

INTERNET OF THINGS

VII C.S.

1-5
INTRODUCTION TO IoT

1

IMPORTANT QUESTIONS

PART-A

Q.1 What is IoT?

Ans. IoT : IoT stands for an Internet of Things. It is largely a network using which matters can talk to each other the use of the internet as an approach to communication between them. All of the matters should be IP protocol enabled on the way to have this concept viable. Not one but more than one technology are concerned to make IoT an exquisite fulfillment.

Q.2 Give examples of the impact of Internet of Things (IoT) on our lives?

Ans. With interconnected devices, you can better arrange your lifestyles and be extra productive, safer, smarter and informed than ever earlier than. As an instance how smooth it'll be that allows you to begin your day if your alarm clock is not only capable of wake you up however also able to talk together with your brewer to tell it that you are awake on the same time notifies your geezer to start water heating.

Q.3 What affects will the internet of things (IoT) have on infrastructure and smart cities region?

Ans. The competencies of the clever grid, smart homes, and its combined with IoT components in other public utilities, including roadways, sewage, and water transport can make

a contribution to greater included and useful infrastructure, particularly in towns.

Q.4 What function does the community play within the IoT?

Ans. The network plays a crucial function in the net of the entirety. It needs to provide a clever, practical, comfortable infrastructure that may scale to assist billions of context-conscious gadgets.

Q.5 How might wireless communications have an effect on the development and implementation of the internet of things (IoT)?

Ans. Both certified and unlicensed electromagnetic spectrum is vital for devices and gadgets to communicate wirelessly. For brand spanking new purposes and industries, IoT devices are being deployed & evolved, and some argue that the current day-framework for spectrum allocation might not serve those new industries well.

Q.6 How does IoT relate to the net of factors?

Ans. The "net of the whole thing" builds on the inspiration of the "net of factors" by adding community intelligence that lets in convergence, orchestration, and visibility throughout previously disparate structures.

Q.7 How is business IoT (IIoT) exclusive from the internet of things (IoT)?

Ans. Internet of Things: Everyday consumer-level gadgets related to one another and made smarter and barely self-aware.

Examples: Consumer cellular phone, smart thermostat.

Industrial Internet of Things: System and structures in industries and groups in which failures may be disastrous.

Examples: Man or woman health video display units and alert structures in hospitals.

Q.8 How will net of things (IoT) impact the sustainability of environment or business?

Ans. Internet of things (IoT) can considerably lessen carbon emissions by way of making commercial enterprise and enterprise greater green. By using road lighting fixtures greater, you may save about 40% of the energy used to lead them to run.

Q.9 What is the difference between the (IoT) and the sensor commercial enterprise?

Ans. Sensors may be utilized in masses of various approaches of IoTs which don't want to be net related. IoT also includes the managing aspect, no longer simply the sensing aspect.

Q.10 What are the main social and cultural impacts of Internet of Things (IoT)?

Ans. The IoT may also create webs of connections a good way to basically transform the manner people and things interact with every other.

Some observers have proposed that the growth of IoT will bring about a hyper-connected world wherein the seamless integration of gadgets and people will purpose the internet to vanish as a separate phenomenon. In this type of system, cyberspace and human space would seem to correctly merge right into an unmarried surrounding, with unpredictable but probably huge societal and cultural influences.

Q.11 What are the important components of an Internet of Things?

Ans. Important Components of an IoT :

1. Hardware : This will make physical items responsive and give them functionality to store records and respond to instructions.

2. Software Program : Allowing the facts collection, storage, processing, manipulating and instructing.

3. Conversation Infrastructure : Most essential of all is the communication infrastructure which consists of protocols and technologies which allow two bodily gadgets to exchange information.

Q.12 What is the difference between IoT devices and embedded devices?

Ans. Difference between IoT Devices and Embedded Devices : Internet of Things is a type of embedded system that connects to the internet. Embedded systems tend to be small software programs that implement a few functions. Internet of Things may be updated constantly according to the environment and learn by itself.

PART-B

Q.13 What are the key features of IoT?

Ans. IoT Key Features: The most important features of IoT include artificial intelligence, connectivity, sensors, active engagement, and small device use. A brief review of these features is given below:

- 1. AI :** IoT essentially makes virtually anything "smart", meaning it enhances every aspect of life with the power of data collection, artificial intelligence algorithms, and networks. This can mean something as simple as enhancing your refrigerator and cabinets to detect when milk and your favorite cereal run low, and to then place an order with your preferred grocer.
- 2. Connectivity :** New enabling technologies for networking, and specifically IoT networking, mean networks are no longer exclusively tied to major providers. Networks can exist on a much smaller and cheaper scale while still being practical. IoT creates these small networks between its system devices.
- 3. Sensors :** IoT loses its distinction without sensors. They act as defining instruments which transform IoT from a standard passive network of devices into an active system capable of real-world integration.

4. **Active Engagement** : Much of today's interaction with connected technology happens through passive engagement. IoT introduces a new paradigm for active content, product or service engagement.
5. **Small Devices** : Devices, as predicted, have become smaller, cheaper, and more powerful over time. IoT exploits purpose-built small devices to deliver its precision, scalability, and versatility.

Q.14 *State the advantages and disadvantages of Internet of Things.*

Ans. Advantages of IoT : The advantages of IoT span across every area of lifestyle and business. Here is a list of some of the advantages that IoT has to offer:

1. **Improved Customer Engagement** : Current analytics suffer from blind-spots and significant flaws in accuracy; and as noted, engagement remains passive. IoT completely transforms this to achieve richer and more effective engagement with audiences.
2. **Technology Optimization** : The same technologies and data which improve the customer experience also improve device use, and aid in more potent improvements to technology. IoT unlocks a world of critical functional and field data.
3. **Reduced Waste** : IoT makes areas of improvement clear. Current analytics give us superficial insight, but IoT provides real-world information leading to more effective management of resources.
4. **Enhanced Data Collection** : Modern data collection suffers from its limitations and its design for passive use. IoT breaks it out of those spaces, and places it exactly where humans really want to go to analyze our world. It allows an accurate picture of everything.

Disadvantages of IoT : Though IoT delivers an impressive set of benefits, it also presents a significant set of challenges. Here is a list of some its major issues:

1. **Security** : IoT creates an ecosystem of constantly connected devices communicating over networks. The system offers little control despite any security measures. This leaves users exposed to various kinds of attackers.
2. **Privacy** : The sophistication of IoT provides substantial personal data in extreme detail without the user's active participation.

3. **Complexity** : Some find IoT systems complicated in terms of design, deployment, and maintenance given their use of multiple technologies and a large set of new enabling technologies.
4. **Flexibility** : Many are concerned about the flexibility of an IoT system to integrate easily with another. They worry about finding themselves with several conflicting or locked systems.
5. **Compliance** : IoT, like any other technology in the realm of business, must comply with regulations. Its complexity makes the issue of compliance seem incredibly challenging when many consider standard software compliance a battle.

Q.15 *What is the Internet of Things (IoT)?*

Internet of Things : When people talk about the internet, they are usually referring to the electronic network that permits computers around the world to communicate with each other. What, then, is the IoT? There is no universally agreed-upon definition, but generally, the term is used to describe networks of objects that are not themselves computers but that have embedded components that connect to the internet. "Things" may include, for example, smart meters, fitness trackers, personal vehicles, home appliances, medical devices, and even clothing used by individual consumers. They may also include embedded devices in roadways and in other components of infrastructure such as electric grids, manufacturing plants and other buildings, farms, and virtually any other object, element, or system for which remote communications, control, or data collection and processing might be useful.

While fixed and mobile computing devices such as desktop computers, smartphones, and tablets are generally not considered to be IoT objects, smartphones in particular have features such as motion and position sensors that blur the distinctions. Some smartphone applications, for example, enable them to be used in fitness tracking and other health monitoring.

In other words, the IoT potentially includes huge numbers and kinds of interconnected objects. In practice, IoT refers not to a simple or uniform network of objects but rather to a complex collection of objects and networks. Specific dimensions of the IoT may be referred to by terms such as smart grid, connected cities, and industrial internet.

Other terms may also be used in the context of IoT to denote related concepts such as cyber-physical systems and the Internet of Everything.

The IoT is often considered the next major stage in the evolution of cyberspace. The first electronic computers were developed in the 1940s, but forty years passed before connecting computers through wired devices began to spread in the 1980s. The first decade of the twenty-first century saw the next stage, marked by the rapid spread of smartphones and other mobile devices that use wireless communications, as well as social media, big-data analytics, and cloud computing. Building on those advances, connections between two or more machines (M2M) and between machines and people are expected by many observers to lead to huge growth in the IoT by 2020.

Q.16 How does the IoT work?

Ans. The IoT is not separate from the internet, but rather, a potentially huge extension and expansion of it. The "things" that form the basis of the IoT are objects. They could be virtually anything – streetlights, thermostats, electric meters, fitness trackers, factory equipment, automobiles, unmanned aircraft systems (UASs or drones), or even cows or sheep in a field. What makes an object part of the IoT is embedded or attached computer, chips or similar components that give the object both a unique identifier and internet connectivity. Objects with such components are often called "smart" – such as smart meters and smart cars.

Internet connectivity allows a smart object to communicate with computers and with other smart objects. Connections of smart objects to the internet can be wired, such as through Ethernet cables, or wireless, such as via a Wi-Fi or cellular network.

To enable precise communications, each IoT object must be uniquely identifiable. That is accomplished through an Internet Protocol (IP) address, a number assigned to each internet-connected device, whether a desktop computer, a mobile phone, a printer, or an IoT object. Those IP addresses ensure that the device or object sending or receiving information is correctly identified.

Smart objects can also be involved in command networks. For example, industrial control systems can adjust manufacturing processes based on input from both other IoT objects and human operators. Network connectivity can

permit such operations to be performed in "real time" – that is, almost instantaneously.

Smart objects can form systems that communicate information and commands among themselves, usually in concert with computers they connect to. This kind of communication enables the use of smart systems in homes, vehicles, factories, and even entire cities.

Smart systems allow for automated and remote control of many processes. A smart home can permit remote control of lighting, security, HVAC (heating, ventilating, and air conditioning), and appliances. In a smart city, an intelligent transportation system (ITS) may permit vehicles to communicate with other vehicles and roadways to determine the fastest route to a destination, avoiding traffic jams, and traffic signals can be adjusted based on congestion information received from cameras and other sensors. Buildings might automatically adjust electric usage, based on information sent from remote thermometers and other sensors. An industrial internet application can permit companies to monitor production systems and adjust processes, remotely control and synchronize machinery operations, track inventory and supply chains, and perform other tasks.

IoT connections and communications can be created across a broad range of objects and networks and can transform previously independent processes into integrated systems. These integrated systems can potentially have substantial effects on homes and communities, factories and cities, and every sector of the economy, both domestically and globally.

Q.17 Write the definition and characteristics of IoT.

Ans. Definition : A dynamic global network infrastructure with self-configuring capabilities based on standard and interoperable communication protocols where physical and virtual "things" have identities, physical attributes, and virtual personalities and use intelligent interfaces, and are seamlessly integrated into the information network, often communicate data associated with users and their environments.

Let us examine this definition of IoT further to put some of the terms into perspective.

- **Dynamic and Self – Adapting:** IoT devices and systems may have the capability to dynamically adapt with the changing contexts and take actions based on their operating conditions, user's context, or sensed

environment. For example, consider a surveillance system comprising of a number of surveillance cameras. The surveillance cameras can adapt their modes (to normal or infra-red night modes) based on whether it is day or night. Cameras could switch from lower resolution to higher resolution modes when any motion is detected and alert nearby cameras to do the same. In this example, the surveillance system is adapting itself based on the context and changing (e.g., dynamic) conditions.

- **Self – Configuring:** IoT devices may have self-configuring capability, allowing a large number of devices to work together to provide certain functionality (such as weather monitoring). These devices have the ability to configure themselves in association with the IoT infrastructure, setup the networking, and fetch latest software upgrades with minimal manual or user intervention.
- **Interoperable Communication Protocols:** IoT devices may support a number of interoperable communication protocols and can communicate with other devices and also with the infrastructure.
- **Unique Identity:** Each IoT device has a unique identity and a unique identifier (such as an IP address or a URI). IoT systems may have intelligent interfaces which adapt based on the context, allow communicating with users and the environmental contexts. IoT device interfaces allow users to query the devices, monitor their status, and control them remotely, in association with the control, configuration and management infrastructure.
- **Integrated into Information Network:** IoT devices are usually integrated into the information network that allows them to communicate and exchange data with other devices and systems. IoT devices can be dynamically discovered in the network, by other devices and/or the network, and have the capability to describe themselves (and their characteristics) to other devices or user applications. For example, a weather monitoring node can describe its monitoring capabilities to another connected node so that they can communicate and exchange data. Integration into the information network helps in making IoT systems “smarter” due to the collective intelligence of the individual devices in collaboration with the infrastructure. Thus, the data from a large number of connected weather monitoring

IoT nodes can be aggregated and analyzed to predict the weather.

Q.18 Write difference between web socket and REST API.

Ans.

| Parameter | Web Socket | REST |
|-------------|--|---|
| HTTP | Utilization of HTTP happens in initial connection. | HTTP is commonly used in REST based API. |
| State | Web socket protocol is stateful, that is, the state is retained between the message transfers. | The stateless HTTP is used for all the requests. No data is retained between two requests. On each request, fresh data reaches the server, which is deleted after the response is made. |
| Connection | Bi-directional | Uni-directional |
| Uses | Real-time applications. | Request-response based applications. |
| Cost | Less communication cost. | Overhead is more as compared to web socket based API. |
| Performance | Better in applications with more communication. | Better in applications where the communication is occasional. |
| Dependency | The IP address and port number | Based completely on HTTP. |

For IOT Levels ,watch on my YouTube channel!

Q.19 Explain IoT level-1 with neat diagram.

Ans. IoT Level-1: A level-1 IoT system has a single node/device that performs sensing and/or actuation, stores data, performs analysis and hosts the application as shown in Figure. Level-1 IoT systems are suitable for modeling low-cost and low-complexity solutions where the data involved is not big and the analysis requirements are not computationally intensive.

Let us now consider an example of a level-1 IoT system for home automation. The system consists of a single node that allows controlling the lights and appliances in a home remotely. The device used in this system interfaces with the lights and appliances using electronic relay switches. The

status information of each light or appliance is maintained in a local database. REST services deployed locally allow retrieving and updating the state of each light or appliance in the status database. The controller service continuously monitors the state of each light or appliance (by retrieving state from the database) and triggers the relay switches accordingly. The application which is deployed locally has a user interface for controlling the lights or appliances. Since the device is connected to the internet, the application can be accessed remotely as well.

