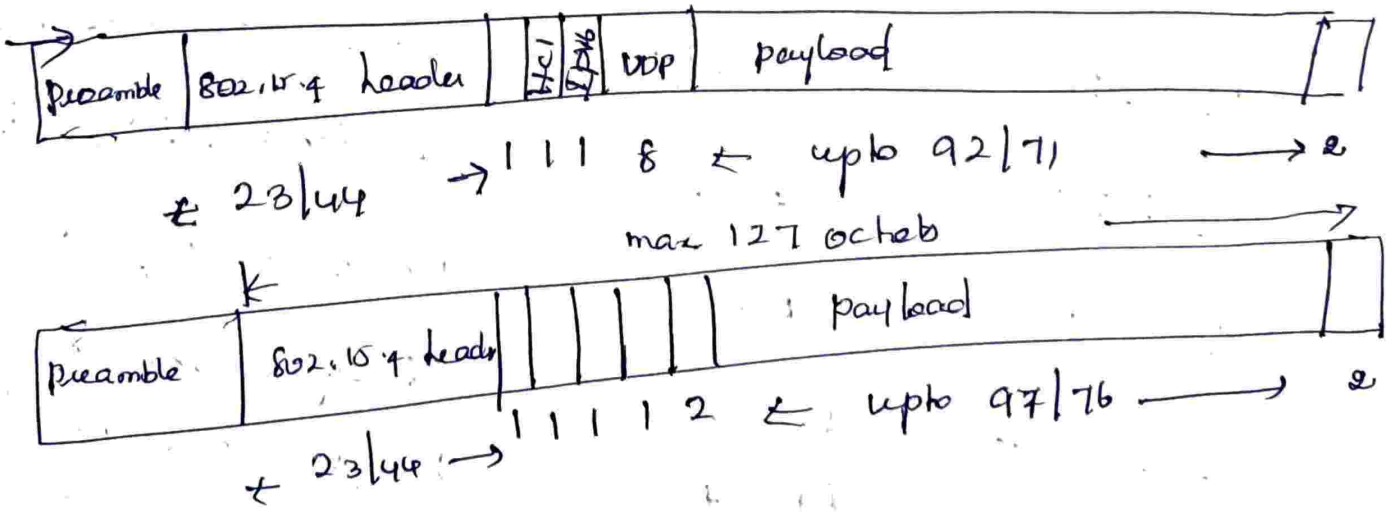
6LowPAN frame formats Uncompressed.



Dispatch code (01000001_2) indicates no compression.

→ upto 57|33 octets left for payload with a max. size MAC header with null | AES-CCM-128 Security

→ The relationship of header information to application payload is obviously really bad.



→ Dispatch code $(01000010)_2$ indicates HC1 compression.

→ HC1 compression may indicate HC2 compression follows.

→ This shows the maximum compression achievable for link-local Addresses.

Any non-compressible header fields are carried after the .Hc1 or Hc1/Hc2 tags (partial compression)

Compression principles:

→ Omit any header fields that can be calculated from the context, send the remaining fields unmodified.

→ Nodes do not have to maintain compression state (stateless compression)

→ Support (almost) arbitrary combinations of compressed/uncompressed header fields.

Stateless header Compression

→ IP + UDP header Compression : stateless.

→ called HC1 – HC2 Compression (Not recommended)

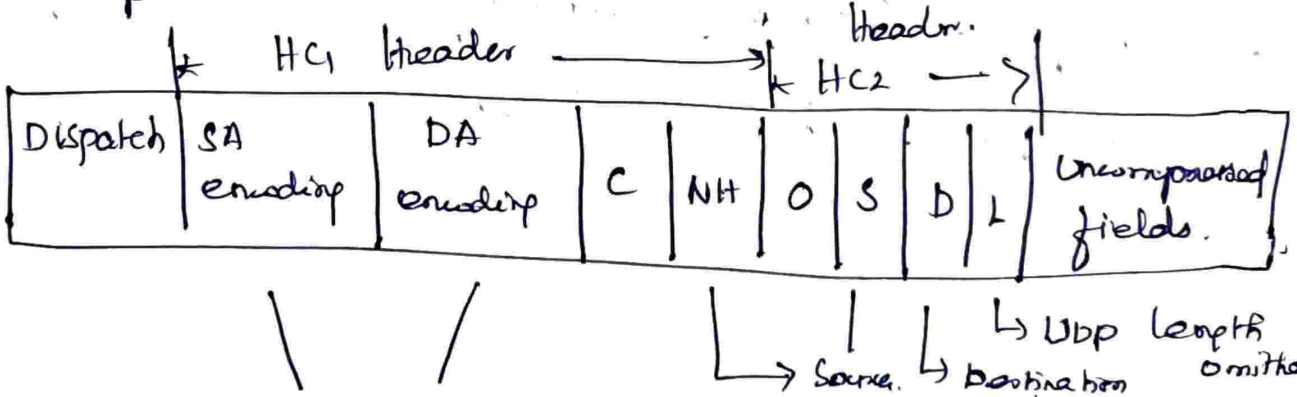
→ Ip version field is omitted.

→ Flow table field if Zero is omitted and C = 1

→ Only 4 byte UDP ports are sent if bl.w

bl6l6_bl631 (FOBx)

→ UDP length field is omitted, IP addrs are Compressed.



	Prefix	IID
00	Uncompressed	Uncompressed
01	Uncompressed	Derived from L2
10	FE80::/80 omitted	Uncompressed
11	FE80::/64 omitted	Derived from L2

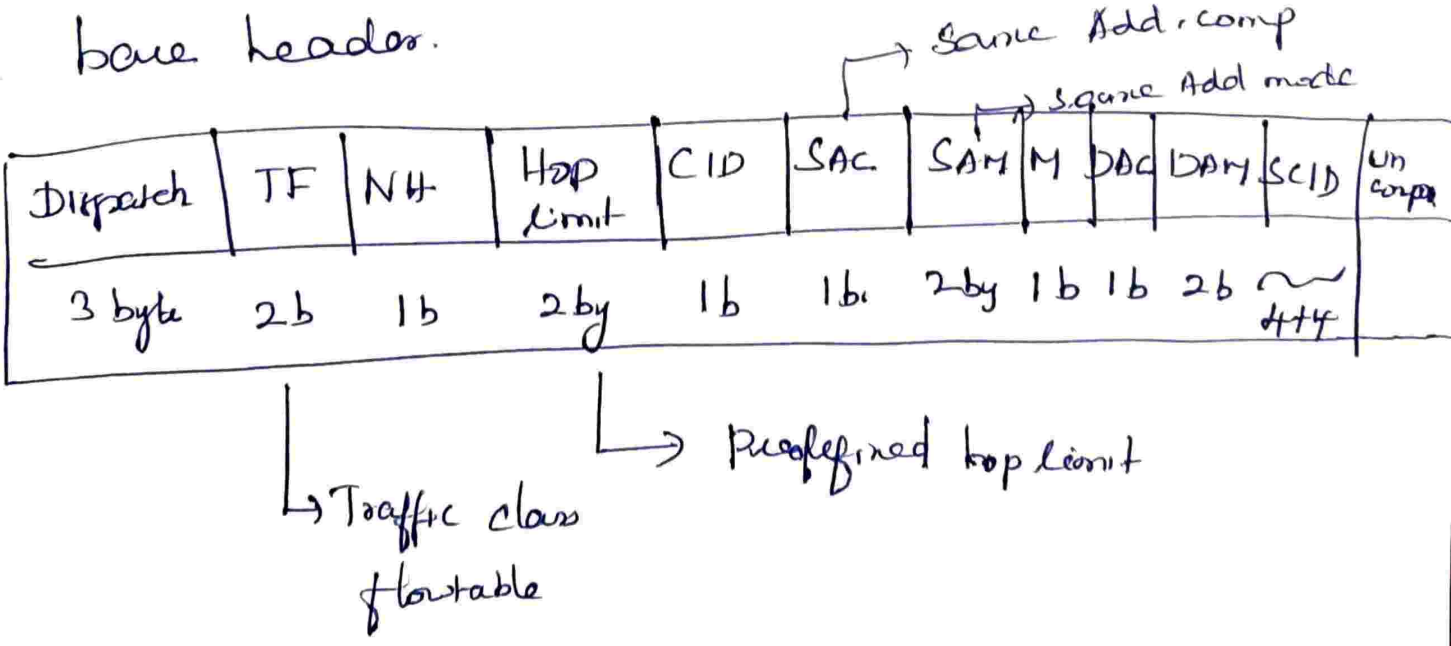
Context - Based Header Compression

To enable the compression of global addresses new specification assumes that there is a way for nodes to establish some Additional context when joining the LowPAN.

HC1 works only with link-local Addrs.

→ Need globally routable IPv6 addresses for outside nodes.

IPHC uses 3 bytes dispatch code and a 13 bit base header.



MANET Routing protocol.

→ The mobile adhoc N/w working group at the IETF was formed in 1997

→ The WG has produced a large variety of routing protocol which can be categorized by their scheme for dealing with route updates, proactively or reactively and by their routing technique.

→ In addition to Routing protocols, MANET has also produced valuable work on basic mechanisms for supporting routing in these environments. A common packet format for use by all MANET protocol has recently been developed in RFC 5444.

→ The protocols and mechanisms developed in MANET can be applied to 6LoWPAN N/ws. They are especially useful for applications with very similar requirements to typical MANET Applications.

→ The biggest challenge for applying a MANET algorithm to 6LoWPAN is in reducing the overhead of signaling packet and simplifying the algorithms.

AODV

The Ad-hoc on demand distance vector protocol enables mobile ad-hoc multihop nlws by quickly estabilishing and maintaining routes between nodes, even with quickly changing dynamic topologies.

→ AODV creates routes to destination when needed for data communications and only maintains actively used routes.

→ It includes methods for local repair and includes a destination sequence number to ensure loop-free operation.

→ AODV is purely a route table management protocol after routes have been estabilished they are simply used by IP for forwarding.

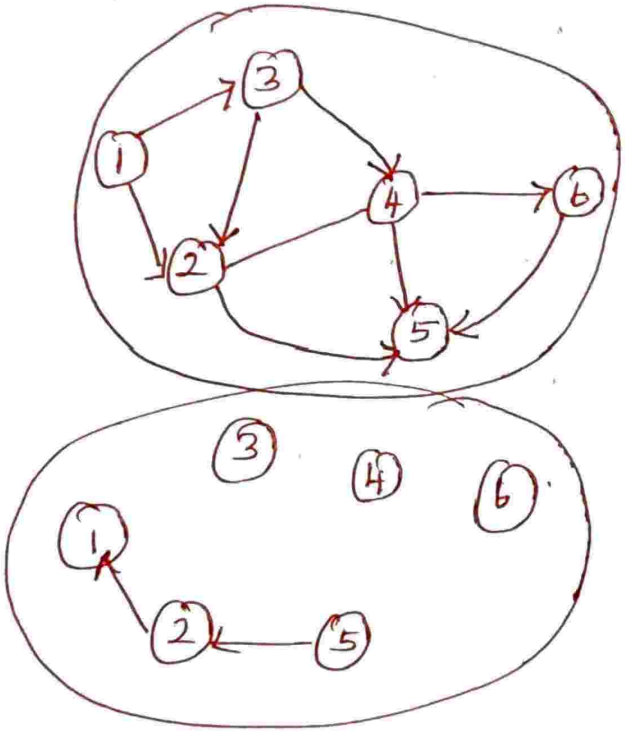
→ A small set of messages are used for discovering and maintaining routes by AODV and similar protocols

→ A route request (RREQ) is broadcast throughout the nlw in order to find paths.

→ This is responded to with a route reply (RREP) by an intermediate router or by the destination.

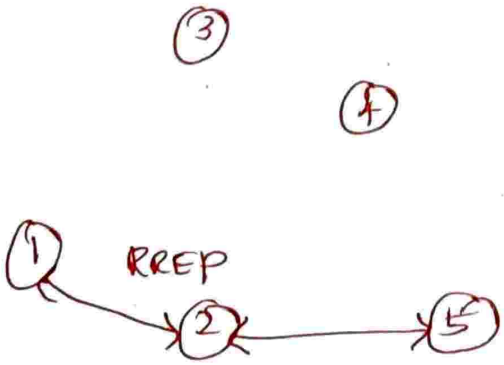
→ The route error (RERR) message is used to notify about broken links along a path. These message are sent over UDP, one hop at a time, between the AODV processes running on ad-hoc routers.

→



RREQ for node 5 broadcast over multiple hops.

RREP unicast back to node 1, creating route entries.



Route entries in 1, 2 and 5 enable forwarding.

→ Support for dynamic topologies and mobility is required.

→ Support is required for multipath routing, and thus multiple forwarding options.

→ Support is required for multiple node and route metrics, and their applications in constraint based and multi topology routing.