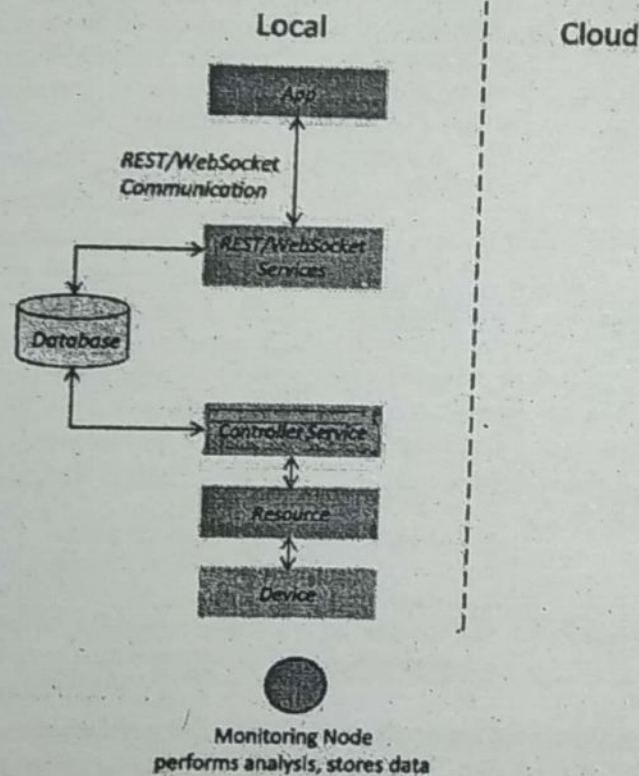


Fig. : IoT Level-1

Q.20 Write short note on IoT level-2.

Ans. IoT Level-2: A level-2 IoT system has a single node that performs sensing and/or actuation and local analysis as shown in Figure. Data is stored in the cloud and application is usually cloud-based. Level-2 IoT systems are suitable for solutions where the data involved is big, however, the primary analysis requirement is not computationally intensive and can be done locally itself.

Let us consider an example of a level-2 IoT system for smart irrigation. The system consists of a single node that

monitors the soil moisture level and controls the irrigation system. The device used in this system collects soil moisture data from sensors. The controller service continuously monitors the moisture levels. If the moisture level drops below a threshold, the irrigation system is turned on. For controlling the irrigation system actuators such as solenoid valves can be used. The controller also sends the moisture data to the computing cloud. A cloud-based REST web service is used for storing and retrieving moisture data which is stored in the cloud database. A cloud-based application is used for visualizing the moisture levels over a period of time, which can help in making decisions about irrigation schedules.

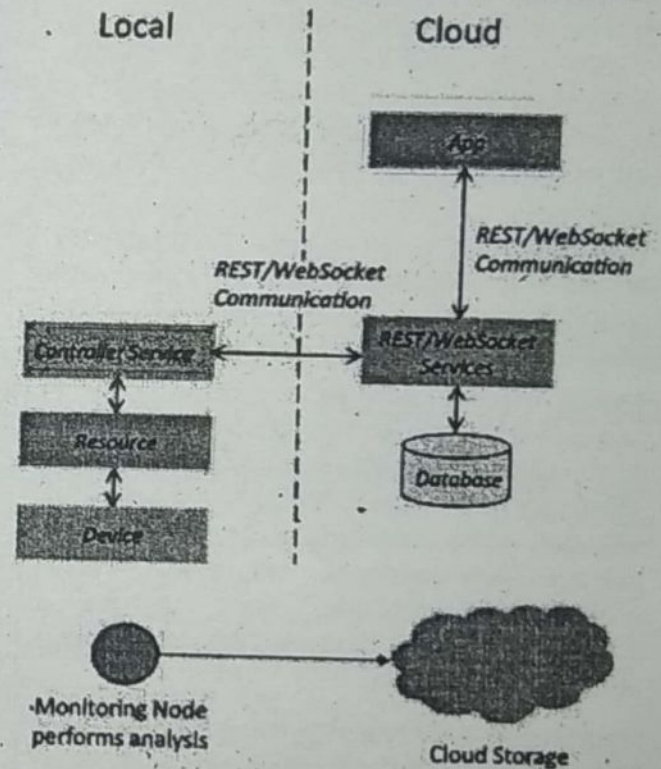


Fig. : IoT Level-2

Q.21 What do you understand by IOT level-3? Explain with figure.

Ans. IoT Level-3: A level-3 IoT system has a single node. Data is stored and analyzed in the cloud and application is cloud-based as shown in Figure. Level-3 IoT systems are suitable for solutions where the data involved is big and the analysis requirements are computationally intensive.

Let us consider an example of a level-3 IoT system for tracking package handling. The system consists of a single node (for

a package) that monitors the vibration levels for a package being shipped. The device in this system uses accelerometer and gyroscope sensors for monitoring vibration levels. The controller service sends the sensor data to the cloud in real-time using a WebSocket service. The data is stored in the cloud and also visualized using a cloud-based application. The analysis components in the cloud can trigger alerts if the vibration levels become greater than a threshold. The benefit of using WebSocket service instead of REST service in this example is that, the sensor data can be sent in real time to the cloud. Moreover, cloud based applications can subscribe to the sensor data feeds for viewing the real-time data.

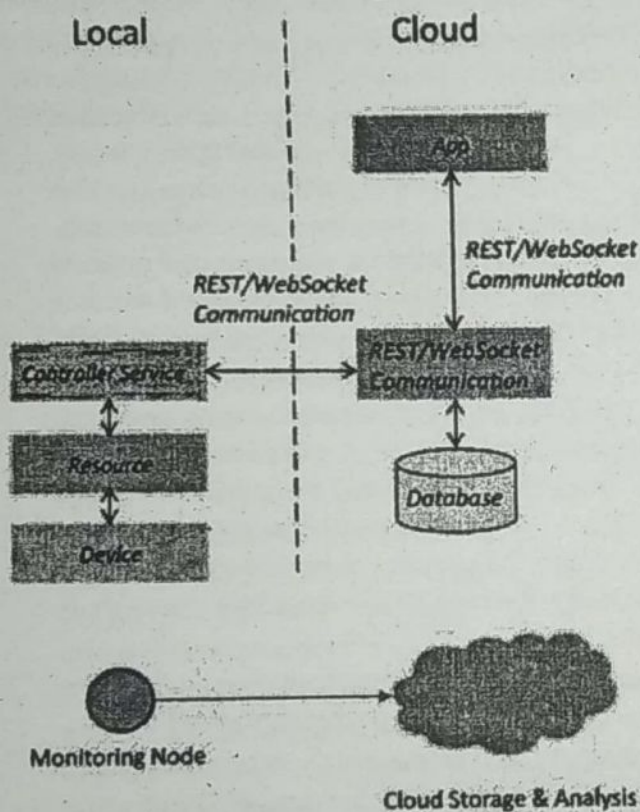


Fig. : IoT Level-3

Q.22 Write short note on IoT level-4.

Ans. IoT Level-4: A level-4 IoT system has multiple nodes that perform local analysis. Data is stored in the cloud and application is cloud-based as shown in Figure. Level-4 contains local and cloud-based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices, observer nodes can process information and use it for various applications, however, observer nodes do not perform any control functions. Level-4 IoT systems are

suitable for solutions where multiple nodes are required, the data involved is big and the analysis requirements are computationally intensive.

Let us consider an example of a level-4 IoT system for noise monitoring. The system consists of multiple nodes placed in different locations for monitoring noise levels in an area. The nodes in this example are equipped with sound sensors. Nodes are independent of each other. Each node runs its own controller service that sends the data to the cloud. The data is stored in a cloud database. The analysis of data collected from a number of nodes is done in the cloud. A cloud-based application is used for visualizing the aggregated data.

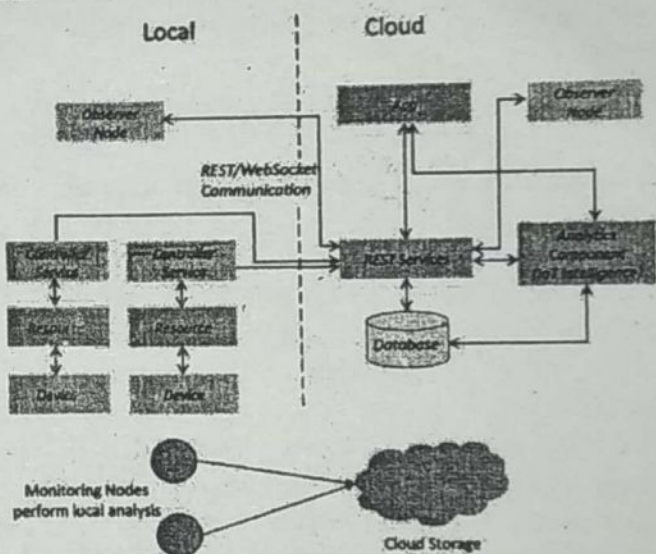


Fig. : IoT Level-4

PART-C

Q.23 What impacts will the IoT have on various sectors? Explain in detail.

Ans. The IoT may significantly affect many aspects of the economy and society, although the full extent and nature of its eventual impacts remains uncertain. Many observers predict that the growth of the IoT will bring positive benefits through enhanced integration, efficiency, and productivity across many sectors of the U.S. and global economies. Among those commonly mentioned are agriculture, energy, health care, manufacturing and transportation. Significant impacts may also be felt more broadly on economic growth,

infrastructure and cities and individual consumers. However, both policy and technical challenges, including security and privacy issues, might inhibit the growth and impact of IoT innovations.

Economic Growth: Several economic analyses have predicted that the IoT will contribute significantly to economic growth over the next decade, but the predictions vary substantially in magnitude. The current global IoT market has been valued at about \$2 trillion, with estimates of its predicted value over the next five to ten years varying from \$4 trillion to \$11 trillion. Such variability demonstrates the difficulty of making economic forecasts in the face of various uncertainties, including a lack of consensus among researchers about exactly what the IoT is and how it will develop.

Economic Sectors

Agriculture: The IoT can be leveraged by the agriculture industry through precision agriculture, with the goal of optimizing production and efficiency while reducing costs and environmental impacts. For farming operations, it involves analysis of detailed, often real-time data on weather, soil and air quality, water supply, pest populations, crop maturity, and other factors such as the cost and availability of equipment and labor. Field sensors test soil moisture and chemical balance, which can be coupled with location technologies to enable precise irrigation and fertilization. Drones and satellites can be used to take detailed images of fields, giving farmers information about crop yield, nutrient deficiencies and weed locations. For ranching and animal operations, radio frequency identification (RFID) chips and electronic identification readers (EID) help monitor animal movements, feeding patterns, and breeding capabilities, while maintaining detailed records on individual animals.

Energy: Within the energy sector, the IoT may impact both production and delivery, for example through facilitating monitoring of oil wellheads and pipelines. When IoT components are embedded into parts of the electrical grid, the resulting infrastructure is commonly referred to as the "smart grid." This use of IoT enables greater control by utilities over the flow of electricity and can enhance the efficiency of grid operations." It can also expedite the integration of microgenerators into the grid.

Smart-grid technology can also provide consumers with greater knowledge and control of their energy usage through the use of smart meters in the home or office. Connection of smart meters to a building's HVAC, lighting and other systems

can result in "smart buildings" that integrate the operation of those systems. Smart buildings use sensors and other data to automatically adjust room temperatures, lighting and overall energy usage, resulting in greater efficiency and lower energy cost. Information from adjacent buildings may be further integrated to provide additional efficiencies in a neighborhood or larger division in a city.

Health Care: The IoT has many applications in the health care field, in both health monitoring and treatment, including telemedicine and telehealth. Applications may involve the use of medical technology and the internet to provide long-distance health care and education. Medical devices which can be wearable or nonwearable, or even implantable, injectable, or ingestible can permit remote tracking of a patient's vital signs, chronic conditions, or other indicators of health and wellness. Wireless medical devices may be used not only in hospital settings but also in remote monitoring and care, freeing patients from sustained or recurring hospital visits. Some experts have stated that advances in healthcare IoT applications will be important for providing affordable, quality care to the aging U.S. population.

Manufacturing: Integration of IoT technologies into manufacturing and supply chain logistics is predicted to have a transformative effect on the sector. The biggest impact may be realized in optimization of operations, making manufacturing processes more efficient. Efficiencies can be achieved by connecting components of factories to optimize production, but also by connecting components of inventory and shipping for supply chain optimization. Another application is predictive maintenance, which uses sensors to monitor machinery and factory infrastructure for damage. Resulting data can enable maintenance crews to replace parts before potentially dangerous and/or costly malfunctions occur.

Transportation: Transportation systems are becoming increasingly connected. New motor vehicles are equipped with features such as global positioning systems (GPS) and in-vehicle entertainment, as well as advanced driver assistance systems (ADAS), which utilize sensors in the vehicle to assist the driver, for example with parking and emergency braking. Further connection of vehicle systems enables fully autonomous or self-driving automobiles, which are predicted to be commercialized in the next 5 – 20 years.

Additionally, IoT technologies can allow vehicles within and across modes including cars, buses, trains, airplanes, and unmanned aerial vehicles (drones to "talk" to one another and to components of the IoT infrastructure, creating intelligent

transportation systems (ITS). Potential benefits of ITS may include increased safety and collision avoidance, optimized traffic flows and energy savings among others.

Infrastructure and Smart Cities: The capabilities of the smart grid, smart buildings and ITS combined with IoT components in other public utilities such as roadways, sewage and water transport and treatment, public transportation and waste removal can contribute to more integrated and functional infrastructure, especially in cities. For example, traffic authorities can use cameras and embedded sensors to manage traffic flow and help reduce congestion. IoT components embedded in street lights or other infrastructure elements can provide functions such as advanced lighting control, environmental monitoring and even assistance for drivers in finding parking spaces. Smart garbage cans can signal waste removal teams when they are full streamlining the routes that garbage trucks take.

This integration of infrastructure and service components is increasingly referred to as smart cities, or other terms such as connected, digital, or intelligent cities or communities. A number of cities in the United States and elsewhere have developed smart-city initiatives.

As with IoT and other popular technology terms, there is no established consensus definition or set of criteria for characterizing what a smart city is. Specific characterizations vary widely, but in general they involve the use of IoT and related technologies to improve energy, transportation, governance, and other municipal services for specified goals such as sustainability or improved quality of life. The related technologies include:

- (i) Social media (such as Facebook and Twitter)
- (ii) Mobile computing (such as smartphones and wearable devices)
- (iii) Data analytics (big data – the processing and use of very large data sets, and open data – databases that are publicly accessible)
- (iv) Cloud computing (the delivery of computing services from a remote location, analogous to the way utilities such as electricity are provided)

Together, these are sometimes called SMAC.

Social and Cultural Impacts: The IoT may create webs of connections that will fundamentally transform the way people and things interact with each other. The emerging cyberspace platform created by the IoT and SMAC has been described as potentially making cities “like ‘computers’ in open air,”

where citizens engage with the city “in a real-time and ongoing loop of information”.

Some observers have proposed that the growth of IoT will result in a hyperconnected world in which the seamless integration of objects and people will cause the internet to disappear as a separate phenomenon. In such a world, cyberspace and human space would seem to effectively merge into a single environment, with unpredictable but potentially substantial societal and cultural impacts.

Q.24 What issues might affect the development and implementation of the IoT?

Ans. The Internet of Things is often lauded for its potentially revolutionary applications. Indeed, IoT devices are today being implemented in many different sectors for a vast array of purposes. However, it is still unclear how IoT will progress due to challenges associated with both technical and policy issues.

Technical Issues: Prominent technical limitations that may affect the growth and use of the IoT include a lack of new internet addresses under the most widely used protocol, the availability of high-speed and wireless communications, and lack of consensus on technical standards.

Internet Addresses: A potential barrier to the development of IoT is the technical limitations of the version of the Internet Protocol (IP) that is used most widely. IP is the set of rules that computers use to send and receive information via the internet, including the unique address that each connected device or object must have to communicate. Version 4 (IPv4) is currently in widest use. It can accommodate about four billion addresses, and it is close to saturation, with few new addresses available in many parts of the world.

Some observers predict that internet traffic will grow faster for IoT objects than any other kind of device over the next five years, with more than 25 billion IoT objects in use by 2020, and perhaps 50 billion devices altogether. IPv4 appears unlikely to meet that growing demand, even with the use of workarounds such as methods for sharing IP addresses.

Version 6 (IPv6) allows for a huge increase in the number IP addresses. With IPv4, the maximum number of unique addresses, 4.2 billion, is not enough to provide even one address for each of the 7.3 billion people on Earth. IPv6, in contrast, will accommodate over 10 addresses more than a trillion trillion per person.

It is highly likely that to accommodate the anticipated growth in the numbers of internet connected objects, IPv6 will have to be implemented broadly. It has been available since 1999 but was not formally launched until 2012. In most countries, fewer than 10% of IP addresses were in IPv6 as of September 2015. Adoption is highest in some European countries and in the United States, where adoption has doubled in the past year to about 20%. Globally, adoption has doubled annually since 2011, to about 7% of addresses in mid-2015. While growth in adoption is expected to continue, it is not yet clear whether the rate of growth will be sufficient to accommodate the expected growth in the IoT. That will depend on a number of factors, including replacement of some older systems and applications that cannot handle IPv6 addresses, resolution of security issues associated with the transition, and availability of sufficient resources for deployment.

Efforts to transition federal systems to IPv6 began more than a decade ago. According to estimates by NIST, adoption for public-facing services has been much greater within the federal government than within industry or academia. However, adoption varies substantially among agencies, and some data suggest that federal adoption plateaued in 2012. Data were not available for this report on domains that are not public – facing, and it is not clear whether adoption of IPv6 by federal agencies will affect their deployment of IoT applications.

High-Speed Internet: Use and growth of the IoT can also be limited by the availability of access to high-speed internet and advanced telecommunications services, commonly known as broadband, on which it depends. While many urban and suburban areas have access, that is not the case for many rural areas, for which private sector providers may not find establishment of the required infrastructure profitable, and government programs may be limited.

Wireless Communications: Many observers believe that issues relating to access to the electromagnetic spectrum will need to be resolved to ensure the functionality and interoperability of IoT devices. Access to spectrum, both licensed and unlicensed, is essential for devices and objects to communicate wirelessly. IoT devices are being developed and deployed for new purposes and industries, and some argue that the current framework for spectrum allocation may not serve these new industries well.

Standards: Currently, there is no single universally recognized set of technical standards for the IoT, especially with respect to communications, or even a commonly accepted definition

among the various organizations that have produced IoT standards or related documents. Many observers agree that a common set of standards will be essential for interoperability and scalability of devices and systems. However, others have expressed pessimism that a universal standard is feasible or even desirable, given the diversity of objects that the IoT potentially encompasses. Several different sets of de facto standards have been in development, and some observers do not expect formal standards to appear before 2017. Whether conflicts between standards will affect growth of the sector as it did for some other technologies is not clear.

Other Technical Issues: Several other technical issues might impact the development and adoption of IoT. For example, if an object's software cannot be readily updated in a secure manner, that could affect both function and security. Some observers have therefore recommended that smart objects have remote updating capabilities. However, such capabilities could have undesirable effects such as increasing power requirements of IoT objects or requiring additional security features to counter the risk of exploitation by hackers of the update features.

Energy consumption can also be an issue. IoT objects need energy for sensing, processing and communicating information. If objects isolated from the electric grid must rely on batteries, replacement can be a problem, even if energy consumption is highly efficient. That is especially the case for applications using large numbers of objects or placements that are difficult to access. Therefore, alternative approaches such as energy harvesting, whether from solar or other sources, are being developed.

Cybersecurity: The security of devices and the data they acquire, process, and transmit is often cited as a top concern in cyberspace. Cyberattacks can result in theft of data and sometimes even physical destruction. Some sources estimate losses from cyberattacks in general to be very large in the hundreds of billions or even trillions of dollars. As the number of connected objects in the IoT grows, so will the potential risk of successful intrusions and increases in costs from those incidents.

Cybersecurity involves protecting information systems, their components and contents, and the networks that connect them from intrusions or attacks involving theft, disruption, damage, or other unauthorized or wrongful actions. IoT objects are potentially vulnerable targets for hackers. Economic and other factors may reduce the degree to which such objects are designed with adequate cybersecurity

capabilities built in. IoT devices are small, are often built to be disposable, and may have limited capacity for software updates to address vulnerabilities that come to light after deployment.

The interconnectivity of IoT devices may also provide entry points through which hackers can access other parts of a network. For example, a hacker might gain access first to a building thermostat, and subsequently to security cameras or computers connected to the same network, permitting access to and exfiltration or modification of surveillance footage or other information. Control of a set of smart objects could permit hackers to use their computing power in malicious networks called botnets to perform various kinds of cyberattacks.

Access could also be used for destruction, such as by modifying the operation of industrial control systems, as with the Stuxnet malware that caused centrifuges to self-destruct at Iranian nuclear plants. Among other things, Stuxnet showed that smart objects can be hacked even if they are not connected to the internet. The growth of smart weapons and other connected objects within DOD has led to growing concerns about their vulnerabilities to cyberattack and increasing attempts to prevent and mitigate such attacks, including improved design of IoT objects. Cybersecurity for the IoT may be complicated by factors such as the complexity of networks and the need to automate many functions that can affect security, such as authentication. Consequently, new approaches to security may be needed for the IoT.

IoT cybersecurity will also likely vary among economic sectors and subsectors, given their different characteristics and requirements. Each sector will have a role in developing cybersecurity best practices, unique to its needs. The federal government has a role in securing federal information systems, as well as assisting with security of nonfederal systems, especially critical infrastructure. Cybersecurity legislation considered in the 114th Congress, while not focusing specifically on the IoT, would address several issues that are potentially relevant to IoT applications, such as information sharing and notification of data breaches.

Safety: Given that smart objects can be used both to monitor conditions and to control machinery, the IoT has broad implications for safety; with respect to both improvements and risks. For example, objects embedded in pipelines can monitor both the condition of the equipment and the flow of contents. Among other benefits, that can help both to expedite shutoffs in the event of leaks and to prevent them through

predictive maintenance. Connected vehicles can help reduce vehicle collisions through crash avoidance technologies and other applications." Wireless medical devices can improve patient safety by permitting remote monitoring and facilitating adjustments in care.

However, given the complexities involved in some applications of IoT, malfunctions might in some instances result in catastrophic system failures, creating significant safety risks, such as flooding from dams or levees. In addition, hackers could potentially cause malfunctions of devices such as insulin pumps or automobiles, potentially creating significant safety risks.

Privacy: Cyberattacks may also compromise privacy, resulting in access to and exfiltration of identifying or other sensitive information about an individual. For example, an intrusion into a wearable device might permit exfiltration of information about the location, activities, or even the health of the wearer.

In addition to the question of whether security measures are adequate to prevent such intrusions, privacy concerns also include questions about the ownership, processing, and use of such data. With an increasing number of IoT objects being deployed, large amounts of information about individuals and organizations may be created and stored by both private entities and governments.

With respect to government data collection, the U.S. Supreme Court has been reticent about making broad pronouncements concerning society's expectations of privacy under the Fourth Amendment of the Constitution while new technologies are in flux, as reflected in opinions over the last five years. Congress may also update certain laws, such as the Electronic Communications Privacy Act of 1986, given the ways that privacy expectations of the public are evolving in response to IoT and other new technologies. IoT applications may also create challenges for interpretation of other laws relating to privacy, such as the Health Insurance Portability and Accountability Act and various state laws, as well as established practices such as those arising from norms such as the Fair Information Practice Principles.

Other Policy Issues

Federal Role: As described in the section, "What Is the Current Federal Role?" many federal agencies are involved in different aspects of the IoT. Some business representatives and others have stressed the role of effective public/private partnerships in the development of this technology space.

IoT.12

However, observers have also expressed concerns about the role of government regulations and policy, as discussed further in sections below, and about the degree and effectiveness of coordination among the involved federal agencies. Concerns of some extend beyond the federal role to that of state, local, and foreign governments.

Given the eclectic nature of the IoT, overall coordination of federal efforts may be challenging with respect to identification of both the goals of coordination and the methods for achieving them. Nevertheless, several observers have argued in favor of a national strategy for the IoT, including in resolutions considered in the 114th Congress.

Some interagency initiatives have been established with respect to specific aspects of the IoT. For example, in addition to the R&D coordination activities for cyber-physical systems under the NITRD program, a specific framework has been developed for smart cities as part of the overall White House initiative involving several federal agencies, local governments, and the private sector.

Q.25 Explain physical design of IoT in detail.

Ans. Things in IoT : The "Things" in IoT usually refers to IoT devices which have unique identities and can perform remote sensing, actuating and monitoring capabilities. IoT devices can exchange data with other connected devices and applications (directly or indirectly), or collect data from other devices and process the data either locally or send the data to centralized servers or cloud-based application back-ends for processing the data, or perform some tasks locally and other tasks within the IoT infrastructure, based on temporal and space constraints (i.e., memory, processing capabilities, communication latencies and speeds, and deadlines).

Figure 1 shows a block diagram of a typical IoT device. An IoT device may consist of several interfaces for connections to other devices, both wired and wireless. These include (i) I/O interfaces for sensors (ii) interfaces for internet connectivity (iii) memory and storage interfaces and (iv) audio/video interfaces. An IoT device can collect various types of data from the on-board or attached sensors, such as temperature, humidity, light intensity. The sensed data can be communicated either to other devices or cloud-based servers/storage. IoT devices can be connected to actuators that allow them to interact with other physical entities (including non-IoT devices and systems) in the vicinity of the device. For example, a relay switch connected to an IoT device can turn

an appliance on/off based on the commands sent to the IoT device over the internet.

IoT devices can also be of varied types, for instance, wearable sensors, smart watches, LED lights, automobiles and industrial machines. Almost all IoT devices generate data in some form or the other which when processed by data analytics systems leads to useful information to guide further actions locally or remotely. For instance, sensor data generated by a soil moisture monitoring device in a garden, when processed can help in determining the optimum watering schedules. Figure shows different types of IoT devices.

IoT Protocols

1. Link Layer: Link layer protocols determine how the data is physically sent over the network's physical layer or medium (e.g., copper wire, coaxial cable, or a radio wave). The scope of the link layer is the local network connection to which host is attached. Hosts on the same link exchange data packets over the link layer using link layer protocols. Link layer determines how the packets are coded and signaled by the hardware device over the medium to which the host is attached (such as a coaxial cable). Let us now look at some link layer protocols which are relevant in the context of IoT.

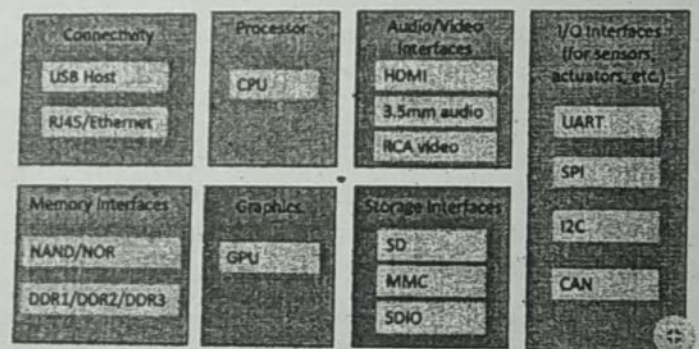
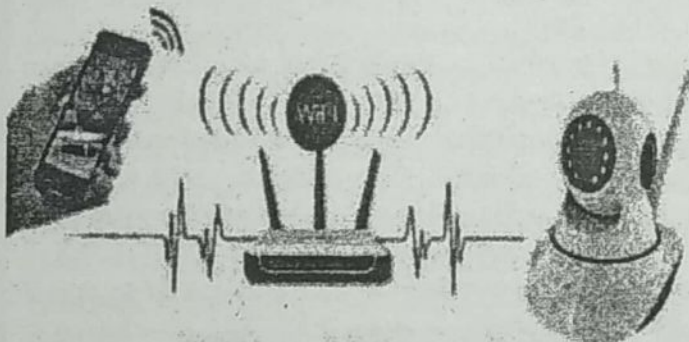


Fig. 1

- (i) **802.3 – Ethernet :** IEEE 802.3 is a collection of wired Ethernet standards for the link layer. For example, 802.3 is the standard for 10BASE5 Ethernet that uses coaxial cable as a shared medium, 802.3.i is the standard for 10BASE-T Ethernet over copper twisted-pair connections, 802.3.j is the standard for 10BASE-F Ethernet over fiber optic connections, 802.3ae is the standard for 10 Gbit/s Ethernet over fiber, and so on. These standards provide data rates from 10 Mb/s to 40 Gb/s and higher. The shared medium in Ethernet can be a coaxial cable, twisted-pair wire or an optical fiber. The shared medium (i.e., broadcast medium) carries the communication for all the devices on the

network, thus data sent by one device can be received by all devices subject to propagation conditions and transceiver capabilities. The specifications of the 802.3 standards are available on the IEEE 802.3 working group website.

- (ii) **802.11 – WiFi:** IEEE 802.11 is a collection of wireless local area network (WLAN) communication standards, including extensive description of the link layer. For example, 802.11a operates in the 5 GHz band, 802.11b and 802.11g operate in the 2.4 GHz band, 802.11n operates in the 2.4/5 GHz bands, 802.11ac operates in the 5 GHz band and 802.11ad operates in the 60 GHz band. These standards provide data rates from 1 Mb/s to up to 6.75 Gb/s. The specifications of the 802.11 standards are available on the IEEE 802.11 working group website.
- (iii) **802.16 – WiMax:** IEEE 802.16 is a collection of wireless broadband standards, including extensive descriptions for the link layer (also called WiMax). WiMax standards provide data rates from 1.5 Mb/s to 1 Gb/s. The recent update (802.16m) provides data rates of 100 Mbit/s for mobile stations and 1 Gbit/s for fixed stations.



PTZ Camera 355 degree rotation

Sliding your finger to monitor without blind angles

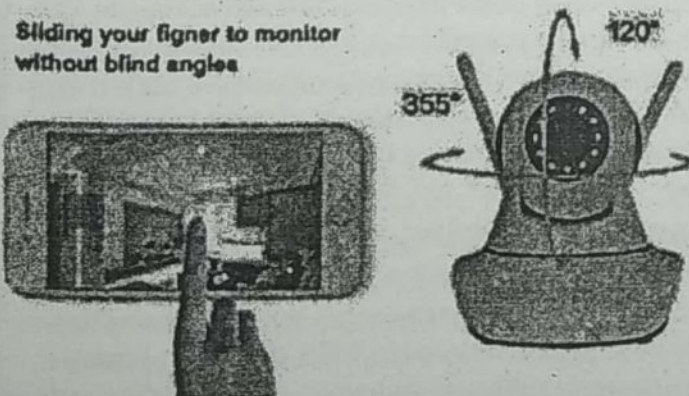


Fig. 2 : Smart Camera

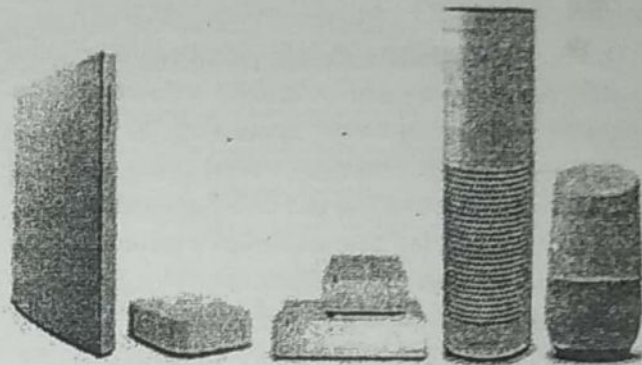


Fig. 3 : Smart hubs in IoT

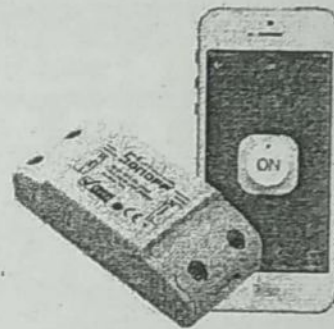


Fig. 4 : IoT Switch



Fig. 5 : Smart TV Environment

The specifications of the 802.11 standards are readily available on the IEEE 802.16 working group website.