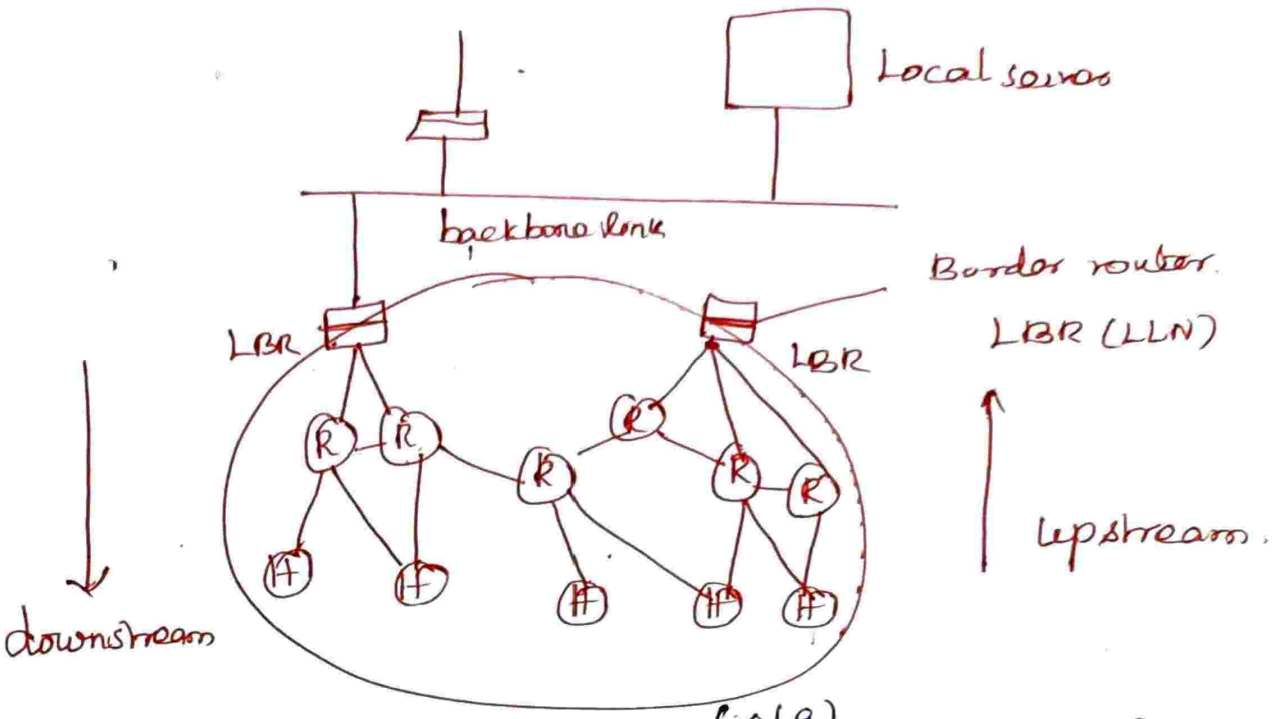
→ The evolution of metrics and support of multiple scenarios are important

→ A course grained depth metric is assumed for general use, which is independent of the specific scenario. It is not assured that this metric provides absolute loop avoidance.

→ The Generalized routing architecture is similar to that of an Extended LOWPAN defined by 6LoWPAN

→ Routers with interfaces to the LLN and another IP link are called LLN border Router (LBRs)

→ There may be several LBRs connecting an LLN to a backhaul or backbone link.



fig(a)

→ RoLL routing protocol operates within the LLN domain and terminates at the LBR.

→ The base of the routing protocol uses a graph structure b/w nodes and LBR, which can be seen in fig. (a)

→ The Basic topology needs to be discovered and maintained using minimal amount of signaling. After the basic topology is constructed, the routing protocol maintains upstream (from node to LBR) and downstream (from LBR to node) paths.

→ Forwarding along these paths is often performed using IPv6 forwarding. The co-ordination of Constrained routing, multi topology routing and traffic engg typically needs to be performed from a centralized place in the N/w. for which LBR are a logical place.

→ Other options may be performed in a distributed manner such as node to node optimized routes.

→ There are two concepts important to understand ROLL protocol operation.

Metric Granularity: ROLL uses the concept of a very granular route metric called depth. This metric used by the ROLL protocol mechanisms for building the graph, making use of siblings and for loop avoidance.

→ The evaluation of depth is simple for all routers and nodes and is independent of the application scenario.

Routing time scale:

ROLL makes routing decisions on two different time scales. route setup time and packet forwarding time,

→ In Route setup time the routing protocol maintain the basic graph topology and routing tables using static or slowly moving metric, which is a continuous process.

→ In addition, ROLL enables packet forwarding time decisions to be made using dynamic metrics on a packet by packet basis, for example the immediate use of alternative next hop routers upon failure.

Border Routing:

Simply routing within a LOWPAN is not very useful for the majority of wireless embedded Internet Applications.

→ Where most traffic flows are coming from the internet towards one or more LOWPAN Nodes, or from LOWPAN Nodes towards the Internet

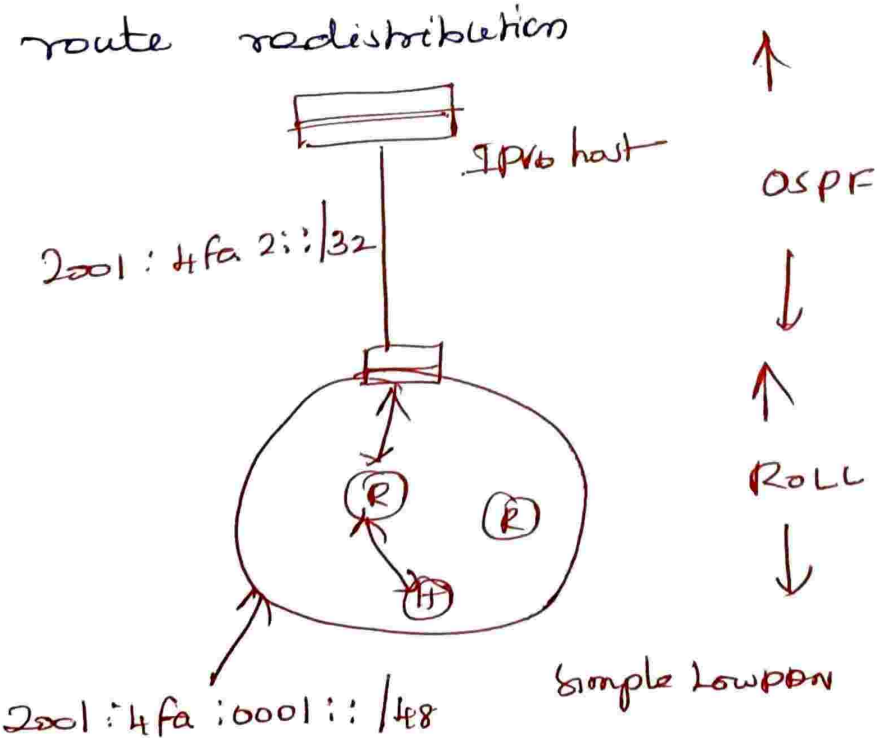
Border Routing between two IP routing domain is a common issue on the internet where intra domain and inter domain routing protocol interseet.

→ Traditionally it is less common on the very edge of the internet, as local access technologies such as NIFI. and Ethernet use bridging technique to connect device to the internet

→ Simple LOWPAN

→ Extended LOWPAN

→ route redistribution



Border Routing between an Extended LoWPAN and an IP n/w can be performed in two different places.

→ As the LoWPAN interfaces and IPV6 interfaces of edge routers in extended LoWPAN are in the same subnet, routing b/w them must be done using destination (exact match) route entries.

→ The simplest way to achieve this is to use edge router whiteboard entries to maintain these route table entries.

→

Mobile IPV6

→ The mobility of nodes on the internet can be dealt with at the network layer using a protocol called mobile IP (MIP), which was originally designed for IPV4.

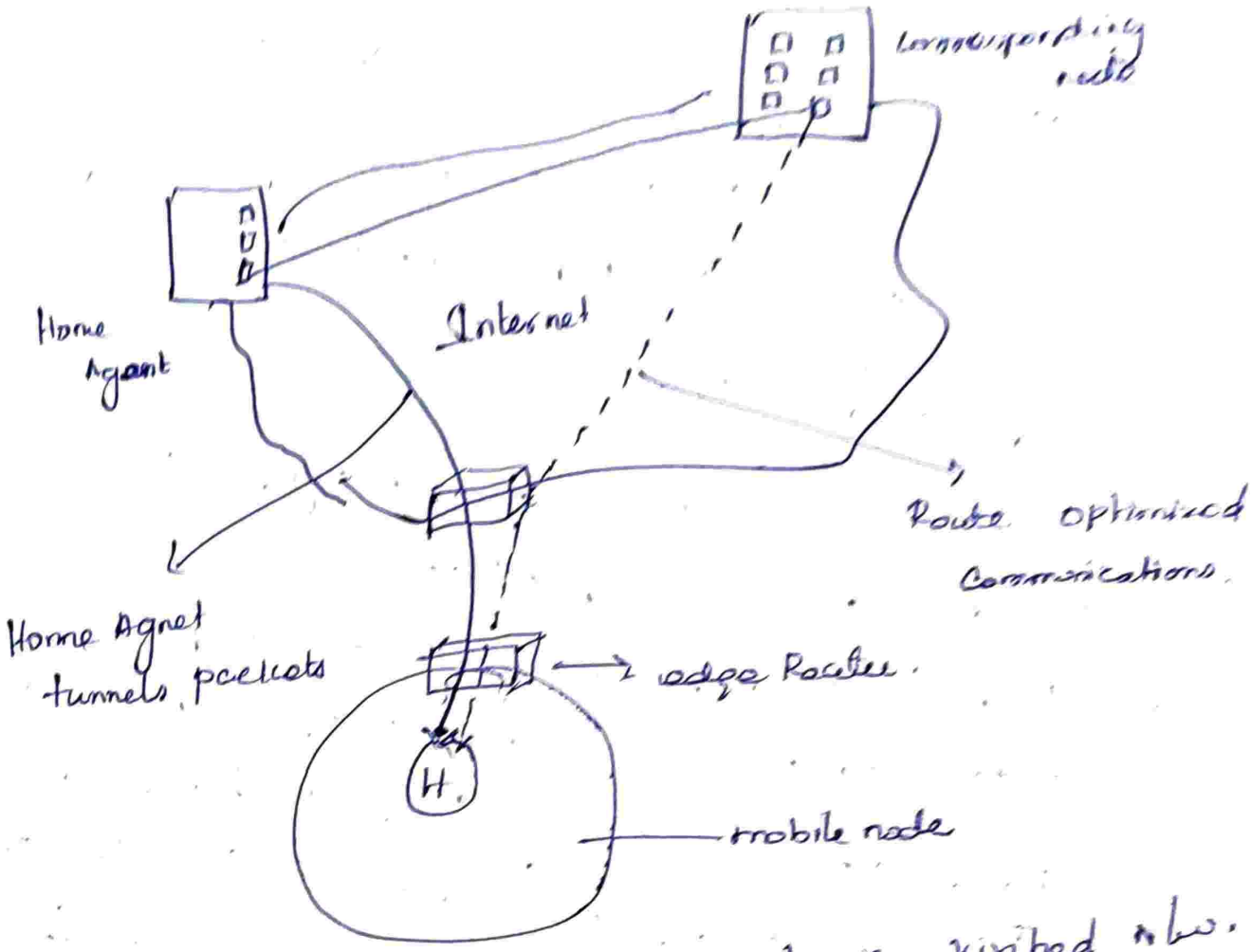
→ The new version take advantages of IPV6 mechanisms and provide route optimization.

→ Mobile IP does this using the concept of a home address which is associated with a host's home nw.

→ When a host is away from its home nw and attaches to another nw domain C called the visited nw), the new IP address it configured there is called its care-of address.

→ A Node communicating with a mobile node roaming in a visited nw is called the correspondent node.

→ The concept of 6LowPAN is for new simple IPV6 node to be able to participate in IPV6 nws. over low power, low B.w wireless links



→ When a mobile node roams to a visited nw.

it uses MIPV6 :

* After detaching that the subnet has changed, and that the node is no longer in its home nw. it sends a MIPV6 binding update message to its HA.

→ If the node doesn't know the location of its HA or its home prefix, there are methods to discover both.

→ The binding Update is acknowledged by the HA with a binding acknowledgment. These messages must be secured using IPsec methods.

→

Proxy Home Agent:

A proxy Home Agent is an entity which Perform MIPV6 functions on behalf of a local mobile node, interacts with the actual Home Agent of the node, and handles route optimization on its behalf. This greatly simplifies the function that a mobile node needs to perform to participate in MIPV6

→ In however this is an especially critical optimization.

→ The PHA is located in the visited n/w where a mobile node is receiving its traffic.

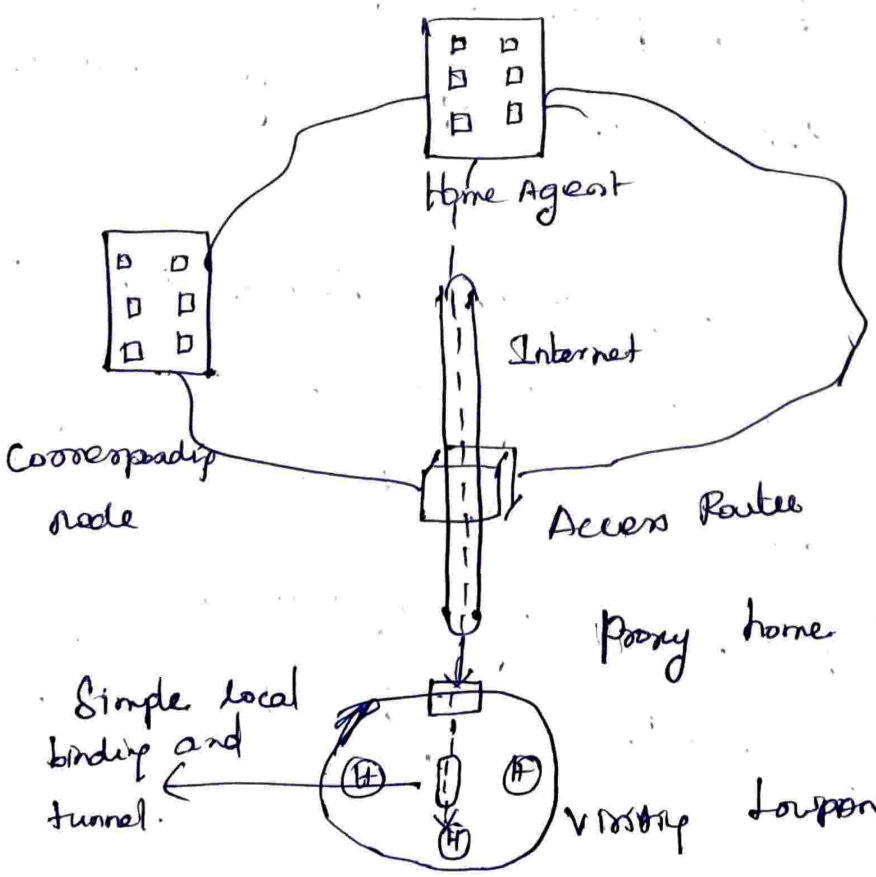
→ PHA act like a normal MIPV6 host, but additionally performs binding updates,

→ HA tunneling and Route optimization on behalf of other nodes.