- (iv) **802.15.4-LR-WPAN:** IEEE 802.15.4 is a collection of standards for Low-rate wireless personal area networks (LR-WPANs). These standards form the basis of specifications for high level communication protocols such as ZigBee. LR-WPAN standards provide data rates from 40 Kb/s to 250 Kb/s. These standards provide low-cost and low-speed communication for power constrained devices. The

specifications of the 802.15.4 standards are available on the IEEE 802.15 working group website.

- (v) **2G/3G/4G - Mobile Communication:** There are different generations of mobile communication standards including second generation (2G including GSM and CDMA), third generation (3G – including UMTS and CDMA2000) and fourth generation (4G – including LTE), IoT devices based on these standards can communicate over cellular networks. Data rates range from 9.6 Kb/s (for 2G) to upto 100 Mb/s (for 4G) and are available from the 3GPP websites.

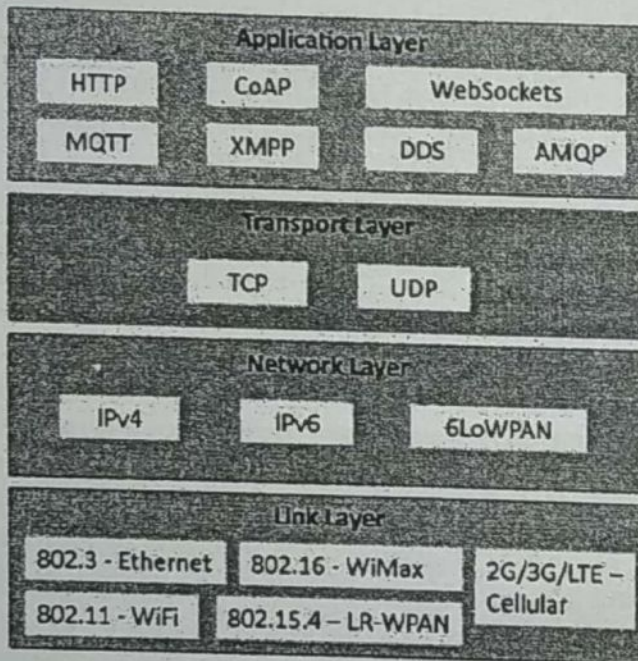


Fig. 6

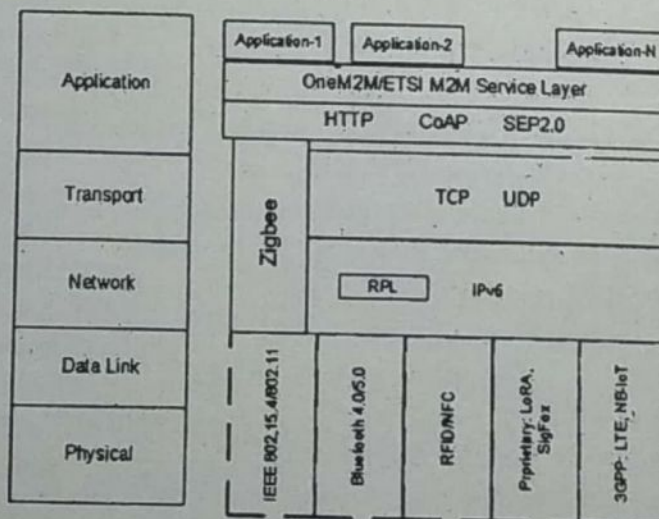


Fig. 7

2. Network/Internet Layer: The network layers are responsible for sending of IP datagrams from the source network to the destination network. This layer performs the host addressing and packet routing. The datagrams contain the source and destination addresses which are used to route them from the source to destination across multiple networks. Host identification is done using hierarchical IP addressing schemes such as IPv4 or IPv6.

- (i) **IPv4:** Internet Protocol version 4 (IPv4) is the most deployed Internet Protocol that is used to identify the devices on a network using a hierarchical addressing scheme. IPv4 uses a 32-bit address scheme that allows total of 2^{32} or 4,294,967,296 addresses. As more and more devices got connected to the internet, these addresses got exhausted in the year 2011. IPv4 has been succeeded by IPv6. The IP protocols establish connections on packet networks, but do not guarantee delivery of packets. Guaranteed delivery and data integrity are handled by the upper layer protocols (such as TCP). IPv4 is formally described in RFC 791.
- (ii) **IPv6:** Internet Protocol version 6 (IPv6) is the newest version of Internet Protocol and successor to IPv4, IPv6 uses 128-bit address scheme that allows total of 2^{128} or 3.4×10^{38} addresses. IPv6 is formally described in RFC 2460.
- (iii) **6LoWPAN :** 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks) brings IP protocol to the low-power devices which have limited processing capability. 6LoWPAN operates in the 2.4 GHz frequency range and provides data transfer rates of 250 Kb/s. 6LoWPAN works with the 802.15.4 link layer protocol and defines compression mechanisms for IPv6 datagrams over IEEE 802.15.4-based networks.

3. Transport Layer: The transport layer protocols provide end-to-end message transfer capability independent of the underlying network. The message transfer capability can be set up on connections, either using handshakes (as in TCP) or without handshakes/acknowledgements (as in UDP). The transport layer provides functions such as error control, segmentation, flow control and congestion control.

- (i) **TCP:** Transmission Control Protocol (TCP) is the most widely used transport layer protocol, that is used by web browsers (along with HTTP, HTTPS application layer protocols), email programs (SMTP application

layer protocol) and file transfer protocol (FTP). TCP is a connection oriented and stateful protocol. While IP protocol deals with sending packets, TCP ensures reliable transmission of packets in-order. TCP also provides error detection capability so that duplicate packets can be discarded and lost packets are retransmitted. The flow control capability of TCP ensures that rate at which the sender sends the data is not too high for the receiver to process. The congestion control capability of TCP helps in avoiding network congestion and congestion collapse which can lead to degradation of network performance. TCP is described in RFC 793.

- (ii) **UDP:** Unlike TCP, which requires carrying out an initial setup procedure, UDP is a connectionless protocol. UDP is useful for time-sensitive applications that have very small data units to exchange and do not want the overhead of connection setup. UDP is a transaction oriented and stateless protocol. UDP does not provide guaranteed delivery, ordering of messages and duplicate elimination. Higher levels of protocols can ensure reliable delivery or ensuring connections created are reliable. UDP is described in RFC 768.

4. Application Layer: Application layer protocols define how the applications interface with the lower layer protocols to send the data over the network. The application data, typically in files, is encoded by the application layer protocol and encapsulated in the transport layer protocol which provides connection or transaction oriented communication over the network. Port numbers are used for application addressing (for example port 80 for HTTP, port 22 for SSH, etc.). Application layer protocols enable process-to-process connections using ports.

- (i) **HTTP:** Hypertext Transfer Protocol (HTTP) is the application layer protocol that forms the foundation of the World Wide Web (WWW). HTTP includes commands such as GET, PUT, POST, DELETE, HEAD, TRACE, OPTIONS, etc. The protocol follows a request-response model where a client sends requests to a server using the HTTP commands. HTTP is a stateless protocol and each HTTP request is independent of the other requests. An HTTP client can be a browser or an application running on the client (e.g., an application running on an IoT device, a mobile application or other software). HTTP protocol uses Universal Resource Identifiers (URIs) to identify

HTTP resources. HTTP is described in RFC 2016.

- (ii) **CoAP:** Constrained Application Protocol (CoAP) is an application layer protocol for machine-to-machine (M2M) applications, meant for constrained environments with constrained devices and constrained networks. Like HTTP, CoAP is a web transfer protocol and uses a request – response model, however it runs on top of UDP instead of TCP. CoAP uses a client-server architecture where clients communicate with servers using connectionless datagrams. CoAP is designed to easily interface with HTTP. Like HTTP, CoAP supports methods such as GET, PUT, POST, and DELETE. CoAP draft specifications are available on IETF Constrained environments (CoRE) Working Group website.
- (iii) **WebSocket :** WebSocket protocol allows full-duplex communication over a single socket connection for sending messages between client and server. WebSocket is based on TCP and allows streams of messages to be sent back and forth between the client and server while keeping the TCP connection open. The client can be a browser, a mobile application or an IoT device. WebSocket is described in RFC 6455.
- (iv) **MQTT:** Message Queue Telemetry Transport (MQTT) is a light-weight messaging protocol based on the publish – subscribe model. MQTT uses a client – server architecture where the client (such as an IoT device) connects to the server (also called MQTT Broker) and publishes messages to topics on the server. The broker forwards the messages to the clients subscribed to topics. MQTT is well suited for constrained environments where the devices have limited processing and memory resources and the network bandwidth is low. MQTT specifications are available on IBM developer Works.
- (v) **XMPP:** Extensible Messaging and Presence Protocol (XMPP) is a protocol for real – time communication and streaming XML data between network entities. XMPP powers wide range of applications including messaging, presence, data syndication, gaming, multi-party chat and voice/video calls. XMPP allows sending small chunks of XML data from one network entity to another in near real-time. XMPP is a decentralized protocol and uses a client-server architecture. XMPP supports both client-to-server and server-to-server communication paths. In the context of IoT, XMPP

allows real-time communication between IoT devices. XMPP is described in RFC 6120.

(vi) **DDS:** Data Distribution Service (DDS) is a data-centric middleware standard for device-to-device or machine-to-machine communication. DDS uses a publish – subscribe model where publishers (e.g. devices that generate data) create topics to which subscribers (e.g., devices that want to consume data) can subscribe. Publisher is an object responsible for data distribution and the subscriber is responsible for receiving published data. DDS provides quality-of-service (QoS) control and configurable reliability, DDS is described in Object Management Group (OMG) DDS specification.

(vii) **AMQP:** Advanced Message Queuing Protocol (AMQP) is an open application layer protocol for business messaging. AMQP supports both point-to-point and publisher/subscriber models, routing and queuing. AMQP brokers receive messages from publishers (e.g., devices or applications that generate data) and route them over connections to consumers (applications that process data). Publishers publish the messages to exchanges which then distribute message copies to queues. Messages are either delivered by the broker to the consumers which have subscribed to the queues or the consumers can pull the messages from the queues. AMQP specification is available on the AMQP working group website.

Q.26 Write detailed note on IoT standards.

Ans. IoT Standards: IoT is a distributed multi-technology system relying on communications, network techniques, sensor technologies and data processing techniques. Such systems are guided by a range of industrial standards and which are being developed by different regulatory organizations and industrial bodies such as the IEEE, ETSI, IETF, OneM2M, NIST (National Institute of Standards and Technology), ISO (International Organization of Standardization), etc. In this section we focus on some of the standards to introduce international R&D and standardization activities.

One of the major international initiatives to develop IoT standards is known as the oneM2M which was launched in 2012 initially by fourteen partners. Their current membership includes manufacturers, network operators, service and content providers, universities and R&D

organizations, user organizations, consulting and partnership companies. Currently the membership of oneM2M has grown to 200 members. The group has published a range of technical specifications to govern the development of IoT technologies. The oneM2M layered model and the functional architecture is shown in Fig. 2 The layered model includes different functionalities required to provide services using the Cloud IoT architecture as shown in Fig. 1. The layered model distributes all functions among these three layers. The lowest layer supports all communication and networking tasks. The common service layer's main functions include transaction management, service charging and accounting, security, registration, network service related functions as well as device, data communication and application layer management functionalities. This layer is also responsible for intra M2M service provider communications. Details on the standard can be found in the standards document.

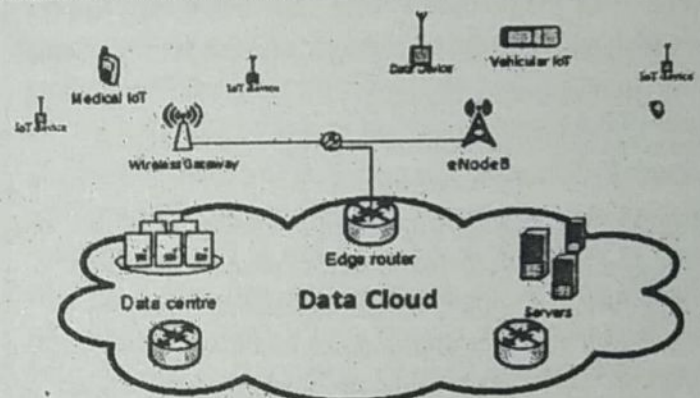


Fig. 1 : A typical cloud IoT operational system-

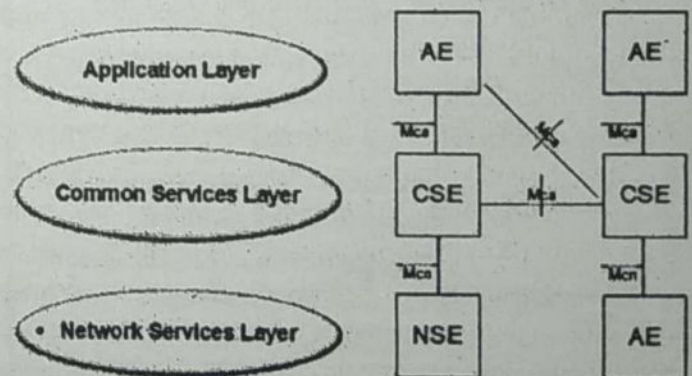


Fig. 2 : OneM2M layered model and the functional architecture

The AE (Application Entity) implements an M2M application service logic. Examples of such application entities are smart metering monitoring in a smart grid or a patient monitoring system using an IoT infrastructure. The NSE provides services from the underlying network to the CSE entity. Examples of NSE services include device management, location services and device triggering.

Many international organizations such as the ETSI, ARIB (Association of Radio Industries and Business), TTA (Telecommunications Industry Association), etc. and major equipment vendors such as IBM, CISCO, Motorola, LG, Nokia, Samsung as well as many Telco's are members of the oneM2M. Such a multinational body which aims at developing a standard is likely to be an international one which will be followed by the IoT sectors. In summary the key technical specifications and reports generated by oneM2M are listed below:

- Use cases and requirements for common services
- High level detailed service architecture
- Open interfaces and protocols
- Interoperability and test conformation
- Information models and data management functions

Besides the oneM2M general system framework, there are many other organizations developing different component systems/protocols. Some of the common protocols which are in use in current IoT systems are shown in Fig. 3. As the IoT system is a multi-technology platform, it is necessary for multiple protocols to work in a cooperative manner as shown in Fig. 3. Various IoT protocols are grouped according to their functionalities and compared with the OSI (Open System Interconnection) model. The figure 3 shows that the lower protocols are related to the communication networks which can be seen as the network services layer of the oneM2M model. References can be found to discover detailed information about different standards. The network and transport layer protocols such as IPv6 (Internet Protocol version 6), TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) are the standard ones and have been used by many networking applications for some time. The RPL (Routing Protocol for Low Power and Lossy Networks) has been developed by the IETF to work with standards such as IEEE802.15.4 where transmission channels could be unreliable and packet losses can occur. So, this protocol is attempting to solve the routing issues in battery powered networks where wireless channels are not very reliable.

The Zigbee protocol by the Zigbee alliance has developed their routing, security and packet transport mechanisms on top of the IEEE802.15.4 radio which implements the Physical and MAC layers. Zigbee technology is quite mature and is now extensively used in IoT systems. The Zigbee alliance is an industry standard which provides

certifications for the range of IoT products. The alliance promotes green networking through the use of low power and efficient networking techniques. Detailed information for system developers is available from their websites.

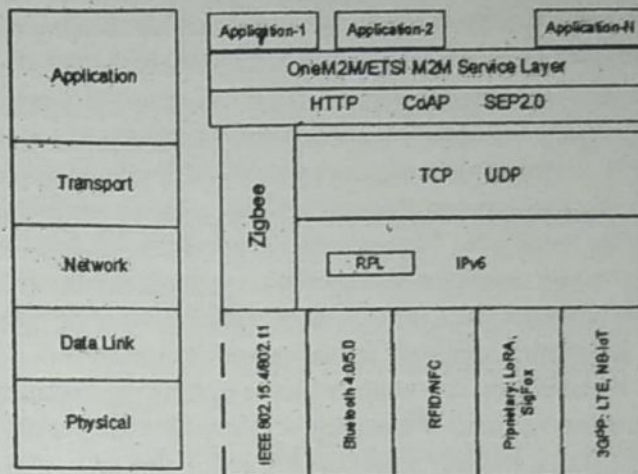


Fig. 3 : Commonly used protocols used in IoT system designs

Figure 3 shows some application layer protocols that can be used for IoT systems. HTTP is a very well defined matured protocol used for internet applications. CoAP (Constrained Application Protocol) is a specialized web transfer protocol developed for the IoT systems that allows nodes and networks to operate in the computing and energy constrained mode. The CoAP protocol was proposed and developed by the IETF. The protocol provides a request/response interaction between end points and supports built in discovery of services and resources. It is a UDP based protocol which supports asynchronous message exchanges with low overhead and low parsing complexities. CoAP protocol also supports simple proxy and caching capabilities.

The IEEE is another major organization which develops a range of computing and communications standards in the IoT area. The IEEE standards association maintains many standards that are applicable to the IoT systems design. List includes standards on WPAN (Wireless Personal Area Network), the IEEE802.15.4.x family, WLAN (Wireless Local Area Networks); the IEEE802.11x family and the WWAN (Wireless Wide Area Networks); the IEEE802.16x family. In addition to the above standard, the IEEE802.22 standards define the cognitive network architecture which can operate in a shared radio spectrum where wireless networks can share transmission channels with television stations. IoT specific other standards include IEEE2030, defining the smart grid architecture and IEEE1609 which supports vehicular

network specifications which can be used to form vehicular IoT networks. The IEEE is also involved with other organizations to develop standards in the areas of IoT systems.

Most of the above standards are open standards that provide reference architecture and guidelines to develop protocols and systems for IoT applications. Among the discussed standards most of them are open standards, except the Zigbee one. There are other commercial groups developing IoT standards which are proprietary in nature where licenses may be required or patented devices to design and develop systems. Examples of such systems are LoRa, SigFox and Ingenu. These systems support communications tasks in unlicensed wireless transmission bands sharing with many other systems. Currently many organizations and groups are developing standards related to various components of IoT technologies. It is expected some of them could merge with others and some of them may or may not be commercially successful. It will take some time before IoT standards attain maturity level, such as we have seen in the development of wireless communication standards.

Q.27 Explain logical design of IoT in detail.

Ans. The logical design illustrates the abstract representation of processes and entities without going into the details of lower-level implementation. The following terms are required to be understood for a complete understanding of logical design:

1. Functional blocks
2. Communication models
3. Communication APIs
4. Functional models

1. IoT Functional Blocks: An IoT system comprises of a number of functional blocks that provide the system the capabilities for identification, sensing, actuation, communication and management as shown in Figure 1. These functional blocks are described as follows:

- (i) **Device:** An IoT system comprises of devices that provide sensing, actuation, monitoring and control functions.
- (ii) **Communication:** The communication block handles the communication for the IoT system.
- (iii) **Services:** An IoT system uses various types of IoT services such as services for device monitoring, device control services, data publishing services and services for device discovery.

- (iv) **Management:** Management functional block provides various functions to govern the IoT system.
- (v) **Security:** Security functional block secures the IoT system and by providing functions such as authentication, authorization, message and content integrity and data security.
- (vi) **Application:** IoT applications provide an interface that the users can use to control and monitor various aspects of the IoT system. Applications also allow users to view the system status and view or analyze the processed data.

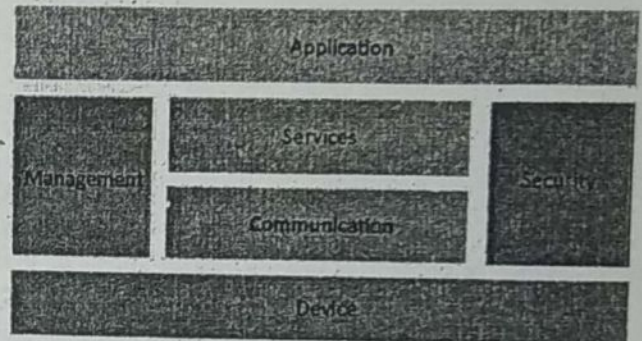


Fig. 1 : IoT functional block

2. Communication models: For operational perception, it is important and useful to understand how various IoT devices communicate with each other. In terms of technical communication models, some major IoT communication models are as described in the following sections :

(i) Request and Response Model: This model follows client-server architecture. The client, when required, requests for the information from the server. This request is usually in encoded format. The server categorizes the request, and fetches the data from the database and its resource representation. This data is converted to response and is transferred in an encoded format to the client. The client, in turn, receives the response. The request-response is a stateless model since the data between the requests is not retained. The working process of request response model is shown in figure 2.

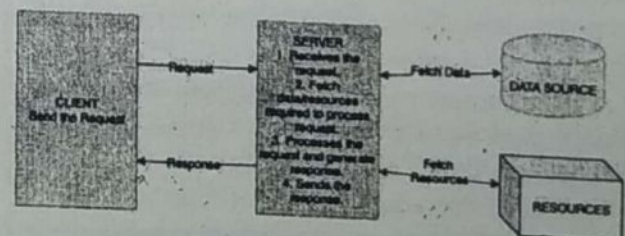


Fig. 2 : Request response model

(ii) **Publisher - Subscriber Model:** The model comprises publishers, brokers, and consumers.

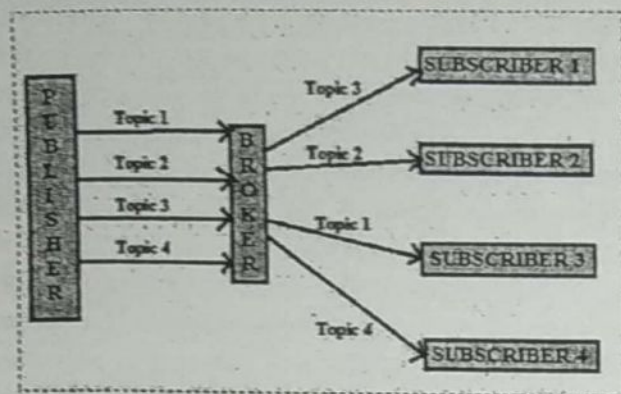


Fig. 3 : Publisher-Subscriber model

Publishers are the source of data, which is sent in the form of topics. The topics are sent to the intermediate broker, which in turn transfers them to subscriber/consumer who has subscribed for the particular topic. The broker only has the information regarding the consumer to which a particular topic belongs to which the publisher is unaware of. The working process of Publisher-Subscriber model is shown in figure 3.

(iii) **Push-Pull Model:** The push-pull model constitutes data publishers, data consumers and data queues. The working process of Push-Pull model is shown in figure 4, whereas publishers publish the messages/data and push it into the queue. The consumer, present on the other side, pulls the data out of the queue. Thus the queue acts as the buffer for the messages when the difference occurs in the rate of push or pull of data on the side of a publisher and consumer.

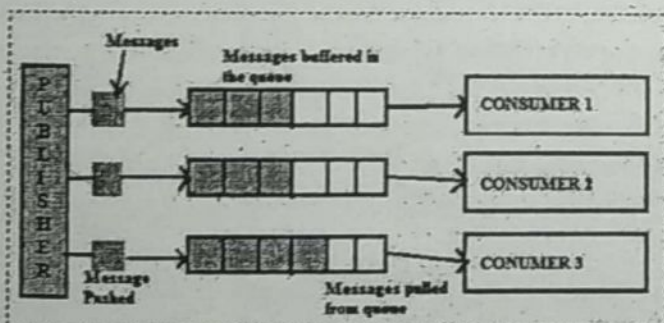


Fig. 4 : Push-Pull model

(iv) **Exclusive Pair Model:** Exclusive pair is a bi-directional model, including full duplex communication among client and server. After the connection is set up between client and server, both can share messages with each other. The once opened connection, will not be closed until the client requests to close the connection. The server has the record of all the connections which has been opened. The working process of exclusive pair model is shown in figure 5.

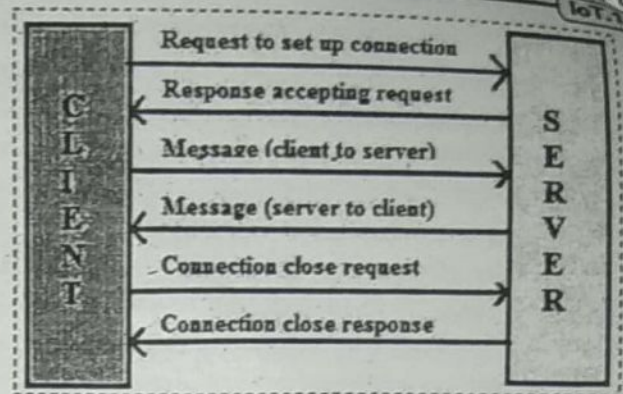


Fig. 5 : Exclusive pair model

3. **IoT communication APIs:** The IoT communication generally uses two APIs. These are:

- (i) REST-based communication APIs
- (ii) Web socket-based communication APIs

(i) **Representational State Transfer (REST) Based Communication APIs :** In case of Representational State Transfer (REST), every time the client wants to retrieve data from a server a connection needs to be established. A client sends a request, which is received by server. The server then processes the request and sends the response/data back to the client. In this way, REST is unidirectional, therefore there is more overhead in this case. The applications demanding real time responses, or those requiring to display the streams of data could not be made to work in this case. An example of REST based communication is shown in figure 6, whereas client send the request to the server using HTTP or HTTPs Protocol.

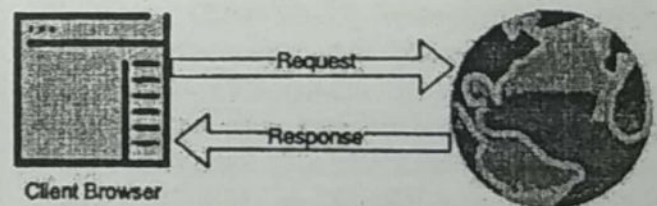


Fig. 6 : REST based communication APIs

(ii) **Web Socket Based Communication APIs:** In web socket-based communication API, over TCP connection: the connection with the server is established only once which is done by initial handshake with the server. The server can send data at any time and the client can handle receiving that data, the client can also send other requests to the server. Such kind of API is really useful when you need to have low latency with data interactions on your web application. An example of Web-Sockets based communication is shown in

figure 7, whereas client and server are communicating through WS or WSS.

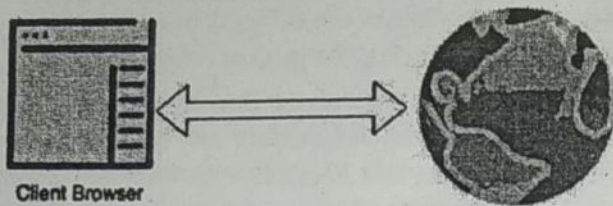


Fig. 7 : Web-Socket based communication APIs

Using web-socket based API; messages can be transferred in both the direction client to server and server to client. Figure 8 shows that the connection with the server is established only once which is done by initial handshake with the server. After successful establishment of the connection, server can send data any time to the client; client can also send a new service request to the server. The established connection can be closed by any device either client or server at any time of instance.

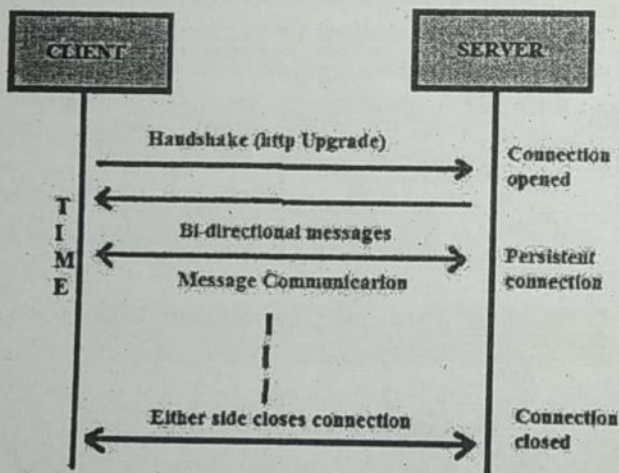


Fig. 8 : Bi-directional message transfer Web-Socket based APIs

Q.28 Write detailed note on IoT enabling technologies.

Ans. 101 Enabling Technologies: IoT is enabled by several technologies including wireless sensor networks, cloud computing, big data analytics, embedded systems, security protocols and architectures, communication protocols, web services, mobile internet and semantic search engines. This section provides an overview of some of these technologies which play a key-role in IoT.

1. Wireless Sensor Networks: A Wireless Sensor Network (WSN) comprises of distributed devices with sensors which

are used to monitor the environmental and physical conditions. A WSN consist of a number of end-nodes and routers and a coordinator. End nodes have several sensors attached to them. End nodes can also act as routers. Routers are responsible for routing the data packets from end-nodes to the coordinator. The coordinator collects the data from all the nodes. Coordinator also acts as a gateway that connects the WSN to the internet. Some examples of WSNs used in IoT systems are described as follows:

- (i) Weather monitoring systems use WSNs in which the nodes collect temperature, humidity and other data, which is aggregated and analyzed.
- (ii) Indoor air quality monitoring systems use WSNs to collect data on the indoor air quality and concentration of various gases.
- (iii) Soil moisture monitoring systems use WSNs to monitor soil moisture at various locations.
- (iv) Surveillance systems use WSNs for collecting surveillance data (such as motion detection data).
- (v) Smart grids use WSNs for monitoring the grid at various points.
- (vi) Structural health monitoring systems use WSNs to monitor the health of structures (buildings, bridges) by collecting vibration data from sensor nodes deployed at various points in the structure.