→ This provides huge improvements in efficiency with regards to security associations and for route optimization which require tunnels and other state for every correspondent node.

→ Logical place to do this would be as an option for the binding ND node registration message.

→ Such an option would need to include the Home Agents address or home prefix, the nodes home address and some credentials.



Proxy MIPV6

→ The network based local mobility Management working group at the IETF works on solution for dealing with mobility locally within a domain

→ without requiring IPV6 nodes moving between points of attachment to change their IPV6 address or to implement MIPV6.

→ In M2M s/m this type of mobility b/w points of attachments within the same domain is quite common.

→ As a solution of this problem space the WG has standardized proxy MIPV6 which uses a local Hierarchical structure of routers to handle mobility on behalf of nodes.

→ The architecture of PMIPV6:

→ The concept of a PMIPV6 domain is introduced, which is controlled by a local mobility anchor (LMA)

→ The LMA function is usually combined with HA functionality.

→ The LMA handles the local mobility of nodes with the help of mobile access gateways (MAGs) which are points of attachment supporting PMIPV6. MAGs send proxy binding updates to the LMA on behalf of mobile nodes attached to them.

→ Using Bidirectional tunnels built btw each MAG and the LMA

→ LMA is then able to forward traffic to the mobile node always using its static address.

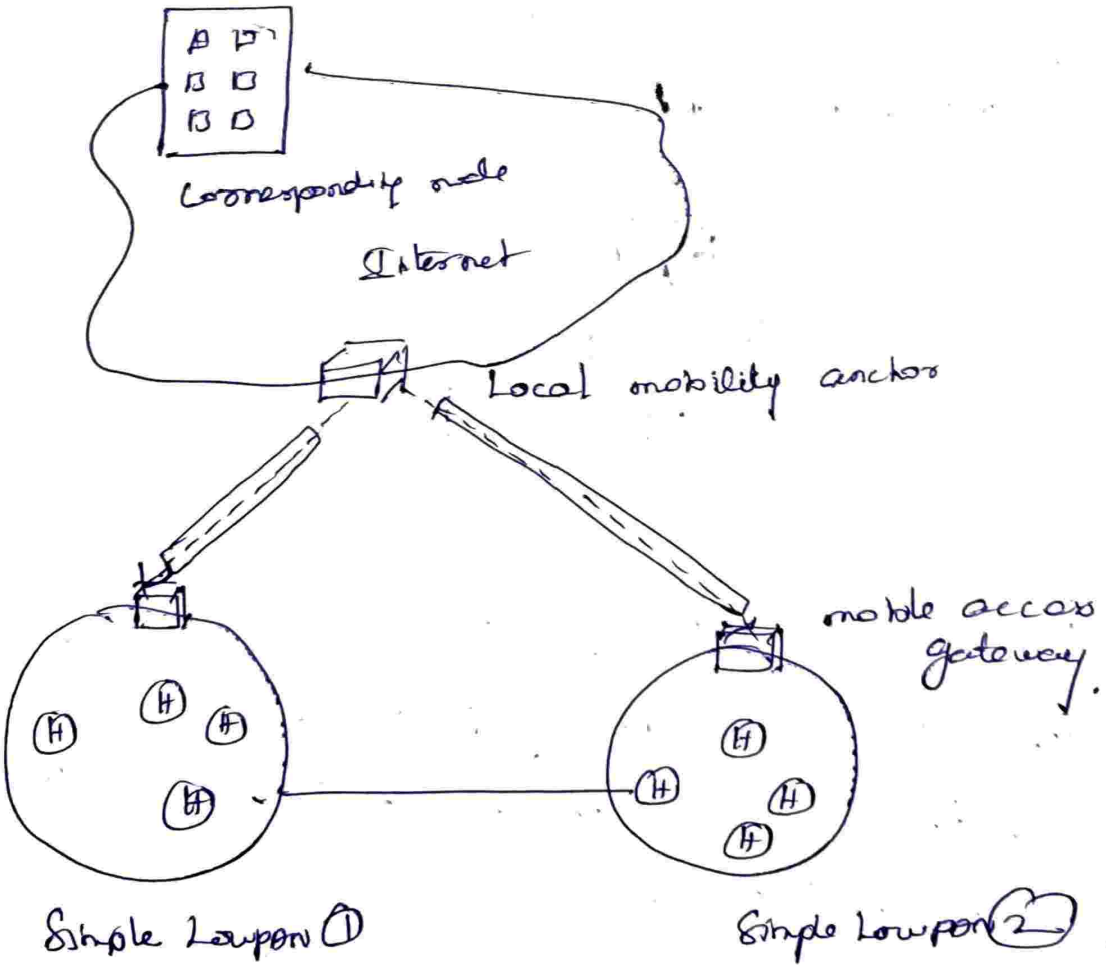
→ A binding in the LMA is needed b/w this address and the temporary address from the visited MAG.

→ The RS/RA exchange defined is not compatible with a multihop Route-over Lowpan, and would require each Lowpan Router to act as a MAG.

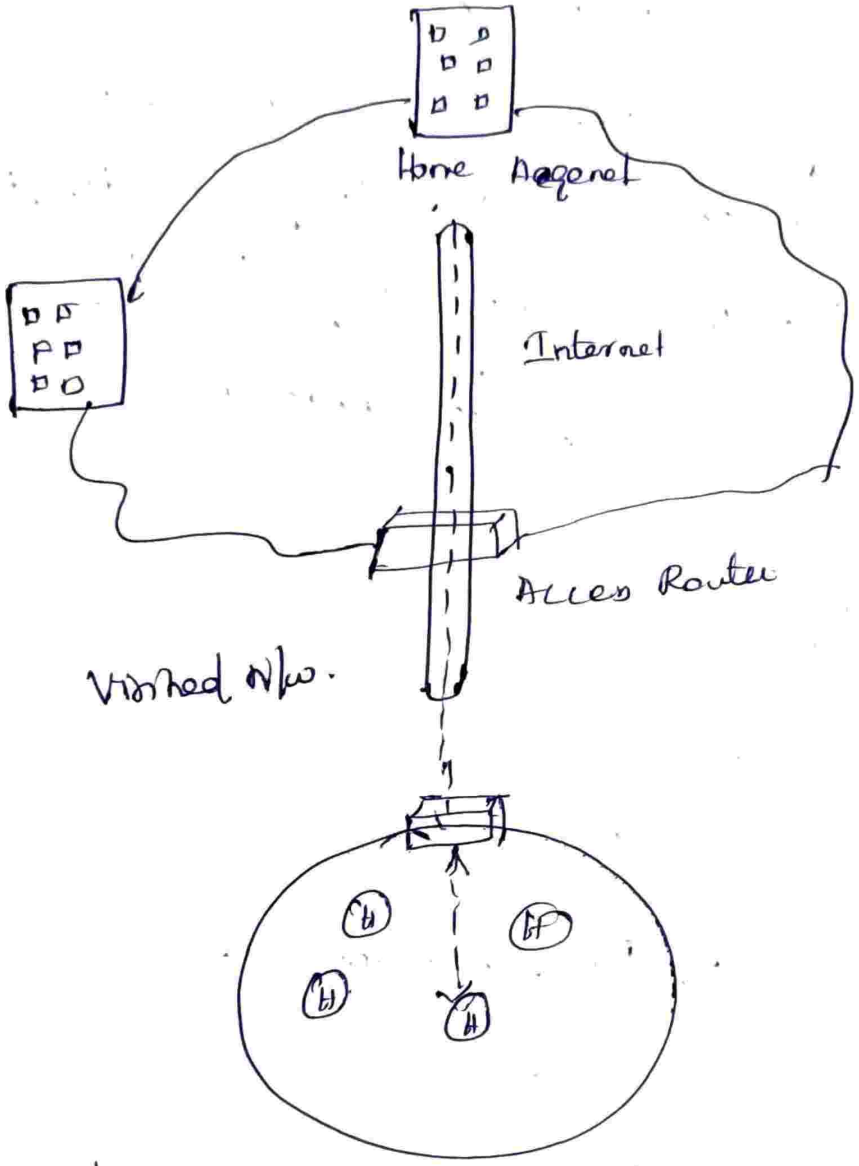
PMIPv6 is meant to provide a separate 64-bit prefix for each mobile node

→ PMIPv6 only enables a node to talk with its point of attachment, and requires, NS/NA exchanges which are not required by 6LoWPAN

→ Nodes otherwise using 6LoWPAN-ND...



NEMO



→ Network mobility is a solution for dealing with network mobility problems, when a router and the node attached to it, move their point of attachment all together

→ The philosophy behind NEMO is to extend mobile IP so that each node does not need to run mobile IP, instead only the router they are

UNIT: IV
APPLICATION

Introduction:

→ The Internet, and especially the web, has become Ubiquitous partly because of its ability to represent content in a universal way using a common Application protocol - the Hyper text transfer Protocol (HTTP)

→ HTTP includes File transfer Protocol (FTP), Real time protocol (RTP), session Initiation Protocol (SIP), service location protocol (SLP) and Simple Nlw Management protocol (SNMP)

→ Application protocol can be defined as all the messages and methods having to do with inter process communication via the protocol (IP).

→ IP protocols uses a socket based approach, where process and ports are identified by 16-bit source and destination port identifiers.

DESIGN ISSUES:

→ Application protocols used over 6LoWPAN need to take a number of requirement into account

Link layer:

Link layer issues include lossy asymmetrical links, typical payload size of 70 - 100 bytes, limited bandwidth and no native multicast support

→ In the presence of radio interference or packet collision there can be high packet loss ratios.

→ Additionally the nature of radio propagation, heterogeneous transmission amplification and receiver sensitivity results in asymmetrical links, with packets successful in one direction but not in the other.

→ Some link layers have even smaller size whereas other may have frames as large as hundreds of bytes.

→ The data rate over these radios are typically 20 - 250 kbits/sec, shared by all nodes on the channel, and quickly reduced over multiple hops.

Networking:

→ Networking related issues includes the use of UDP. limited compressed UDP port space and performance issues regarding the use of fragmentation.

→ Many Internet protocol today rely on TCP for a reliable connection oriented byte stream.

→ If the UDP source or destination ports are compressed, then the port space can be limited down to 16 ports

→ Although 6LowPAN supports fragmentation in order to handle larger payloads coming in from outside the LowPAN, the fragmentation of large payloads increases delay, packet loss probability and congestion.

Host Issues:

→ The IPV6 address changes each time the LowPAN node or the whole LowPAN changes its point of attachment, unless ones of the node or N/w mobility.

→ The mobility of Lowpan Nodes and sinks causes further problems as nodes will often not be continuously available during handover between points of attachment

→ Battery-Powered nodes are implemented to take advantages of aggressive sleep schedule in order to extend battery life. It is even common for a nodes to be active less than 1 percent of the time.

→ The intermittent node availability due to mobility and sleep schedules needs to be taken into account during application design.

Compression:

→ The small payload size available often require compression to be used on existing protocols.

→ Issues to consider include header and payload Compression, and whether it performed end-to-end or by an intermediate proxy.

→ When applying compression for web services an important design consideration is whether to use Compression end-to-end or to implement it with a proxy.

(3).

→ When Applying Compression for web services an important design consideration is whether to use compression end to end or to implement it with proxy.

→ one Appln protocols are directly useful with 6LowPAN such as RTP.

→ HTTP uses a text based human-readable format which takes space and is difficult to parse on simple embedded devices.

→ Techniques to compress XML such as binary XML (BXML) and efficient XML Interchange (EXI) along with embedded web service paradigms.

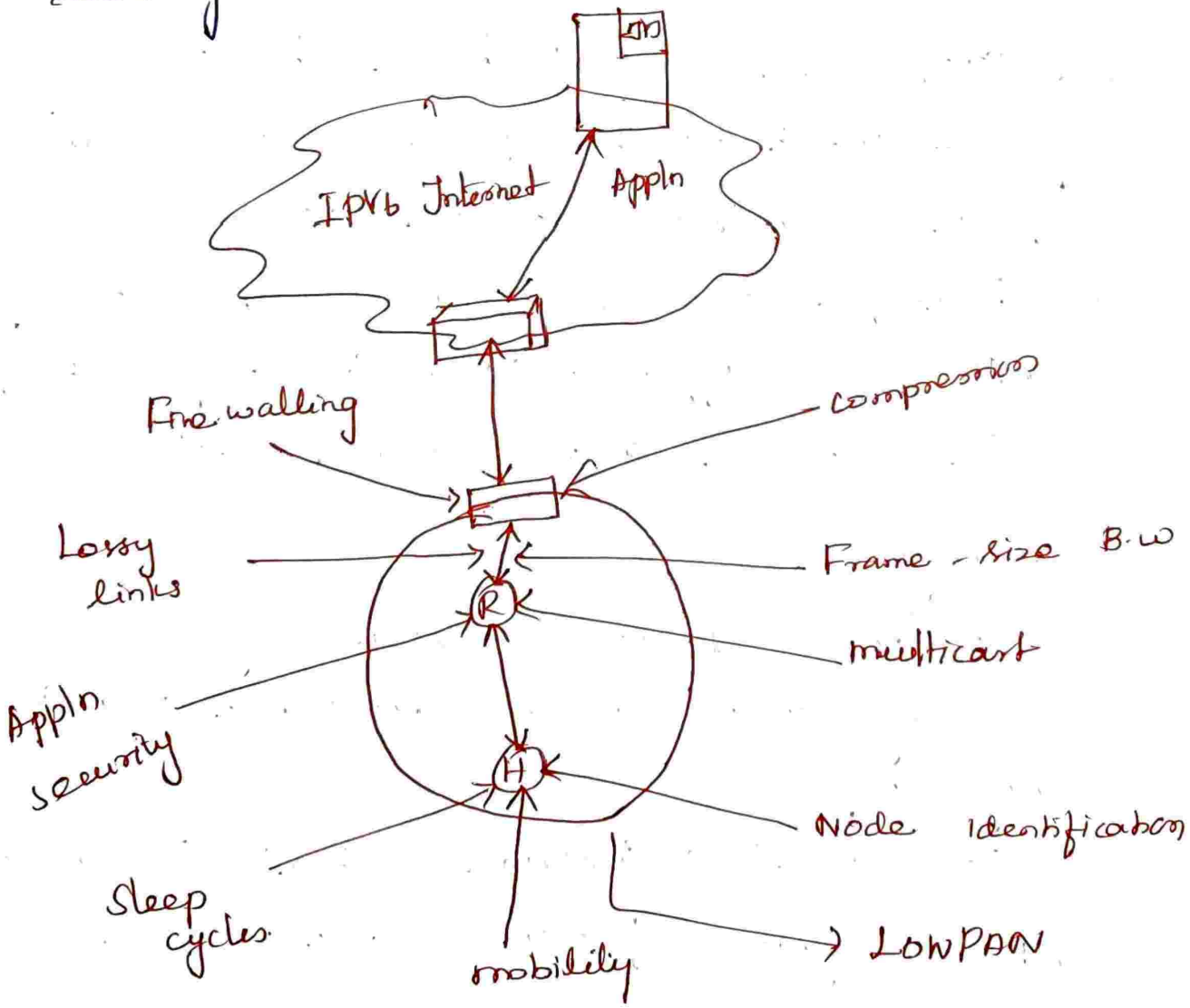
Security:

→ 6LowPAN depends on link-layer encryption for securing links in the LowPAN which protects a single hop.

→ IEEE 802.15.4 includes a built-in 128-bit AES encryption feature which secures each link along the way.

→ 6LoWPAN makes use of link layer encryption Intermediate nodes are susceptible to attack, requiring sensitive application to employ end-to-end application level security.

→ If an application is working with sensitive data then it should apply end-to-end application lay security.



PROTOCOL PARADIGMS:

→ There is a basic set of paradigms by which most internet application protocols function.

→ These include

1. End - to end paradigm
2. Streaming / sessions
3. Publish / subscribe.
4. Web services.

END-TO-END

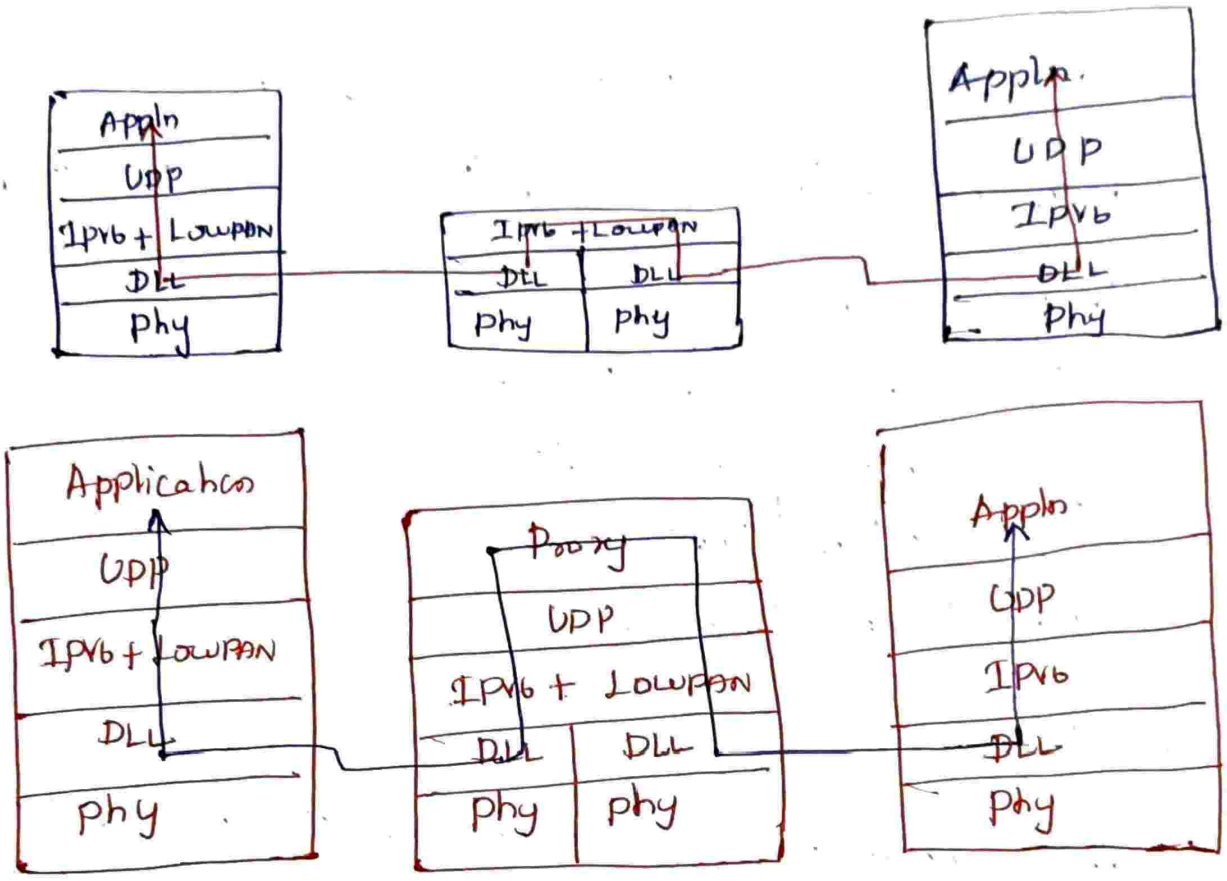
→ The Internet socket model is based on the use of the underlaying transport layer to provide a transparent data gram or byte stream service blw application processes / or so-called Application end points.

→ When considering the application layer this can be called an end to end paradigm where only the end-points participate in the application protocol exchanges.

→ Some application protocols also includes the possibility for intermediate nodes to inspect, cache or modify Application Protocols.

→ This is referred to as Proxying.

→



→ Protocol Compression of an existing protocol can be achieved either by supporting the compressed format natively on the IP Application end-point which is an end-to-end approach.

→ HTTP proxy that performs web-page caching.

Real time Streaming And Sessions:

→ Many Applications for embedded networks deal with real time data stream such as sensor data, audio or video.

→ The Internet protocol works on a best effort approach, without quality of service (QoS) guarantees.

→ Packets may arrive out of order with significant jitter

→ 6LoWPAN Applications performing real time streaming need to take this into account

→ Typically UDP is employed for real time applications as a reliable transport like TCP may make jitter worse.

→ Often, it is better to drop a packet than to delay a real time stream.

→ Internet protocols already provide a good framework for working with real time streams, which can be employed by 6LoWPAN Applications as well.

→ The Real time transport protocol (RTP) encapulates stream with appropriate timestamp and sequence information, while the companion RTP Control protocol (RTCP) is used to control the stream

→ If a relationship between the sender and receiver of a stream needs to be automatically setup and configured, the session initiation protocol (SIP) can be employed.

PUBLISH | SUBSCRIBE:

→ Publish | Subscribe is an asynchronous messaging paradigm in which publishers send data without knowing who the receiver is, and receiver subscribe to data based on the topic or content of the data.

→ pub| sub can be implemented using centralized brokers that match publishers and subscribers in a store and forward fashion, or in a distributed manner where subscribers filter messages directly from publishers.

(b).

This de coupling of the application and points allows for scalability and flexibility

→ For the Internet of things, pub/sub plays an important role as most applications are data centric, ie it is not so important who sends data but rather what the data is..

→ One good examples of a pub/sub protocol is the MQ telemetric transport (MQTT) which is a broker based enterprise pub/sub protocol for telemetry used widely by IBM.

→ This has been adapted for use in sensor N/ws with MQTTS.

Web Service Paradigms:

→ Web service are defined by the W3C as a software system designed to support interoperable machine to machine Communication over a N/w.

→ Web services as a whole commonly work b/w clients and servers over HTTP.

→ There are two types of web services are available
(1) service based web services (2) Resource based web services.

→ Both forms of web services will play an important role in Cloud Applications.

→ Service based web services use XML following SOAP format to provide Remote procedure calls (RPCs) between clients and servers.

→ These SOAP messages and sequences can be described using the web Service description Language (WSDL)

→ This paradigm is widely used in enterprise machine to machine systems.

→ A SOAP interface is typically designed with a single URL that implements several RPCs called Methods.

→ Example.

http://sensor1D.example.com/soap.

→ Methods

get sensor state (SensorID)
get sensor value (SensorID)
set Config (Parameter, value)
get Config (parameter)

COMMON PROTOCOLS:

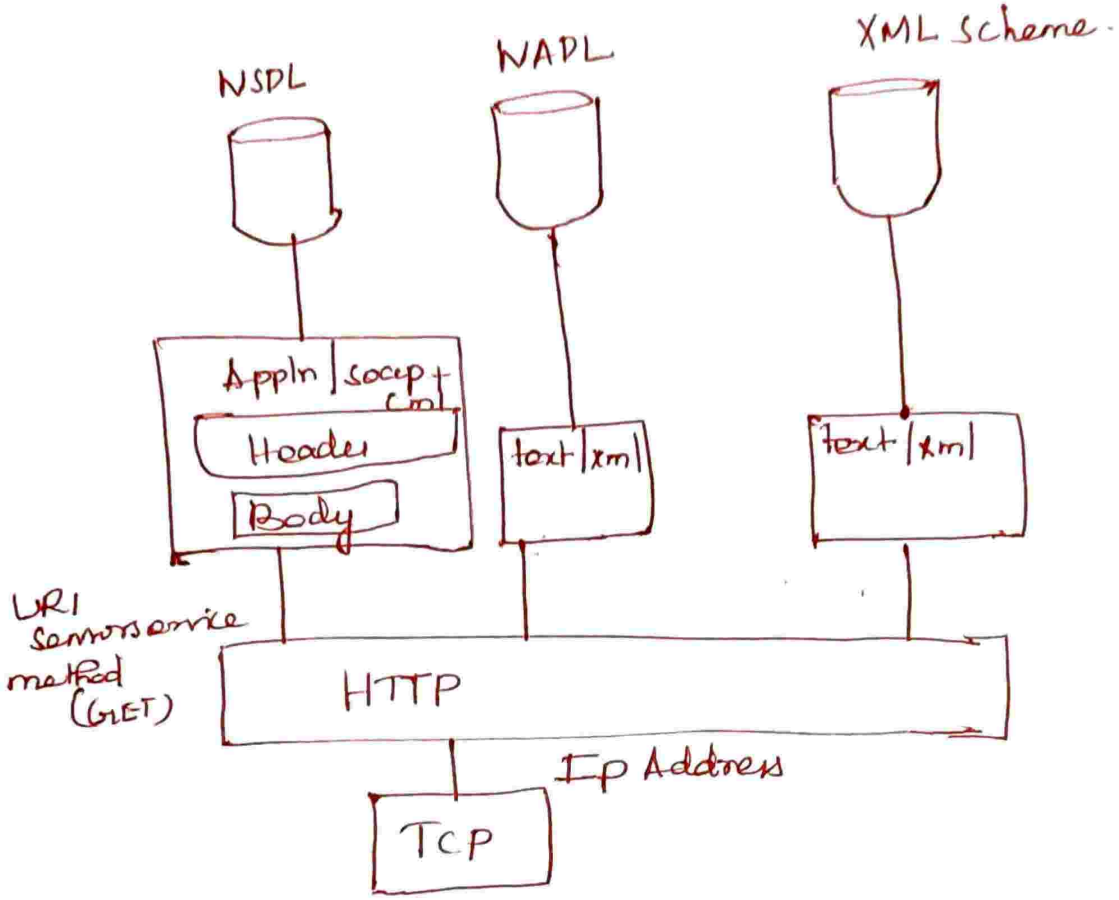
→ Web service protocols, MQTTS, zigbee ZAP service discovery protocol, SNMP, RTP/RTCP SIP and Industry specific protocols are commonly used or have good potential for use over 6LoWPAN.

→ Web Service Protocols (WSP)

→ The web service concept is hugely successful on the internet, especially in enterprise machine to-machine Internet silos

→ As many back end system incorporating information from LoWPAN devices will already be using existing web service principles and protocols, it is expected that 6LoWPAN will be integrated into the web service architecture

→ The use of XML, HTTP and TCP makes the adaptation of web services challenging for LoWPAN Nodes and Networks.



→ Web Services are Simply URLs available on an HTTP Server with services or resources accessible behind them.

→ This services may support any number of methods with corresponding response that are described by a WSDL document

→ SOAP (appln | soap + xml) is an XML format consisting of a header and a body, in which the body carries any number of message.

→ Resource based web services can also be realized using a REST design.

→ By using different HTTP methods on that URL the resource can be accessed.

→ Example: Sending an HTTP GET for /sensors/temp might return a text/xml body with the temp of the sensor.

→ REST designs make use of well known XML or other formats to give meaning to the content that can be understood by all parties.

MQ telemetry transport for Sensor Networks (MQTT-S)

The MQ telemetry transport (MQTT) is a lightweight publish/subscribe protocol designed for use in enterprise application over low-bandwidth wide area N/w.

→ The protocol was designed by IBM

→ MQTT uses a broker-based pub/sub architecture to which clients publish data based on matching topic names.

→ Subscribers then request data from the broker based on the topic Names.

→ Although MQTT was designed to be lightweight it requires the use of TCP and the format is inefficient over 6LoWPAN Nws.

→ In order to allow for MQTT to be used also in Sensor Networks, MQ telemetry transport for sensor Networks (MQTT-S) was developed.

→ This optimized protocol can be used over zigbee UDP/6LoWPAN or any other simple Nw providing a bi-directional datagram Service.

→ MQTT-S is optimized for Low bandwidth wireless Networks with small frame size and simple devices.

MQTT-S Architecture:

* MQTT brokers
* MQTT-S Gateways
* MQTT-S forwarders
* MQTT-S Clients.