WSNs are enabled by wireless communication protocols such as IEEE 802.15.4. ZigBee is one of the most popular wireless technologies used by WSNs. ZigBee specifications are based on IEEE 802.15.4. ZigBee operates at 2.4 GHz frequency and offers data rates upto 250 KB/s and range from 10 to 100 meters depending on the power output and environmental conditions. The power of WSNs lies in their ability to deploy large number of low-cost and low-power sensing nodes for continuous monitoring of environmental and physical conditions. WSNs are self-organizing networks. Since WSNs have large number of nodes, manual configuration for each node is not possible. The self-organizing capability of WSN makes the network robust. In the event of failure of some nodes or addition of new nodes to the network, the network can reconfigure itself.

2. Cloud Computing: Cloud computing is a transformative computing paradigm that involves delivering applications and services over the internet. Cloud computing involves provisioning of computing, networking and storage resources on demand and providing these resources as metered services

to the users, in a "pay as you go" model. Cloud computing resources can be provisioned on-demand by the users, without requiring interactions with the cloud service provider. The process of provisioning resources is automated. Cloud computing resources can be accessed over the network using standard access mechanisms that provide platform-independent access through the use of heterogeneous client platforms such as workstations, laptops, tablets and smart-phones. The computing and storage resources provided by cloud service providers are pooled to serve multiple users using multi-tenancy. Multi-tenant aspects of the cloud allow multiple users to be served by the same physical hardware. Users are assigned virtual resources that run on top of the physical resources.

Cloud computing services are offered to users in different forms :

- (i) **Infrastructure-as-a-Service (IaaS):** IaaS provides the users the ability to provision computing and storage resources. These resources are provided to the users as virtual machine instances and virtual storage. Users can start, stop, configure and manage the virtual machine instances and virtual storage. Users can deploy operating systems and applications of their choice on the virtual resources provisioned in the cloud. The cloud service provider manages the underlying infrastructure. Virtual resources provisioned by the users are billed based on a pay-per-use paradigm.
- (ii) **Platform-as-a-Service (PaaS):** PaaS provides the users the ability to develop and deploy application in the cloud using the development tools, application programming interfaces (APIs), software libraries and services provided by the cloud service provider. The cloud service provider manages the underlying cloud infrastructure including servers, network, operating systems and storage. The users, themselves, are responsible for developing, deploying, configuring and managing applications on the cloud infrastructure.
- (iii) **Software-as-a-Service (SaaS):** SaaS provides the users a complete software application or the user interface to the application itself. The cloud service provider manages the underlying cloud infrastructure including servers, network, operating systems, storage and application software, and the user is unaware of the underlying architecture of the cloud. Applications are provided to the user through a thin client interface (e.g., a browser). SaaS applications are platform

independent and can be accessed from various client devices such as workstations, laptops, tablets and smart-phones, running different operating systems. Since the cloud service provider manages both the application and data, the users are able to access the applications from anywhere.

3. Big Data Analytics: Big data is defined as collections of data sets whose volume, velocity (in terms of its temporal variation), or variety, is so large that it is difficult to store, manage, process and analyze the data using traditional databases and data processing tools. Big data analytics involves several steps starting from data cleansing, data munging (or wrangling), data processing and visualization. Some examples of big data generated by IoT systems are described as follows:

- (i) Sensor data generated by IoT systems such as weather monitoring stations.
- (ii) Machine sensor data collected from sensors embedded in industrial and energy systems for monitoring their health and detecting failures.
- (iii) Health and fitness data generated by IoT devices such as wearable fitness bands.
- (iv) Data generated by IoT systems for location and tracking of vehicles.
- (v) Data generated by retail inventory monitoring systems.

The underlying characteristics of big data include:

- (i) **Volume:** Though there is no fixed threshold for the volume of data to be considered as big data, however, typically, the term big data is used for massive scale data that is difficult to store, manage and process using traditional databases and data processing architectures. The volumes of data generated by modern IT, industrial, and health-care systems, for example, is growing exponentially driven by the lowering costs of data storage and processing architectures and the need to extract valuable insights from the data to improve business processes, efficiency and service to consumers.
- (ii) **Velocity:** Velocity is another important characteristic of big data and the primary reason for exponential growth of data. Velocity of data refers to how fast the data is generated and how frequently it varies. Modern IT, industrial and other systems are generating data at increasingly higher speeds.

(iii) **Variety:** Variety refers to the forms of the data. Big data comes in different forms such as structured or unstructured data, including text data, image, audio, video and sensor data.

4. Communication Protocols: Communication protocols form the backbone of IoT systems and enable network connectivity and coupling to applications. Communication protocols allow devices to exchange data over the network. Protocols define the data exchange formats, data encoding, addressing schemes for devices and routing of packets from source to destination. Other functions of the protocols include sequence control (that helps in ordering packets determining lost packets), flow control (that helps in controlling the rate at which the sender is sending the data so that the receiver or the network is not overwhelmed) and retransmission of lost packets.

5. Embedded Systems: An embedded system is a computer system that has computer hardware and software embedded to perform specific tasks. In contrast to general purpose computers or personal computers (PCs) which can perform various types of tasks, embedded systems are designed to perform a specific set of tasks. Key components of an embedded system include, microprocessor or microcontroller, memory (RAM, ROM cache), networking units (Ethernet, WiFi adapters), input/output units (display, keyboard, etc.) and storage (such as flash memory). Some embedded systems have specialized processors such as digital signal processors (DSPs), graphics processors and application specific processors. Embedded systems run embedded operating systems such as real-time operating systems (RTOS). Embedded systems range from low-cost miniaturized devices such as digital watches to devices such as digital cameras, point of sale terminals, vending machines, appliances (such as washing machines), etc.

Q.29 Explain IoT levels and deployment templates.

Ans. IoT Levels and Deployment Templates: In this we define various levels of IoT systems with increasing complexity. An IoT system comprises of the following components:

1. **Device:** An IoT device allows identification, remote sensing, actuating and remote monitoring capabilities.
2. **Resource:** Resources are software components on the IoT device for accessing, processing and storing

sensor information or controlling actuators connected to the device. Resources also include the software components that enable network access for the device.

3. Controller Service: Controller service is a native service that runs on the device and interacts with the web services. Controller service sends data from the device to the web service and receives commands from the application (via web services) for controlling the device.

4. Database: Database can be either local or in the cloud and stores the data generated by the IoT device.

5. Web Service: Web services serve as a link between the IoT device, application, database and analysis components. Web service can be either implemented using HTTP and REST principles (REST service) or using WebSocket protocol (WebSocket service). A comparison of REST and WebSocket is provided below:

- (i) **Stateless/Stateful:** REST services are stateless in nature. Each request contains all the information needed to process it. Requests are independent of each other. WebSocket on the other hand is stateful in nature where the server maintains the state and is aware of all the open connections.
- (ii) **Uni-directional/Bi-directional:** REST services operate over HTTP and are uni-directional. Request is always sent by a client and the server responds to the requests. On the other hand, WebSocket is a bi-directional protocol and allows both client and server to send messages to each other.
- (iii) **Request-Response/Full Duplex:** REST services follow a request-response communication model where the client sends requests and the server responds to the requests. WebSocket on the other hand allow full-duplex communication between the client and server, i.e., both client and server can send messages to each other independently.
- (iv) **TCP Connections:** For REST services, each HTTP request involves setting up a new TCP connection. WebSocket on the other hand involves a single TCP connection over which the client and server communicate in a full-duplex mode.

(v) **Header Overhead:** REST services operate over HTTP, and each request is independent of others. Thus each request carries HTTP headers which is an overhead. Due the overhead of HTTP headers, REST is not suitable for real-time applications. WebSocket on the other hand does not involve overhead of headers. After the initial handshake (that happens over HTTP), the client and server exchange messages with minimal frame information. Thus WebSocket is suitable for real-time applications.

(vi) **Scalability:** Scalability is easier in the case of REST services as requests are independent and no state information needs to be maintained by the server. Thus both horizontal (scaling-out) and vertical scaling (scaling-up) solutions are possible for REST services. For WebSockets, horizontal scaling can be cumbersome due to the stateful nature of the communication. Since the server maintains the state of a connection, vertical scaling is easier for WebSockets than horizontal scaling.

Analysis Component: The Analysis Component is

responsible for analyzing the IoT data and generate results in a form which are easy for the user to understand. Analysis of IoT data can be performed either locally or in the cloud. Analyzed results are stored in the local or cloud databases.

Application: IoT applications provide an interface that the users can use to control and monitor various aspects of the IoT system. Applications also allow users to view the system status and view the processed data.

Q.30 Write short notes on the following :

(a) IoT level-5

(b) IoT level-6

Ans. (a) IoT Level-5: A level-5 IoT system has multiple end nodes and one coordinator node as shown in figure. The end nodes that perform sensing and/or actuation. Coordinator node collects data from the end nodes and sends to the cloud. Data is stored and analyzed in the cloud and application is cloud-based. Level-5 IoT systems are suitable for solutions based on wireless sensor networks, in which the data involved is big and the analysis requirements are computationally intensive.

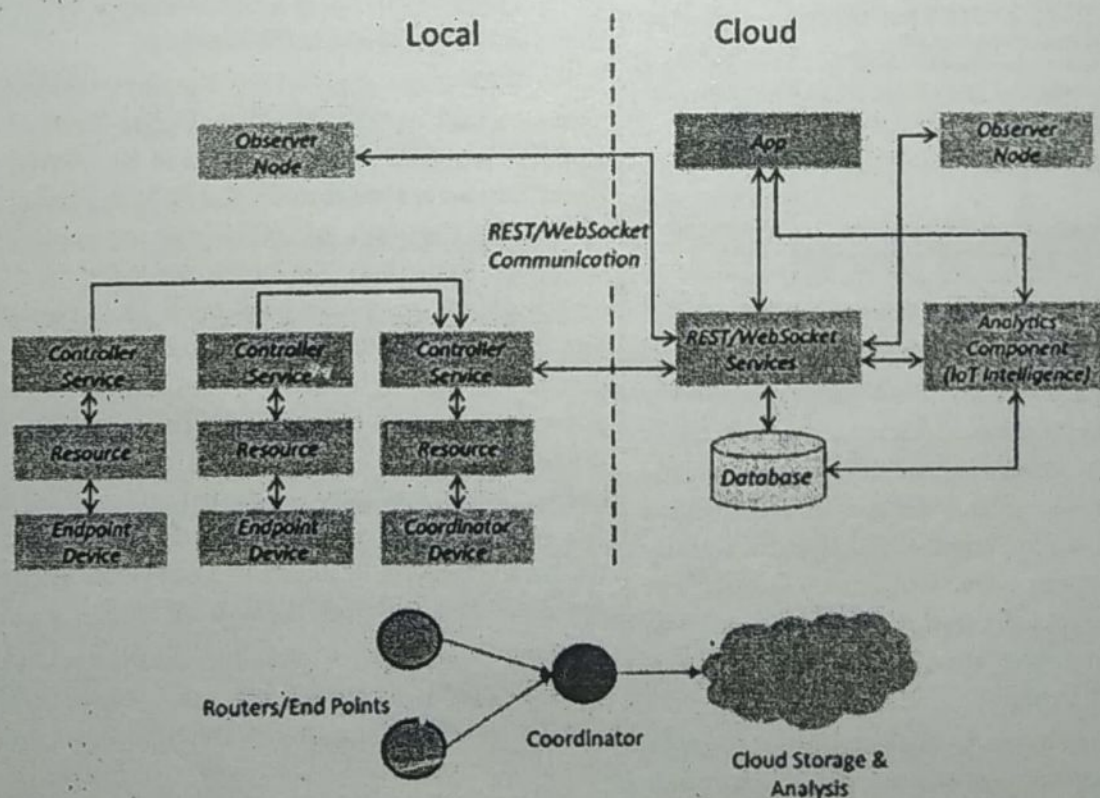


Fig. : IoT Level-5

Let us consider an example of a level-5 IoT system for forest fire detection. The system consists of multiple nodes placed in different locations for monitoring temperature, humidity and carbon dioxide (CO₂) levels in a forest. The end nodes in this example are equipped with various sensors (such as temperature, humidity and CO₂). The coordinator node collects the data from the end nodes and acts as a gateway that provides Internet connectivity to the IoT system. The controller service on the coordinator device sends the collected data to the cloud. The data is stored in a cloud database. The analysis of data is done in the computing cloud

to aggregate the data and make predictions. A cloud-based application is used for visualizing the data.

Ans(b) IoT Level-6: A level-6 IoT system has multiple independent end nodes that perform sensing and/or actuation and send data to the cloud. Data is stored in the cloud and application is cloud-based as shown in figure. The analytics component analyzes the data and stores the results in the cloud database. The results are visualized with the cloud-based application. The centralized controller is aware of the status of all the end nodes and sends control commands to the nodes.

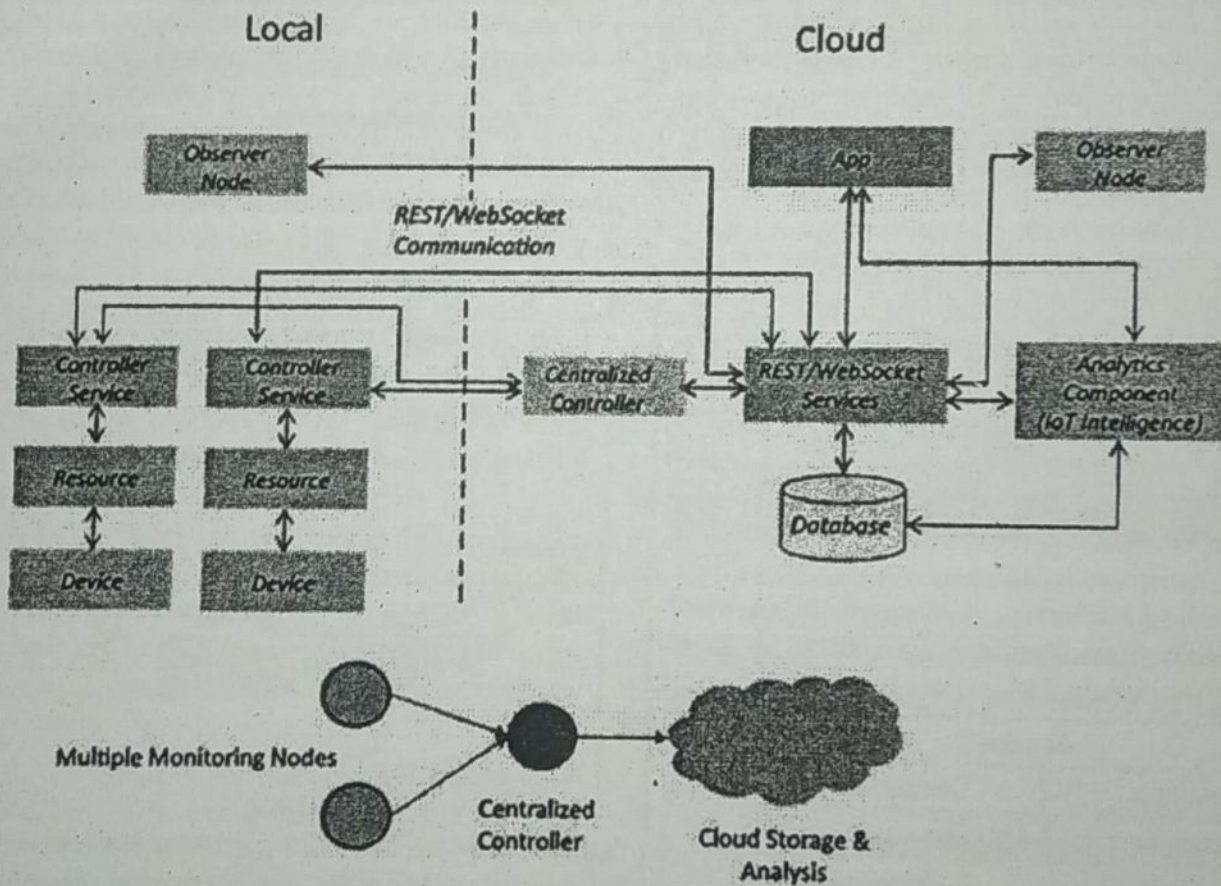


Fig. : IoT Level-6

Let us consider an example of a level-6 IoT system for weather monitoring. The system consists of multiple nodes placed in different locations for monitoring temperature, humidity and pressure in an area. The end nodes are equipped with various sensors (such as temperature, pressure and

humidity). The end nodes send the data to the cloud in real-time using a WebSocket service. The data is stored in a cloud database. The analysis of data is done in the cloud to aggregate the data and make predictions. A cloud-based application is used for visualizing the data.