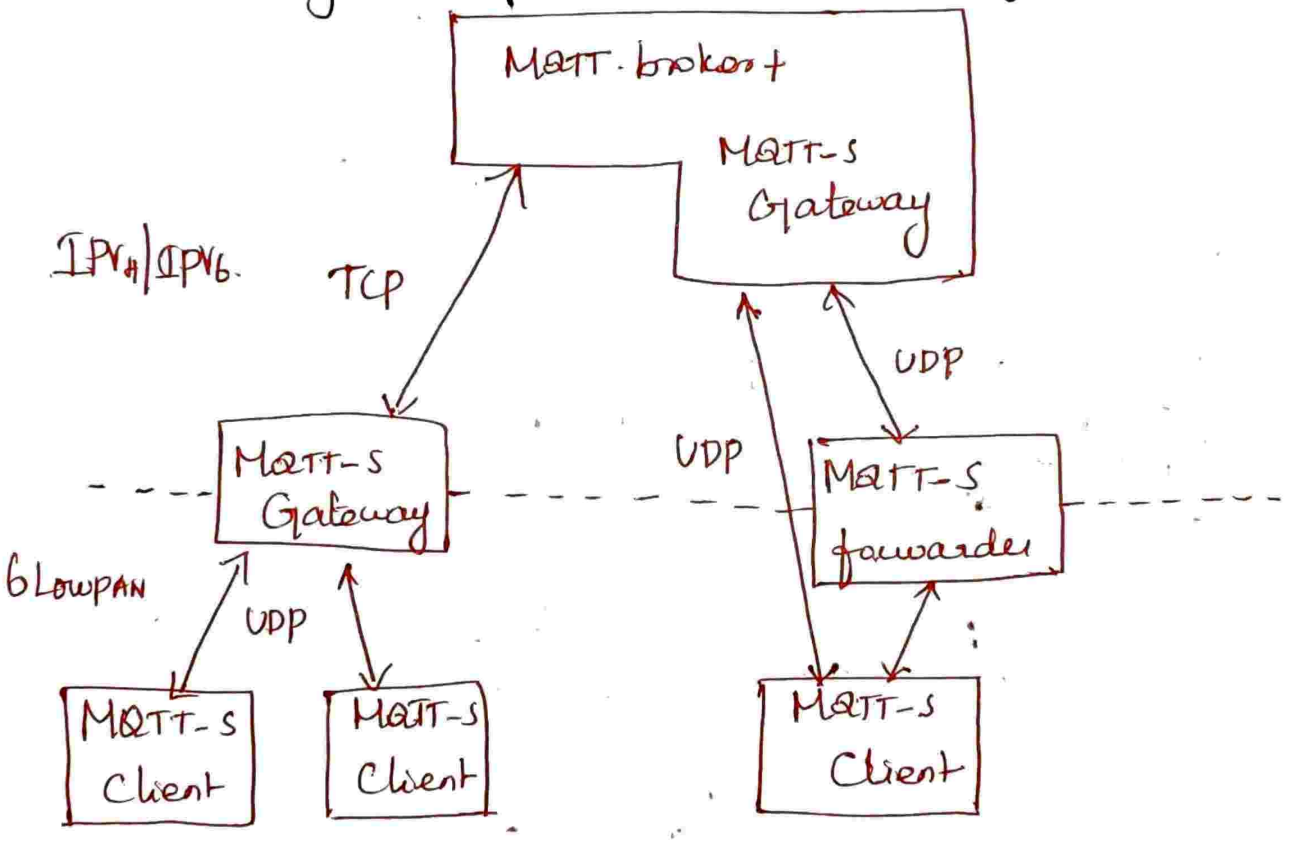
* Clients connect themselves to a broker through a gateway using the MQTT-S protocol.

→ The Gateway may be located eg: on the LoWPAN edge Router, or it may be integrated in the broker itself, in which case MQTT-S message are simply carried end-to-end over UDP.

Gateways translates b/w MQTT-S and MQTT

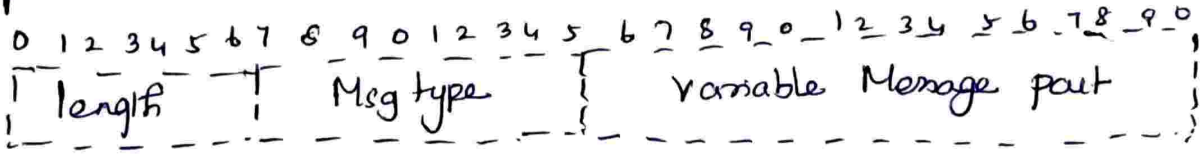
→ In case a Gateway is not directly available, forwarder are used to forward messages between clients and brokers.

→ Forwarders may not be needed with 6LoWPAN as UDP datagrams can be sent directly to a gateway.



MQTT-S Message structure:

→ It consists of a length field, a message type field and then a variable length message part.



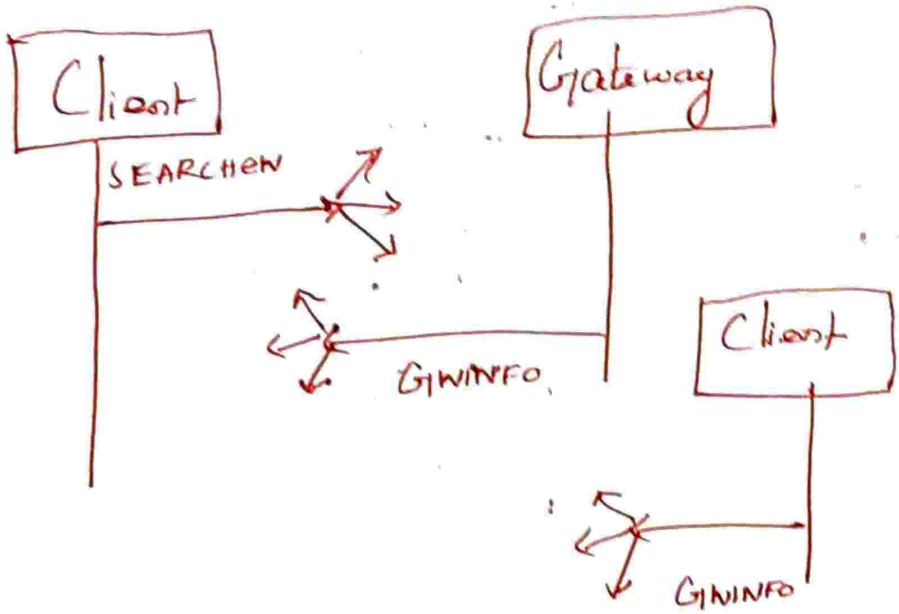
Protocol Operation

MQTT-S includes a gateway discovery procedure, which does not exist in MQTT.

→ Gateways send periodic Advertise message, and Clients may send SEARCHGW messages.

→ A GWINFO Message is sent to a client in response to SEARCHGW with basic information about the gateway.

→ Clients CONNECT to a gateway which responds with an ACK



DISCONNECT is used to end a connection or to indicate the sleep period.

→ Clients can connect with multiple Gateways which are able to perform load balancing.

ZIGBEE COMPACT APPLICATION PROTOCOL (CCAP)

Introduction:

→ The Zigbee Appln layer (ZAL) and zigbee cluster library (ZCL) specify an Appln protocol enabling interoperability b/w Zigbee device and at the application layer.

→ The Zigbee Alliance maintains a series of specifications for ad-hoc networking between embedded devices using a single radio, IEEE 802.15.4.

→ Typical Applns for zigbee include home automation energy Applns and similar local area wireless control Applications.

→ Zigbee makes use of a vertical profile approach over the ZAL and ZCL with profiles for different industry Applications such as zigbee Home Automation profile [Zigbee HA] or zigbee smart Energy profile

→ The ZAL and ZCL provide the key application protocol functionality in zigbee, enabling the

Exchange of Commands and data, Service discovery binding and security along with profile Support.

→ These protocols uses Compact binary formats with the goal of fitting in small IEEE 802.15.4 frames.

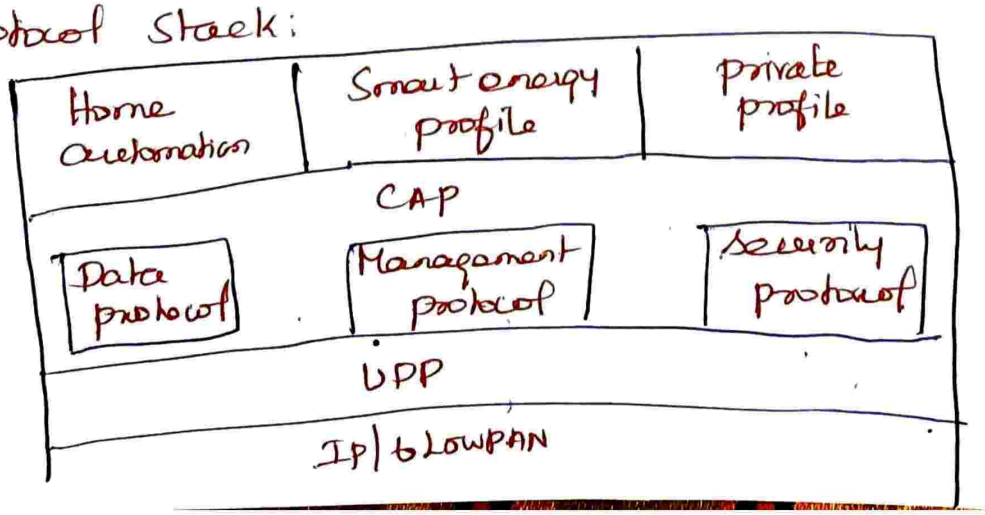
Compact Application Protocol (CAP)

A Soln for using Zigbee application protocol and profile over UDP/IP especially over 6LoWPAN has been proposed.

→ This specifications defines how the ZAL is mapped to standard UDP/IP primitives, enabling the use of any Zigbee profile over 6LoWPAN and standard IP stacks.

→ This Application of the ZAL for use with UDP/IP is called the Compact Application protocol (CAP)

Protocol Stack:



→ The functions of the ZAL and ZCL are implemented by the CAP.

→ The data protocol corresponds to the Zigbee cluster library.

→ The Management protocol corresponds to the Zigbee device profile handling binding and discovery.

→ Finally the security protocol implements zigbee Application Sublayer (APS) security.

→ Any Zigbee public or private application profile can be implemented over CAP in the same way that it would use the native Zigbee ZAL/ZCL.

→ This allows for Zigbee Application profile to directly be applied to IP N/ws.

→ The Main modifications to the ZAL has to do with using IP hosts and IP address instead of IEEE 802.15.4 hosts and IEEE 802.15.4 addresses.

→ Zigbee Appln layer messages are placed inside Upp datagram using CAP data protocol instead of Zigbee Network layer frames.

→ To Receive Unsolicited Notifications CAP listens to a well known UDP Port.

→ The ZAL identifies nodes by their 64-bit or 16 bit IEEE 802.15.4 MAC address.

→ In CAP this is replaced by a CAP address record which can Contain an IPV4 address plus UDP Port, IPV6 address plus UDP Port, or a fully qualified domain name plus UDP port.

→ These can be configured manually, Using DHCP a special DNS entry or using a CAP, Service discovery Message.

→ CAP Supports Secure transmission and the use of APS acknowledgement which provide limited Application protocol reliability.

→ The CAP data protocol is contained within the APS payload and it contains the ZCL command frame, with Support for all ZCL command types.

→ None of the ZCL commands require modification for use with CAP.

→ The CAP Management protocol modifies the Zigbee device profile Command frame to remove IEEE 802.15.4 specific frames or to modify the address where possible.

SERVICE DISCOVERY:

→ Service discovery is an important issues in wireless Embedded Internet Applications, where device are autonomic - also requiring the auto configuration of applications.

→ Service discovery is used to find which services are offered, what IP address they are located.

→ Some application protocol such as zigbee CAP or M@TT-S having their own built-in discovery features.

→ Frameworks such as OGC or SENSEI also have built in service discovery and description mechanisms.

→ The service location protocol is used for general Service discovery over IP N/ws.

→ SLP needs optimizations in order to be effectively used with 6LoWPAN because of the size of typical Messages.

→ There has been a proposal for a Simple service Location protocol (SSLP) which provides a simple lightweight Protocol for service discovery in 6LoWPAN N/ws.

Such a protocol could be easily interconnected with SLP running on IP Nlws by an SSLP translation agent located on an edge router thus allowing 6LowPAN services to be discovered from outside the LowPAN and vice versa.

→ SSLP supports most of the features of SLP, including the optional use of directory agents.

→ The SSLP header format consists of the four byte base header followed by specific message fields.

→ As in SLP, service type, scope and URLs are carried as strings.

→ Strings used with a scheme like SSLP should be kept as short as possible.

→ UPnP is a protocol aimed at making home devices automatically recognizable and controllable.

→ UPnP makes use of three protocols:
Simple service discovery protocol (SSDP) discovering device, the Generic Event Notification Architecture (GENA) for event Notifications and SOAP for controlling devices.

→ UPnP is not directly applicable to 6LoWPAN as it is Similar to SSLP

→ It may be possible to use UPnP, or a subset of it, over 6LoWPAN with web Service Compression and binding applied to UPnP descriptions and protocols.

→ However, this would require a special version of UPnP to be specified for use over 6LoWPAN and Similar Nlws.

→ The devices profile for web service (DPWS) describes a basic set of functionality to enable embedded IP devices with web-service-based discovery, device descriptions, messaging and events.

→ The objectives of DPWS are similar to those of UPnP, but DPWS was a pure web-service Approach.

→ DPWS has recently been standardized under OASIS.

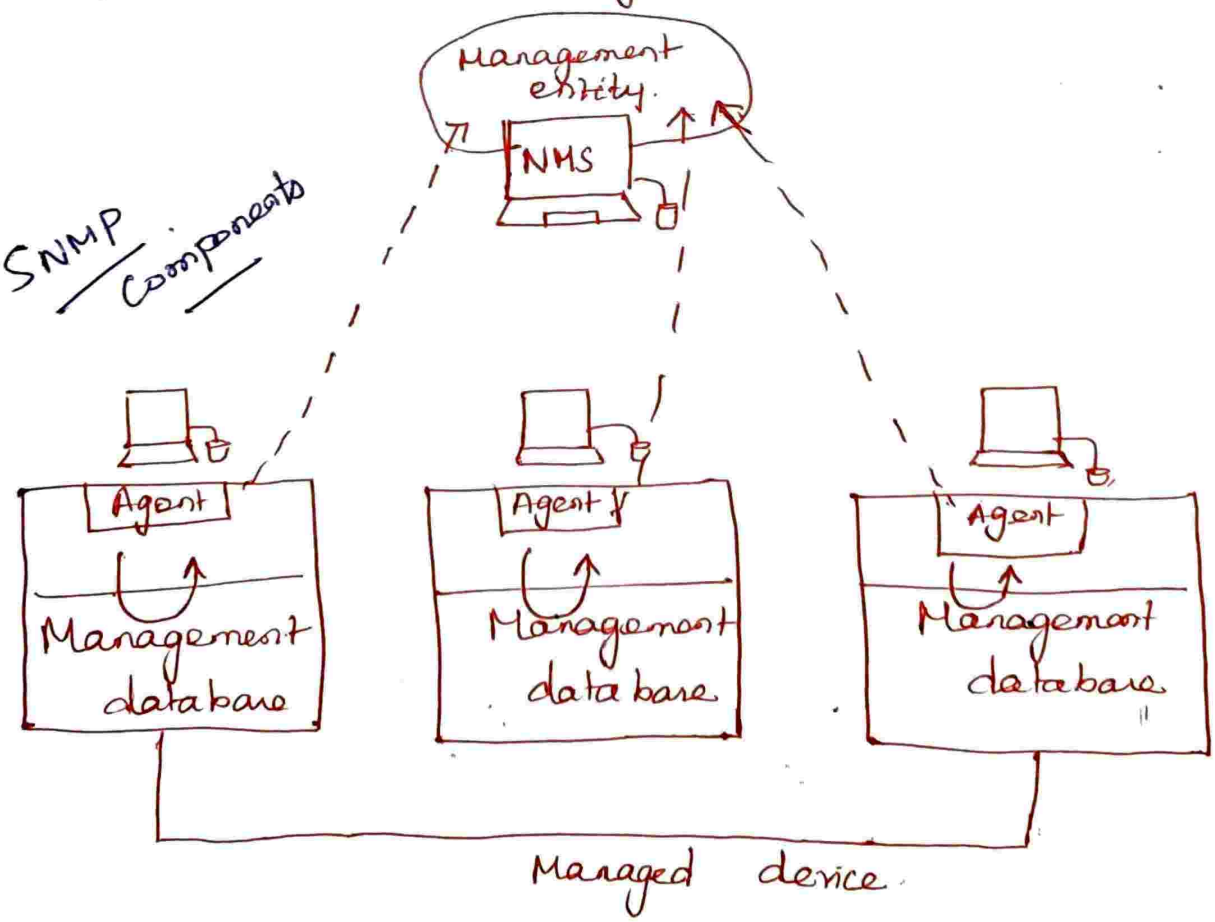
→ DPWS has been gaining is popularity for use in enterprise and industrial S/ms as devices using DPWS can be automatically integrated into back end Systems based on web services.

NETWORK MANAGEMENT PROTOCOLS:

→ Network Management is an important feature of any network deployment and a certain amount of Management is necessary even for autonomous wireless embedded devices.

→ There are several ways of performing Management in IP Networks eg: the simple Management protocol, web services or proprietary protocols.

Simple Network Management Protocol (SNMP)



→ SNMP is a standard for the Management of the Network infrastructure and device in IP N/w.

→ It is part of the TCP/IP Protocol suite

→ SNMP is an application layer protocol that uses UDP port Number 161/162.

→ SNMP is used to monitor the N/w, detect N/w fault, and Sometimes even used to Configure remote devices.

→ It includes an Appln protocol, a database scheme and data objects.

→ The Current Version is SNMPV3.

→ SNMP exposes variables to a Management S/w which can be GET or in some Cases SET in order to Configure or control a device. @BULL = The variable exposed by SNMP are organised in hierarchies called Management Information Bases (MIBs)

SNMP Manager:

→ It is a Centralized System used to monitor N/w. It is also known as Network Management stations

SNMP Agent:

→ It is a software Management module installed on a managed device.

→ Managed device can be Network device like PC, routers, Switches, servers etc.

Management Information Base:

→ MIB Consists of information on resource that are to be managed. This information is organized hierarchically.

→ It consists of objects instances which are essentially variables.

→ A Managed object is one of any number of specific characteristics of a managed device.

→ Managed objects are comprised of one or more object instances, which are essentially variables.

→ Two types of Managed objects exist: scalar and tabular.

→ Scalar objects define a single object instance

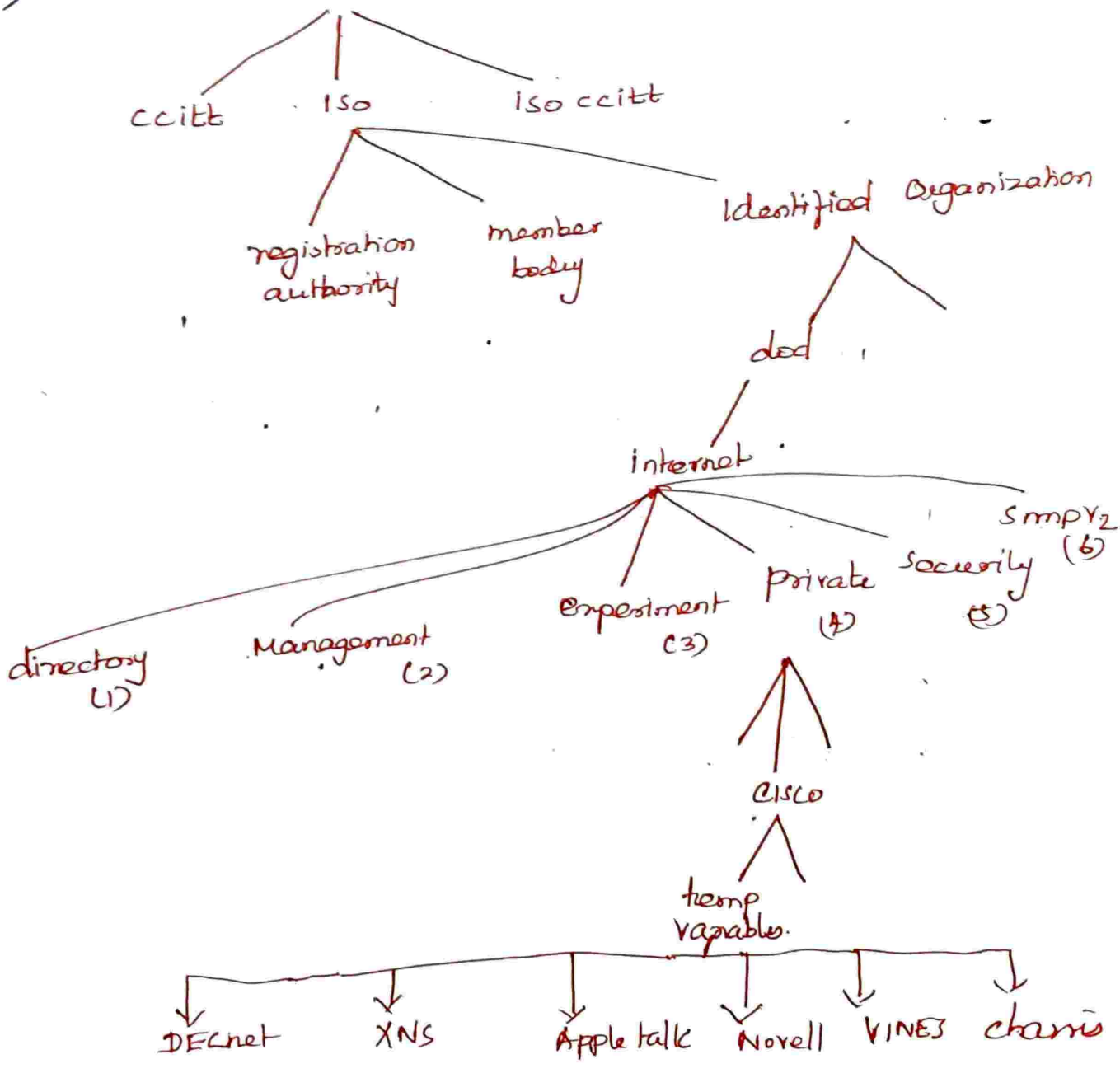
→ Tabular object define multiple related object instance that are grouped in MIB tables.

→ An Object Identifier uniquely identifies a Managed objects in the MIB hierarchy.

→ The MIB hierarchy can be depicted as a tree with a nameless root, the levels of which are assigned by different organizations.

MIB Tree

→



SNMP protocol Operation

→ SNMP is a Simple request / response protocol

→ The N/w Management system issues a request and Managed device return response.

→ This behavior is implemented by using one of four protocol operations. Get, GetNext, Set and Trap.

→ The Get operation is used by the NMS to retrieve the value of one or more object instances from an agent.

→ If the agent responding to the Get operation cannot provide values for all the object instances in a list, it does not provide any values

→ The GetNext operation is used by the NMS to retrieve the value of the next object instance in a table or a list within an agent.

→ The Set operation is used by the NMS to set the values of object instances within an agent

Disadvantages:

→ The Polling Approach used by SNMP (GET Messages) is the biggest drawback of the approach.

→ It Creates unnecessary overhead.

REAL-TIME TRANSPORT AND SESSIONS

→ The Real time transport protocol (RTP) is used for the end-to-end delivery of real time data

→ RTP is designed to be IP version and transport independent, and can be used over both UDP and TCP

→ The base RTP header provides basic features for end-to-end delivery payload-type identification, a sequence number and a time stamp.

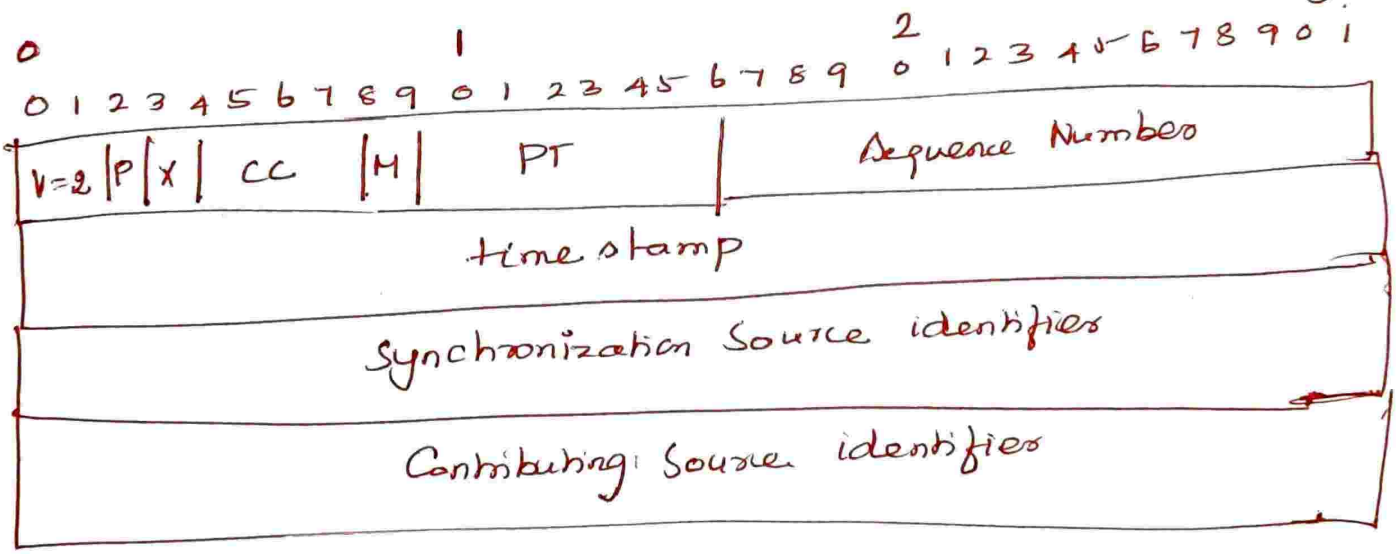
RTP header format:

RTP does not by itself provide any kind of QoS, but it is able to help deal with out-of-order packets and jitter with the

Sequence Number and timestamp fields, respectively

→ The accompanying RTP is used during an RTP session to provide feedback on the QoS of RTP data delivery, to identify the RTP source, adjust the RTCP report interval and to carry session control information.

→ RTP uses the concept of profile, which define possible additional headers features and payload formats for a particular class of application.



→ The RTP audio Video profile (AVP) specifies the profile for common audio and video Applns.

→ RTP is applicable also over 6LowPAN for the transport of real time data.

→ As RTP makes use of UDP, is IP version independent and has a fairly compact header format, it is

directly usable without modifications.

→ Although RTP can be used to delivery and monitor real time data streams, it requires that the sender and receiver somehow know about and find each other.

→ Although this may be the case in specialized embedded applications of RTP over 6LoWPAN, the automatic negotiations of real time sessions or other messaging may be very useful.

→ Session Initiation Protocol (SIP) was designed for establishing, modifying and tearing down multimedia sessions over IP.

→ SIP is widely used for Voice over IP (VoIP) Applications, and forms the backbone for the IP Multimedia System (IMS) upon which future Cellular service will be built.

→ The SIP design is similar to that of HTTP in that it uses a Human readable header format

→ SIP can be used over either UDP or TCP and can be handled by intermediate proxies, round ponder identifiers for dealing with mobility.

SIP exchanges are typically performed b/w SIP user agents

→ Typical methods include REGISTER, INVITE, ACK and BYE.

→ SIP uses a separate session description protocol to negotiate media types.

→ SIP header and body format is typically too large for efficient use over 6LoWPAN.

→ But as SIP can be applied to session setup, alarms, events and IMS integration, it has valuable use in low power embedded networks.

→ One solution for using SIP with sensor networks is TinySIP, which defined alternative messages for use with TinyOS Networking that were then mapped to SIP by a gateway Application.

INDUSTRY - SPECIFIC PROTOCOLS:

→ The industry specific application protocol that can be used over IP, and are relevant for Wireless Embedded internet application using 6LowPAN discussed in this section.

→ Building automation and energy are good examples of industries that have traditionally specified their own application protocol and formats.