□□□

IoT HARDWARE AND SOFTWARE

2

IMPORTANT QUESTIONS

PART-A

Q.1 *What are the most used sensors types in IoT?*

Ans. Most used Sensors Types in IoT :

1. Temperature sensors
2. Proximity sensor
3. Pressure sensor
4. Gas sensor
5. Smoke sensor
6. IR sensors
7. Motion detection sensors

Q.2 *What sensor and actuator are used to control any home appliances from any IoT devices in wired mode?*

Ans. A relay is used to control any home appliances from any IoT or embedded devices. A relay is nothing but an electrically operated switch.

Q.3 *What is the syntax to read analog and digital data from a sensor in Arduino?*

Ans. `digitalRead()` and `digitalWrite()` are respectively used to read and write digital data to the sensors. `analogRead()` and `analogWrite()` are respectively used to read and write analog data to the sensors.

Q.4 *What are interrupts in Arduino?*

Ans. Interrupts allow certain important tasks to happen in the background and are enabled by default. An interrupt's job is to make sure that the processor responds quickly to important events. When a certain signal is detected, an Interrupt interrupts whatever the processor is doing, and executes some code designed to react.

Q.5 *What is an Arduino?*

Ans. Arduino : Arduino is an open-source electronics platform which has easy to use both hardware and software. Arduino boards are a microcontroller which is capable of reading input from sensors to controlling motors and etc. programmatically.

Q.6 *How to write instructions or programs for Arduino boards?*

Ans. The Arduino Software (IDE) allows you to write programs and upload them to your board. A bootloader is needed to upload or flash the code to the board.

Q.7 *What are the hardware communication interfaces present in the Arduino board?*

Ans. It has several communication protocols like I2C, SPI, Serial, PWM and etc.

Q.8 *What programming language is used to code Arduino?*

Ans. Basically C programming language is used to code Arduino boards.

Q.9 What is a Raspberry Pi?

Ans. Raspberry Pi : Raspberry Pi is a credit card sized computer which is capable of doing all operations like a conventional computer. But it also has other built-in features like onboard Wi-Fi, Bluetooth and GPIO pins to communicate with other external things.

Q.10 Difference between Arduino and Raspberry Pi?

Ans. Difference between Arduino and Raspberry Pi : Basically, Arduino is a micro-controller and Raspberry Pi is a microprocessor. Raspberry Pi is slightly superior to Arduino boards like it has better CPU and GPU processing along with onboard Bluetooth, Wi-Fi, Ethernet etc.

Q.11 What is the operating voltage for both Arduino and Raspberry Pi?

Ans. Raspberry Pi works in 5V input voltage and for Arduino, its operating voltage is between 5 – 12V. Arduino boards have a regulator, which help is work on a different input voltage.

Q.12 What are the hardware communication interfaces present in the Raspberry Pi?

Ans. Similar to Arduino boards Raspberry Pi also has several communication protocols like I2C, SPI, Serial, PWM etc.

Q.13 List a few operating systems that Raspberry Pi supports?

Ans. The official operating system for Raspberry Pi is Raspbian. Although it supports other operating systems like Kali Linux, OSMC, Windows 10 IOT Core, Android Things, RetroPie etc.

Q.14 How do you run Raspberry pi in headless mode?

Ans. You can use SSH into Raspberry Pi and run in headless mode. Latest Raspbian OS has inbuilt VNC server installed with that you can take remote desktop on Raspberry Pi.

Q.15 What are the available wireless communications boards present in Raspberry Pi?

Ans. Wi-Fi and Bluetooth/BLE are the wireless communications present in Raspberry Pi.

Q.16 What are the sensors can be used in agriculture?

Ans. Sensors used in Agriculture :

1. Soil moisture sensor
2. Airflow sensors
3. Electro chemical sensors

Q.17 What is purpose of airflow sensors?

Ans. Purpose of Airflow Sensors : It used to measure the air level in soil, we can measure it from the one location or dynamically get from multiple places from the garden.

Q.18 What are the available models in Raspberry Pi?

Ans. Available Models in Raspberry Pi :

- Raspberry Pi 1 model B
- Raspberry Pi 1 model A
- Raspberry Pi 1 model B+
- Raspberry Pi 1 model A+
- Raspberry Pi Zero
- Raspberry Pi 2
- Raspberry Pi 3 model B
- Raspberry Pi Zero W

Q.19 Write down the names of real time usage of Raspberry Pi.

Ans. Names of Real Time Usage of Raspberry Pi :

1. Home automation
2. Internet radio
3. Portable webserver
4. Manipulating the robots

PART-B

Q.20 What do you understand by IoT hardware?

Ans. IoT Hardware : The hardware utilized in IoT systems includes devices for a remote dashboard, devices for control, servers, a routing or bridge device and sensors. These devices manage key tasks and functions such as system activation, action specifications, security, communication, and detection to support specific goals and actions.

IoT Sensors : The most important hardware in IoT might be its sensors. These devices consist of energy modules, power management modules, RF modules and sensing modules. RF modules manage communications through their signal processing, Wi-Fi, ZigBee, Bluetooth, radio transceiver, duplexer and BAW.

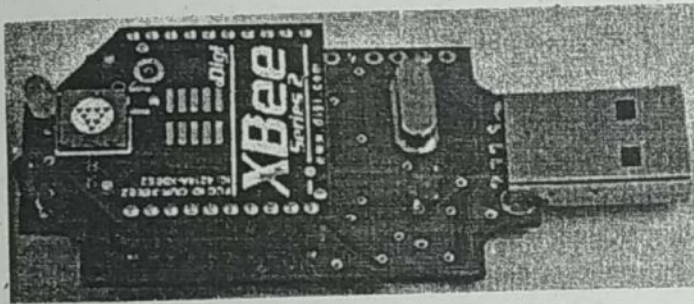


Fig. 1

The sensing module manages sensing through assorted active and passive measurement devices. Here is a list of some of the measurement devices used in IoT:

| Devices | |
|------------------|---------------------|
| Accelerometers | Temperature sensors |
| Magnetometers | Proximity sensors |
| Gyroscopes | Image sensors |
| Acoustic sensors | Light sensors |
| Pressure sensors | Gas RFID sensors |
| Humidity sensors | Micro flow sensors |

Wearable Electronics: Wearable electronic devices are small devices worn on the head, neck, arms, torso and feet. Smart watches not only help us stay connected, but as a part of an IoT system, they allow access needed for improved productivity.

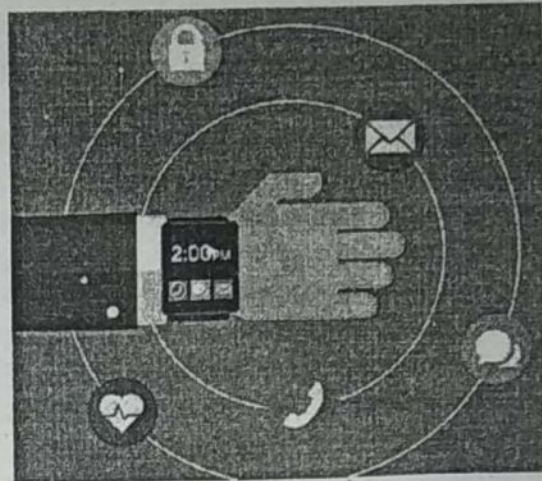


Fig. 2

Current smart wearable devices include:

- (i) **Head** – Helmets, glasses
- (ii) **Neck** – Jewelry, collars
- (iii) **Arm** – Watches, wristbands, rings
- (iv) **Torso** – Clothing, backpacks
- (v) **Feet** – Socks, shoes

Smart glasses help us enjoy more of the media and services we value, and when part of an IoT system, they allow a new approach to productivity.



Fig. 3

Standard Devices: The desktop, tablet and cellphone remain integral parts of IoT as the command center and remotes.

- (i) The desktop provides the user with the highest level of control over the system and its settings.
- (ii) The tablet provides access to the key features of the system in a way resembling the desktop and also acts as a remote.
- (iii) The cellphone allows some essential settings modification and also provides remote functionality.

Other key connected devices include standard network devices like routers and switches.

Q.21 Write short note on working of actuator.

Ans. Working of Actuator : Actuators convert electrical signals (e.g., commands emanating from the control computer) into mechanical motion or other physical variables (e.g., pressure or temperature), and thus actively intervene with the control system and/or set variables. In the field of measurement and control engineering, actuators are the signal-related counterparts to sensors. Types of actuators include hydraulic, pneumatic, electric, mechanical and piezoelectric. They convert signals or setting and regulation specifications of a control into (mostly) mechanical work. A simple example of this is the opening and closing of a valve, for example, in a heating system or in the case of engine controls. The output of optical (via displays) or acoustic signals can also be subsumed under actuators, since they can trigger an effect in the real environment. In robotics, the term effector is often used as an equivalent for actuators. Effectors allow a robot to grasp and manipulate objects, and thus produce an effect. In a computerized world of things, actuators play an increasingly important role in the realization of actions and effects as a counterpart to the (previously) sensory-detected corresponding contexts. Actuators are a key building block in more recent perceptions of the "Fourth Industrial Revolution" in manufacturing as an Industry 4.0 conceptualization postulate.

As an example, we can consider a smart home system, which consists of many sensors and actuators. The actuators are used to lock/unlock the doors, switch on/off the lights or other electrical appliances, alert users of any threats through alarms or notifications, and control the temperature of a home (via a thermostat).

A sophisticated example of an actuator used in IoT is a digital finger, which is used to turn on/off the switches (or anything which requires small motion) and is controlled wirelessly.

Q.22 Write short note on IoT software.

Ans. IoT Software : IoT software addresses its key areas of networking and action through platforms, embedded systems partner systems, and middleware. These individual and master applications are responsible for data collection, device integration, real-time analytics and application and process extension within the IoT network. They exploit integration with critical business systems (e.g., ordering

systems, robotics, scheduling, and more) in the execution of related tasks.

Data Collection : This software manages sensing, measurements, light data filtering, light data security and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine-to-machine networks. Then it collects data from multiple devices and distributes it in accordance with settings. It also works in reverse by distributing data over devices. The system eventually transmits all collected data to a central server.

Device Integration : Software supporting integration binds (dependent relationships) all system devices to create the body of the IoT system. It ensures the necessary cooperation and stable networking between devices. These applications are the defining software technology of the IoT network because without them, it is not an IoT system. They manage the various applications, protocols and limitations of each device to allow communication.

Real-Time Analytics : These applications take data or input from various devices and convert it into viable actions or clear patterns for human analysis. They analyze information based on various settings and designs in order to perform automation-related tasks or provide the data required by industry.

Application and Process Extension : These applications extend the reach of existing systems and software to allow a wider, more effective system. They integrate predefined devices for specific purposes such as allowing certain mobile devices or engineering instruments access. It supports improved productivity and more accurate data collection.

Q.23 Write short note on followings :

(a) Neural Sensors

(b) Environmental and Chemical Sensors

Ans.(a) Neural Sensors : Today, it is possible to understand neural signals in the brain, infer the state of the brain and train it for better attention and focus. This is known as neurofeedback. The technology used for reading brain signals is called EEG (Electroencephalography) or a brain computer interface. The neurons inside the brain communicate electronically and create an electric field, which can be measured from outside in terms of frequencies. Brain waves can be categorized into alpha, beta, gamma, theta and delta waves depending upon the frequency.

Based on the type of wave, it can be inferred whether the brain is calm or wandering in thoughts. This type of neurofeedback can be obtained in real time and can be used to train the brain to focus, pay better attention towards things, manage stress, and have better mental well-being.

Ans.(b) Environmental and Chemical Sensors : Environmental sensors are used to sense parameters in the physical environment such as temperature, humidity, pressure, water pollution and air pollution. Parameters such as the temperature and pressure can be measured with a thermometer and barometer. Air quality can be measured with sensors, which sense the presence of gases and other particulate matter in the air.

Chemical sensors are used to detect chemical and biochemical substances. These sensors consist of a recognition element and a transducer. The electronic nose (e-nose) and electronic tongue (e-tongue) are technologies that can be used to sense chemicals on the basis of odor and taste, respectively. The e-nose and e-tongue consist of an array of chemical sensors coupled with advance pattern recognition software. The sensors inside the e-nose and e-tongue produce complex data, which is then analyzed through pattern recognition to identify the stimulus.

These sensors can be used in monitoring the pollution level in smart cities, keeping a check on food quality in smart kitchens, testing food and agricultural products in supply chain applications.

Q.24 What do you mean by ultrasonic sensors?

Ans. Ultrasonic Sensors : An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves and converts the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound (i.e. the sound that humans can hear). Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver (which encounters the sound after it has travelled to and from the target).

In order to calculate the distance between the sensor and the object, the sensor measures the time it takes between the emission of the sound by the transmitter to its contact with the receiver. The formula for this calculation is $D = \frac{1}{2} T \times C$ (where D is the distance, T is the time and C is the speed of sound ~ 343 meters/second). For example, if a scientist set up an ultrasonic sensor aimed at a box and it took 0.025

seconds for the sound to bounce back, the distance between the ultrasonic sensor and the box would be:

$$D = 0.5 \times 0.025 \times 343 = 4.2875 \text{ meters}$$

Ultrasonic sensors are used primarily as proximity sensors. They can be found in automobile self – parking technology and anti-collision safety systems. Ultrasonic sensors are also used in robotic obstacle detection systems, as well as manufacturing technology. In comparison to infrared (IR) sensors in proximity sensing applications, ultrasonic sensors are not as susceptible to interference of smoke, gas and other airborne particles (though the physical components are still affected by variables such as heat).

Ultrasonic sensors are also used as level sensors to detect, monitor and regulate liquid levels in closed containers (such as vats in chemical factories). Most notably, ultrasonic technology has enabled the medical industry to produce images of internal organs, identify tumors and ensure the health of babies in the womb.

Q.25 What is a temperature sensor? What temperature sensors do?

Ans. Temperature Sensor : A temperature sensor is a device, usually an RTD (Resistance Temperature Detector) or a thermocouple, that collects the data about temperature from a particular source and converts the data into understandable form for a device or an observer. Temperature sensors are used in many applications like HV and AC system environmental controls, food processing units, medical devices, chemical handling and automotive under the hood monitoring and controlling systems, etc.

The most common type of temperature sensor is a thermometer, which is used to measure temperature of solids, liquids and gases. It is also a common type of temperature sensor mostly used for non-scientific purposes because it is not so accurate.