→ As Communication technology have evolved, Industry - specific protocol have steadily evolved to enable use over IP.

→ Many industry - specific protocols may be used over 6LowPAN, whereas others may require the addition of compression, IPV6 support or UDP support for example.

→ Building Automation and Control Networks (BACnet)

→ konnex (KNX)

→ open Building Information Exchange (OBIX)

→ Device Language Message specification.

BACnet:

→ A data Communication protocol that is used to build an automated control N/w is known as BACnet or Building Automation control N/w.

→ The building Automation and control N/w (BACnet) standard was created by American Society of Heating Refrigeration and Air - conditioning Engg (ASHRAE) in 1995.

→ BACnet includes support for use over UDP/IP known as BACnet/IP.

→ To attain interoperability across a broad spectrum of equipment, the BACnet specification includes three major parts, primary, secondary, and teriary. So the primary part defines a technique to represent any kind of building automation apparatus in a normal way.

→ The secondary part describes messages that can be transmitted across a network of computer to check and manage such equipment. The final part describes a set of suitable LAN's which are used for conveying BACnet communications

UNIT: V TOOLS

Introduction:

On the bans of the High level application scenarios, more concrete scenarios and the resulting optimization goals of how a N/w should function are discussed.

→ It is a Commonly acknowledged truth that the properties of the transmission channel and the physical layer shape significant parts of the protocol stack.

→ The fourth good ~~at~~

Sensor Network programming challanges:

→ When applying such a model to programming networked embedded system such as sensor networks, the application programmers need to explicity deal with message passing, event Synchronization, interrupt handling and sensor reading.

→ As a result, an Application is typically implemented as a finite state Machine (FSM) that covers all

Extreme cases: Unreliable communication channels long delays, irregular arrival of messages, simultaneous events etc.

→ Real time scheduling allocates resources to make urgent tasks so that they can be finished early.

→ Event driven execution allows the sys to fall into low-power sleep mode when no interesting events need to be processed.

→ At the extreme, embedded operating sys tend to expose more hardware controls to the programmer who now have to directly face device drivers and scheduling Algorithms, and optimize code at the assembly level.

Operating Sys Design Issues:

→ Traditional operating sys are software including program that manage computing resources, control peripheral device, and provide software abstraction to the application software.

→ Traditional OS functions are therefore to manage processes, memory, CPU, time, file sys, and devices.

→ This is often implemented in a modular and layered fashion, including a lower layer of

Kernels and a higher layer of System libraries.

→ The first issue is process management and scheduling

→ The traditional OS provides process protection by allocating a separate memory space (stack) for each process.

→ Each process maintains data and information in its own space.

→ But this approach usually cause multiple data Copying and context switching between processes.

→ This is obviously not energy efficient of WSNs.

Second issue: → Memory Management

Third issue → kernal model

FSM model have been used to design micro kernals for WSN.

→ The event - driven model may save WSNs well because they look like event driven sims.

→ fourth issue: Application program Interface.

Sensor node need to provide modulay and general API for their applications.

OPERATING SYSTEM – Tiny OS.

→ The design of tiny os. allows application Software to access hardware directly when required.

→ Tiny OS is a tiny micro treaded OS that attempts to address two issues.

1. How to guarantee concurrent data flows among hardware device, and
2. How to provide modularized components with little processing and storage overhead.

→ These issues are important since Tiny OS. is required to manage hardware capabilities and resources effectively while supporting concurrent operation in an efficient manner.

→ TinyOS uses an event-based model to support high levels of concurrent application in a very small amount of memory.

→ Compared with a stack based threaded approach, which would require that stack space be reserved for each execution context, and because the switching rate of execution

→ It can rapidly create task associated with an event, with no blocking or polling.

→ When CPU is idle, the process is maintained in a sleep state to consume energy.

→ TinyOs includes a tiny scheduler and a set of components.

→ The scheduler schedules operation of those components

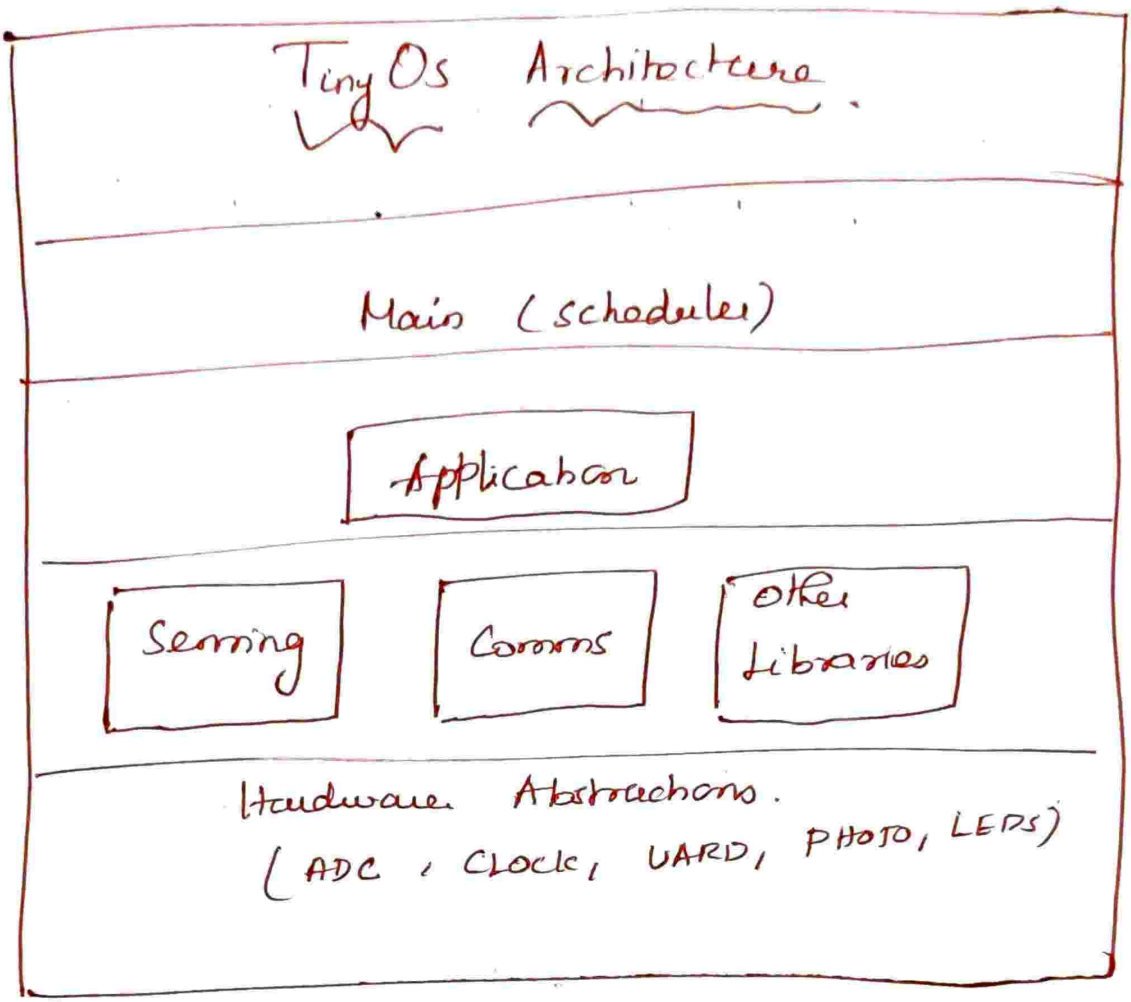
→ each component consists of four parts Command handlers, event handlers, an encapsulated fixed size frame and a group of tasks.

→ Command and task are executed in the context of the frame and operate on its state.

→ each component will declare its commands, and events to enable modularity and easy interaction with other components.

→ The current task scheduler in TinyOs is a simple FIFO mechanism whose scheduling data structure is very small. but it is power efficient since it allows a processor to sleep when the task queue is empty and while the peripheral devices are still running.

→ The frame is fixed in size and assigned statically.



→ Commands and Non blocking requests made to the low level Components

→ Therefore, Commands do not have to wait a long time to be executed.

→ A command provides feedback by returning status indicating whether it was successful

→ A Command often stores request parameters into its frame and Conditionally assign a task for later execution.

(14).

→ Tasks are a Major part of Components.

→ Like events, tasks can call low level commands, issue High level events, and assign other tasks.

→ Through groups of tasks, TinyOs can realize arbitary Computation, is an event-based model.

→ The design of components make it easy to connect various components in the form of function calls.

→ The operating system define three type of Components.

(a) Hardware Abstractions

(b) Synthetic Hardware

(c) High level Software Components.

Hardware abstraction components are lowest level Components.

→ They are actually the mapping, of physical hardware such as LEDs, a radio transceiver, and sensors,

→ each Component is mapped to a certain Hardware abstraction.

→ Synthetic Hardware Components are used to map the behavior of advanced Hardware and

often sit on the Hardware Abstraction components.

→ TinyOS design a hardware components called the Radio-frequency Module (RFM) for the radio transceivers, and a synthetic hardware component called radio byte. which handles data into or out of the underlying RFM.

Advantages of TinyOS.

→ It requires very little code and a small amount of data.

→ Events are propagated quickly and the rate of posting a task and switching the corresponding context is very high

→ It enjoys efficient Modularity.

NesC

→ nesC is a component based, event driven programming language used to build applications for the TinyOS platform

→ TinyOS is an operating s/m environment designed to run on embedded devices used in distributed wireless sensor nws.

→ The name nesC is an abbreviation of "network embedded system C"

nesC is an extension of 'C'

nesC programs are subject to whole program analysis (for safety) and optimization (for performance)

Therefore we do not consider separate compilation in nesC design.

→ The limited program size on motes makes this approach tractable.

→ nesC is a static language'

→ These restrictions make whole program analysis and optimization significantly simpler and more accurate.

→ nesC component model and parameterized interfaces eliminate many needs for dynamic memory allocation and dynamic dispatch.

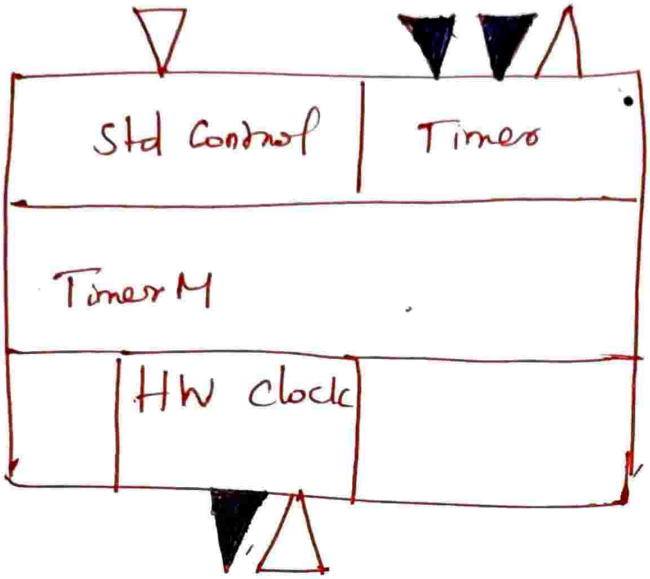
→ nesC is based on the concept of components, and directly support TinyOS's event based concurrency model

→ Additionally nesC explicitly addresses the issue of concurrent access to shared data.

→ In practice, nesC resolved many ambiguities in the

TinyOS Concepts of Components and Concurrency.

Component Specifications:




```
module TimerM {
provides {
  interface StdControl;
  interface Timer;
}
uses interface Clock as Clk;
}....
```

→ nesC applications are built by writing and assembling components.

→ A Component provides and uses interface.

→ These interfaces are the only point of access to the component.

→ An Interface generally models some service and is specified by an interface type.

→ TimerM Component, part of the TinyOS timer service that provides the StdControl and Timer Interface and uses a Clock interface.

(6).

INTERFACE

An nesC Interface is defined as the following.

Interfaces in nesC are bidirectional, They contain Commands and events both of which are essentially functions.

Syntax of an interface definition file interface Name

```
Command datatype name (datatype arg1, ... );
....
event datatype name (datatype, arg1 ...)
...
};
```

→ The timer interface defines two types of commands start and stop.

→ The timer interface fireTimer defines an event, which is also a function

→ While Commands are implemented by the providers of an interface, events are implemented by the users.

→ Similarly, all other interface in this example define both commands and events.

Besides the interface specification, components in nesC also have an implementation.

→ Modules are components implemented by application code, while Configurations are components that are implemented by connecting interfaces of existing components.

→ Every nesC Application has a top level Configuration that describes how components are wired together.

→ Functions in nesC are described as f, (i) where f is a function on an interface i;

→ Functions are invoked using the call operations and the signal operation.

→ Atomic sections are indicated with the atomic keyword, which indicates that a block of statements should be executed atomically, that is without preemption.

MODULE:

A nesC modules defines a lower level Component which can be referred by a higher level one defined by a Configuration.

→ The Syntax of a module is given as the following.

Syntax of a module definition file

```
module name
{
  uses interface name
  ...
  provides
  interface name;
  ...
  } implementation
  {
    datatype, var ... ;
  Command datatype name.name(datatype, arg...){
  // Command implementation statements
  }
  ...
  Event datatype name.name (datatype, arg...){
```

```
// command implementation statements
}
...
task void name()
{
// task statements
}
...
}
```

CONFIGURATIONS:

→ A Configuration is the other kind of Component in nesC

→ A Configuration wire Components to one another via bi-directional interfaces.

→ These wiring statements are most important, because they bring all components defined elsewhere together to be an application.

→ Each NesC application should have a Configuration which is the top level Component and specifies the starting point of its execution.

→ Modules implement Program Logic. Configuration Compose modules into larger abstractions

→ In a tinyos program, there are usually more configuration than modules

Precedence	Operator	Description	Associativity
2	++ --	prefix increment and decrement	Right to left
	+ -	Unary plus and minus	
	! ~	Logical NOT and bit wise NOT	
	type	Type cast	
	*	Indirection	
	&	Address - of	
	Size of	Size - of	
3	* / %	Multiplication division and modulus	Left to right
4	+ -	Addition & subtraction	Left to right
5	<< >>	Bitwise left shift and right shift	Left to right
6	<<=	For relational operators less than (LT) and less than or equal to (LE)	Left to right

TOSSIM:

TOSSIM (TinyOS Mote Simulator) is an open source Operating system specially developed for the wireless embedded sensor networks.

→ There are few hardware platforms available for tiny os, Some commercial and Some non-Commercial.

→ Tiny Os release includes a simulator called TOSSIM.

→ It is built especially for Berkeley Mica Mote platform.

→ TOSSIM is an emulator rather than a simulator, as it runs actual application code.

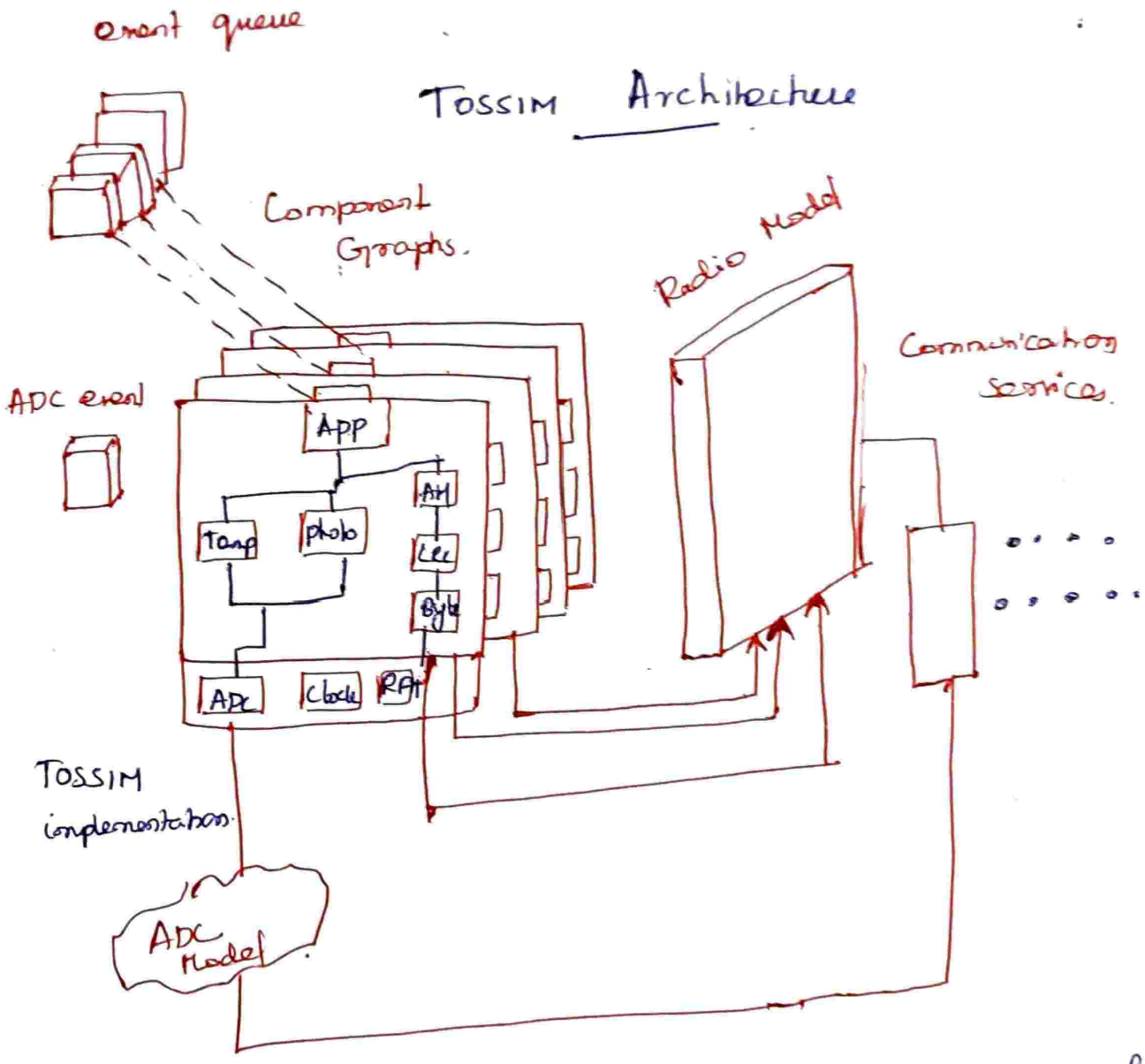
→ Simulated application code can be transformed directly to the platform but it might not run in mote as it runs in a simulation due to the simplifying assumptions in TOSSIM.

→ TOSSIM architecture is consists of five segments.

(a) frame (b) Components (c) models (d) services

(e) Events.

→ TOSSIM is a very simple but powerful emulator for WSN.



→ Each node can be evaluated under perfect transmission conditions, and using this emulator can capture the hidden terminal problems.

→ As a specific Network emulator, TOSSIM can support thousands of nodes simulations.

→ This is a very good feature, because it can more accurately simulate the real world situation.

→ Besides network, TOSSIM can emulate radio models and code executions.

→ This emulator may be provided more precise simulation result at component levels because of compiling directly to native codes.

→ TOSSIM is a bit level discrete event sim emulator built in python, a high level programming language emphasizing code readability, and c++.

→ It can run TOSSIM on Linux operating sys. or on cygwin on windows.

→ TOSSIM also provides open source and online documents

→ Developers had set four requirements for TOSSIM

① scalability
② Completeness
③ Fidelity
④ Bridging.

→ To be scalable, a simulator should manage Networks of thousands of nodes in a wide variety of configurations.

→ To achieve this, each node in TOSSIM is connected in a directed graph where each edge has a probabilistic bit error.

→ For completeness, a simulator must capture behavior of the network with a subtle timing of interactions on a mote and between motes.

→ Requirement for bridging is met as the simulated code runs directly in a real mote.

→ The goal of TOSSIM is to study the behavior of tinyOs and its applications rather than performance metrics of some new protocol.

→ Hence, it has some limitations. for instance, it does not capture energy consumption.

→ Another drawback of this framework is that every node must run the same code.

→ Therefore TOSSIM cannot be used to evaluate some type of heterogeneous Applications.

Contiki OS

→ Contiki OS is open source operating sys for resource Contraint hardware device with low power and low memory.

→ It was developed by Adam Dunkels in 2002.

→ This OS is fully GUI based system require only 30kB ROM and 10kB RAM.

→ It also provide multitasking feature and have the built in TCP/IP Suit.

→ The working environment of the WSN's are often energy limited.

→ This is one of the most important Constraint for WSN's

Communication stack:

→ WSN should have some important hardware and software features to cope with these constraints

→ Contiki OS is one of the convenient Solutions to cope with mentioned Contraint to its flexibility.

(11).

Contiki can provide Communication over IPV4, IPV6 and Rime network stack.

→ Many Contiki System are severely power Constrained.

→ Battery operated wireless sensor may need to provide years of unattended operations and with little means to recharge or replace batteries.

→ Contiki provides a set of mechanisms to reduce the power consumption of System on which it runs.

→ The default mechanism for attaining low-power operation of the radio is called Contiki MAC.

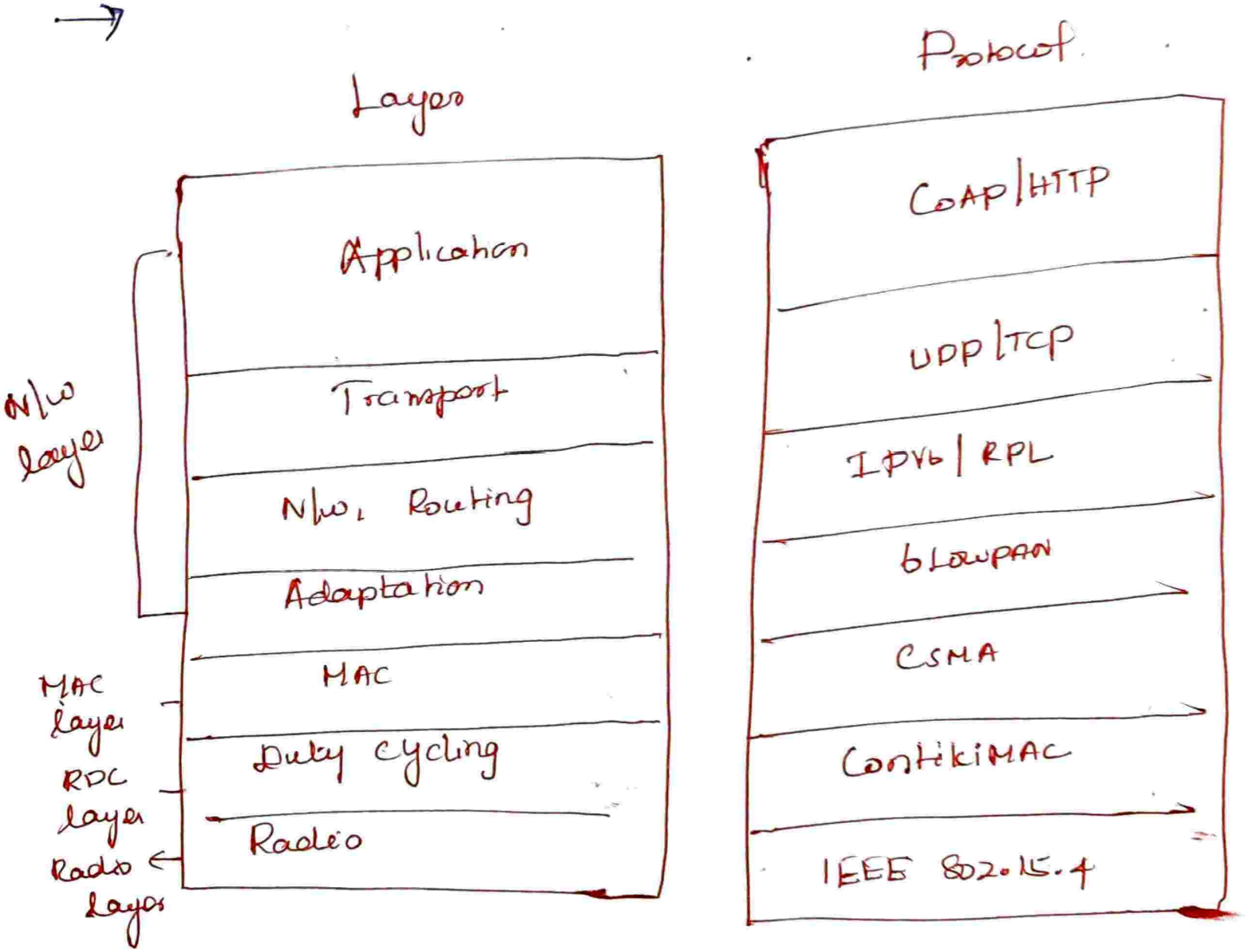
→ with Contiki MAC, nodes can be running in low-power mode and still be able to receive and relay radio messages.

→ The Contiki programming model is based on Protothreads

→ A protothread is a memory efficient Programming abstraction that shares features of both multithreading and event-driven programming to attain a low memory overhead of each protothread.

→ The kernel invokes the protothread of a process in response to an internal or external event

→



→ ContikiOs Support the resource constraint hardware has the following features:

① low power limited memory.

② Slow CPU

③ Size

→ The motes Supported by Contiki OS. are as follows, Mica Z, Wismote mote Z1 mote, Sky motes ESB mote.

→ At the kernal level it follows the event driven model, but it provides optional threading facilities to individual processes.

→ This kernel comprises of a light-weight event Scheduler that dispatches events to running-Process

→ Process execution is triggered by events dispatched by the kernel to the processes or by a polling mechanism.

→ This polling mechanism is used to avoid race Conditions.

→ Any Scheduled event will run to completion, however event handlers can use internal mechanisms for preemption.

→ Asynchronous events and Synchronous event are Supported by Contiki OS.

→ Synchronous events are dispatched immediately to the target process that causes it to be

Scheduled.

Contiki operating sys.
Application-1
Application-2.
Application-3
Application-N.
Node Management
Sensor data locator
Core updates
Sensors configuration
Network configuration
Contiki core
uIP
Loader
Proto Threads
Driver
CPU
Radio
Sensors
Oscillators
Others
Hardware
CPU
Radio
Sensors
Oscillators
Others

Contiki provides serialized access to all resources due to events run to completion and Contiki does not allow interrupt handlers to post new events.

→ Contiki provides an implementation of TCP/IP protocol stack for small 8 bit Microcontroller

→ µIP does not require its peers to have a Complete protocol stack, but it can communicate with peers running a similar lightweight stack.

→ The µIP implementation is written in 'C' and it has the minimum set of features needed for a full TCP/IP stack.

→ µIP can only support one N/w Interface, and it supports TCP, UDP, ICMP, and IP protocols

→ Support for real-time Applications is not allowed.

COOJA

→ Cooja Simulator is the efficient Simulate WSN

→ Cooja is the default simulator of Contiki operating system. that helps to simulate the WSN is addition it helps to do the performance evolution

→ Contiki is a light weight operating sys that is developed mainly for wireless nodes.

→ The motes that are developed by the Contiki offers many advantages.

→ contiki offeres a java based simulator called as Cooja which is used to simulate the wireless sensor.

→ Cooja Simulator is more flexible so that many parts of the simulator is replaceable and extendable.

Characteristics of Cooja

(a) scalability (b) Efficiency (3) Extensibility

(4) flexibility.

Contiki Cooja WSN Simulator:

Contiki Cooja is the best Simulator to simulate any wireless sensors with its own property.

→ For example

If we are designing a WSN that detects the earth quake, the sensor has its own property like lifetime, withstand ability, capacity etc.

→ We can design this WSN with the same property in Contiki Cooja.

→ When Compared to other Simulator Cooja is developed purely for Wireless Sensor Networks.

→ In addition Cooja is more flexible to change the properties of a node so that we could implement our own idea exactly.

→ Wireless sensor play important role in IOT. Where Contiki operating System was developed mainly for IoT device. Cooja is a Simulator Comes with the Contiki.

So we can use the cooja Simulator for simulating any WSN.