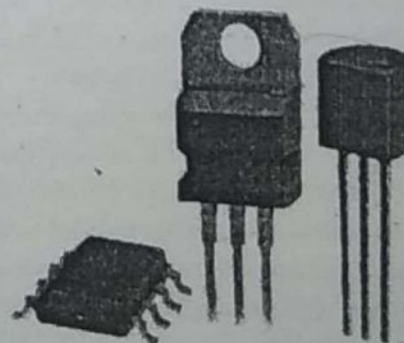


Fig. : Temperature Sensors can use sensors to monitor medical parameters

A temperature sensor is a device that is designed to measure the degree of hotness or coolness in an object. The working of a temperature meter depends upon the voltage across the diode. The temperature change is directly proportional to the diode's resistance. The cooler the temperature, lesser will be the resistance, and vice-versa.

The resistance across the diode is measured and converted into readable units of temperature (Fahrenheit, Celsius, Centigrade, etc.) and displayed in numeric form over readout units. In geotechnical monitoring field, these temperature sensors are used to measure the internal temperature of structures like bridges, dams, buildings, power plants, etc.

Q.26 What is a temperature sensor used for? What are the functions of a temperature sensor?

Ans. Well, there are many types of temperature sensors, but, the most common way to categorise them is based upon the mode of connection which includes, contact and non-contact temperature sensors.

Contact sensors include thermocouples and thermistors because they are in direct contact with the object they are to measure. Whereas, the non-contact temperature sensors measure the thermal radiation released by the heat source. Such temperature meters are often used in hazardous environments like nuclear power plants or thermal power plants.

In geotechnical monitoring, temperature sensors measure the heat of hydration in mass concrete structures. They can also be used to monitor the migration of groundwater or seepage. One of the most common areas where they are used is while curing the concrete because it has to be relatively warm in order to set and cure properly. The seasonal variations cause structural expansion or contraction thereby, changing its overall volume.

Q.27 Explain the working of temperature sensors.

Ans. Working of Temperature Sensors : The basic principle of working of the temperature sensors is the voltage across the diode terminals. If the voltage increases, the temperature also rises, followed by a voltage drop between the transistor terminals of base and emitter in a diode.

Besides this, Encardio-Rite has a vibrating wire temperature sensor that works on the principle of stress change to temperature change.

The vibrating wire temperature meter is designed on the principle that dissimilar metals have a different linear coefficient of expansion with temperature variation.

It primarily consists of a magnetic, high tensile strength stretched wire, the two ends of which are fixed to any dissimilar metal in a manner that any change in temperature directly affects the tension in the wire and, thus, its natural frequency of vibration.

The dissimilar metal, in the case of the Encardio-Rite temperature meter, is aluminium (Aluminum has a larger coefficient of thermal expansion than steel.) As the temperature signal is converted into frequency, the same read-out unit which is used for other vibrating wire sensors can also be used for monitoring temperature also.

The change in temperature is sensed by the specially built Encardio-rite vibrating wire sensor and is converted to an electrical signal which is transmitted as a frequency to the read-out unit. The frequency, which is proportional to the temperature and in turn to the tension ' σ ' in the wire, can be determined as follows:

$$f = 1/2 [\sigma g / \rho] / 2l \text{ Hz}$$

Where:

σ = tension of the wire

g = acceleration due to gravity

ρ = density of the wire

l = length of wire

Q.28 What is arduino? Write advantages of it.

Ans. Arduino : Arduino was invented at the Ivrea Interaction Design Institute. It was designed for fast prototyping, targeting the hobbyist without any programming background. Soon the user-friendly platform attracted an audience covering a wider community and started changing to adapt the latest trends in the market, from an 8-bit board to IoT products, wearable devices, and an embedded environment. Arduino boards are completely open source and can be used for application development with particular requirements. The Arduino software is user-friendly and easy to begin with a flexible environment for advanced users. It can be operated on Mac, Linux, and Window platforms. New things can be learned with Arduino.

Advantages of Arduino:

Cost: Arduino boards are less expensive compared to other microcontroller boards.

Platform: The Arduino Software (IDE) is compatible with most of the operating systems like Macintosh OSX, Windows, and Linux.

User Friendly: The Arduino Software (IDE) is user-friendly, easy to begin, and has flexibility for the skilled programmers.

Open Source: The Arduino is an open-source software that can be programmed with C, C++, or AVR-C languages. So a variety of modules can be designed by users.

Q.29 What do you mean by raspberry Pi? How it is work?

Ans. Raspberry Pi : Raspberry Pi is a microcomputer that we can connect with any keyboard, mouse or a monitor. Raspberry Pi is able to perform all tasks like creating word documents, spreadsheet, browsing, coding, games and much more. It is a tiny Linux computer that is used by many companies for performing all computer-based activities, learning and electronic projects. The Raspberry Pi uses a 32 bit ARM processor and was first developed by Raspberry Pi foundation. The low cost makes it more popular among young people.

Working: The memory card slot is used for inserting an SD card that acts as the hardware of Raspberry Pi. The USB port, HDMI port, and the audio/video port help us connect with a monitor, TV or any other device. This is how Raspberry Pi is capable of working anytime and anywhere. The processor gives the correct speed for running our computer programs all the time.

Q.30 Write short note on TinyOS.

Ans. TinyOS : TinyOS is an open-source non-Linux-based OS designed explicitly for low-end IoT devices, embedded and wireless devices such as sensor node networks, smart buildings, and smart sensory platforms. TinyOS is built based on a set reusable software component. It is written using NesC programming language, which has a similar syntax to C language. Each TinyOS component has a frame and a structure of private variables. These components have three computational abstractions: commands, events, and tasks. Commands are used to call a component to do a specific task. Events are mechanisms for entering component communication, while tasks are used to represent component concurrency.

Architecture and Kernel Models: TinyOS has a monolithic architecture and uses a component-based architecture that

depends on the application requirements. This reduces the size of the code needed to setup hardware. Different components are grouped with the scheduler to run on the mote platform. The mote platform has very insufficient physical resources depending on which components are active. Typical TinyOS motes consist of a 1 microprocessor without interlocked pipeline stages (MIPS) processor and tens of kilobytes of storage. A component is an independent computational element that shows one or more interfaces. Components have three computational abstractions: commands, events, and tasks. Mechanisms for inter-component communication are commands and events, whereas tasks are used to express intra-component concurrency. A command is a request to perform some service while the event signals represent the completion of service. Fig. shows the architecture of TinyOS. 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. Commands and tasks are performed in the context of the frame and operate on its state. Each component declares its commands and events to allow the modularity and easy interaction with other components.

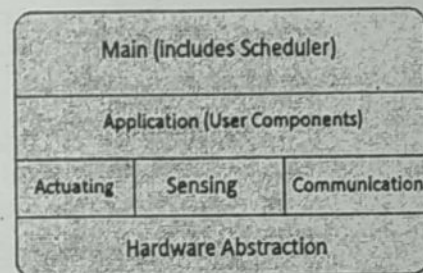


Fig. : Architecture of TinyOS

PART-C

Q.31 Write detailed note on sensors. Explain with neat diagram.

Ans. Sensors : Sensors are technical components for the qualitative or quantitative measurement of certain chemical or physical variables and properties, for example, temperature, light (intensity and color), acceleration, electricity, and so on. The recorded measured values are usually converted into electronic signals. Currently, we are already surrounded by sensors in many places. For example, modern automobiles contain hundreds of sensors, for example, rain sensors for

windshield wiper systems, crash sensors for air bag release systems, and lane and parking-assist sensors. Indeed, modern automobiles, some with far more than 200 sensors and a few dozen microprocessors (Economist, 2009), constitute a good example of this. In fact, the ordinary automobile is increasingly becoming one unified computerized object. In addition, when a sensor is employed together with a processor (controller), a power supply, and a unit for data transmission, this is referred to as a sensor node.

A sensor node's primary function is to collect, preprocess, and transmit sensor data from its environment to other sensor nodes or a base station. Examples of sensor categories include (Baras and Brito, 2017) the following:

- Location: GPS, GLONASS, Galileo
- Biometric: fingerprint, iris, face
- Acoustic: microphone
- Environmental: temperature, humidity, pressure
- Motion: accelerometer, gyroscope

Sensor nodes can form Wireless Sensor Networks (WSN) by means of their transmission unit. For example, these are utilized to (i) detect earthquakes, forest fires, avalanches, as well as terrorist attacks (ii) monitor vehicle traffic, particularly in tunnels (iii) track the movements of wild animals (iv) protect property (v) operate and manage machines and vehicles efficiently (vi) establish security areas (vii) monitor supply chain management (viii) discover chemical, biological, and radiological material. For the operation of sensor networks, special software is required, which ensures a dynamic and robust self-organization of the sensor network that functions in a safe and scalable manner. This is because sensor nodes can fail, change their position, or be only online intermittently. WSN can consist of several hundred or hundreds of thousands of sensor nodes, which are deployed either inside of the phenomenon or very close to it. Sensor nodes are connected to an intermediary network that forward the data that they collect to a computer for analysis.

Sensor nodes are installed in their workspace to function for years, preferably without requiring any maintenance or human intervention. They must therefore have a low energy requirement and have batteries that are functional over several years. The construction of a typical WSN is layered (see Figure) (Hill et al., 2004). Specifically, it begins with sensors on the lower level and continues up to the top-level nodes for data collection, analysis, and storage. Simple and complex data are routed through a network to an automated facility that provides continuous monitoring and control of the dedicated environment. WSNs do not

necessarily operate on all layers with the common TCP/IP stack and may use dedicated lightweight protocols instead.

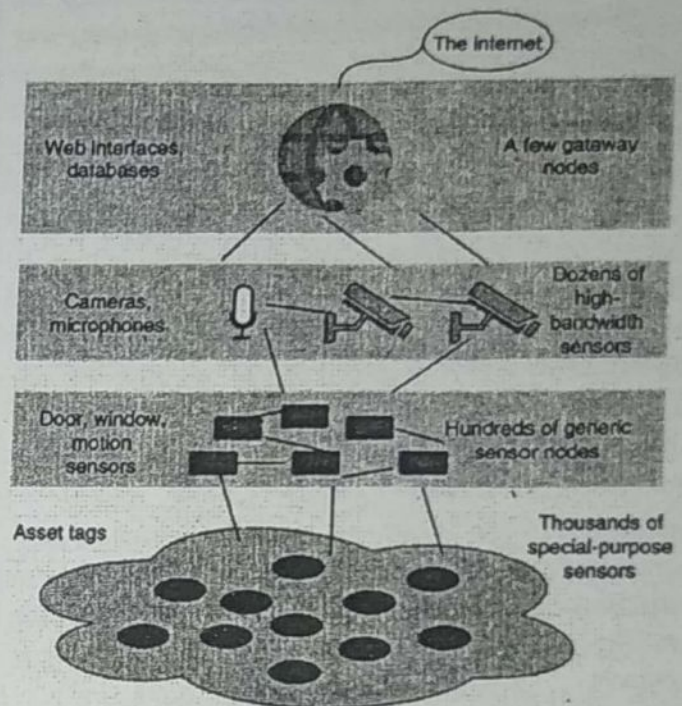


Fig. : Hierarchical deployment of a wireless sensor network

Each platform class handles different types of sensing (according to Hill et al., 2004). As sensors are foundational to both smart objects and sensor nodes, they are a crucial component of an IoT world. In fact, WSNs will facilitate the proliferation of many applications. The small, robust, inexpensive, and low-powered WSN sensors will bring the IoT to even the smallest objects installed in any kind of environment, at reasonable costs (IEC, 2014).

Q.32 What is mobile phone based sensors?

Ans. Mobile Phone Based Sensors : First of all, let us look at the mobile phone, which is ubiquitous and has many types of sensors embedded in it. In specific, the smartphone is a very handy and user friendly device that has a host of built in communication and data processing features. With the increasing popularity of smartphones among people, researchers are showing interest in building smart IoT solutions using smartphones because of the embedded sensors. Some additional sensors can also be used depending upon the requirements. Applications can be built on the smartphone that uses sensor data to produce meaningful results. Some of the sensors inside a modern smartphone are as follows :

- (i) The accelerometer senses the motion and acceleration of a mobile phone. It typically measures changes in velocity of the smartphone in three dimensions. There are many types of accelerometers :

In a mechanical accelerometer, we have a seismic mass in a housing, which is tied to the housing with a spring. The mass takes time to move and is left behind as the housing moves, so the force in the spring can be correlated with the acceleration. In a capacitive accelerometer, capacitive plates are used with the same setup. With a change in velocity, the mass pushes the capacitive plates together, thus changing the capacitance. The rate of change of capacitance is then converted into acceleration. In a piezoelectric accelerometer, piezoelectric crystals are used, which when squeezed generate an electric voltage. The changes in voltage can be translated into acceleration. The data patterns captured by the accelerometer can be used to detect physical activities of the user such as running, walking, and bicycling.

- (ii) The gyroscope detects the orientation of the phone very precisely. Orientation is measured using capacitive changes when a seismic mass moves in a particular direction.
- (iii) The camera and microphone are very powerful sensors since they capture visual and audio information, which can then be analyzed and processed to detect various types of contextual information.
- (iv) The magnetometer detects magnetic fields. This can be used as a digital compass and in applications to detect the presence of metals.
- (v) The GPS (Global Positioning System) detects the location of the phone, which is one of the most important pieces of contextual information for smart applications. The location is detected using the principle of trilateration [28]. The distance is measured from three or more satellites (or mobile phone towers in the case of A-GPS) and coordinates are computed.
- (vi) The light sensor detects the intensity of ambient light. It can be used for setting the brightness of the screen and other applications in which some action is to be taken depending on the intensity of ambient light. For example, we can control the lights in a room.
- (vii) The proximity sensor uses an infrared (IR) LED, which emits IR rays. These rays bounce back when they strike some object. Based on the difference in time, we can calculate the distance. In this way, the distance

to different objects from the phone can be measured. For example, we can use it to determine when the phone is close to the face while talking. It can also be used in applications in which we have to trigger some event when an object approaches the phone.

- (viii) Some smartphones such as Samsung's Galaxy S4 also have a thermometer, barometer, and humidity sensor to measure the temperature, atmospheric pressure, and humidity, respectively.

We have studied many smart applications that use sensor data collected from smartphones. For example, activity detection is achieved by applying machine learning algorithms to the data collected by smartphone sensors. It detects activities such as running, going up and down stairs, walking, driving, and cycling. The application is trained with patterns of data using data sets recorded by sensors when these activities are being performed

Many health and fitness applications are being built to keep track of a person's health continuously using smartphones. They keep track of users physical activities, diet, exercises, and lifestyle to determine the fitness level and give suggestions to the user accordingly. Wang et al. describe a mobile application that is based completely on a smartphone. They use it to assess the overall mental health and performance of a college student. To track the location and activities in which the student is involved, activity recognition (accelerometer) and GPS data are used. To keep a check on how much the student sleeps, the accelerometer and light sensors are used. For social life and conversations, audio data from a microphone is used. The application also conducts quick questionnaires with the students to know about their mood. All this data can be used to assess the stress levels, social life, behavior, and exercise patterns of a student.

Another application by McClernon and Choudhury detects when the user is going to smoke using context information such as the presence of other smokers, location, and associated activities. The sensors provide information related to the user's movement, location, visual images, and surrounding sounds. To summarize smartphone sensors are being used to study different kinds of human behavior and to improve the quality of human life.

Q.33 Explain medical sensors in detail.

Ans. Medical Sensors : The Internet of things can be really beneficial for health care applications. We can use sensors, which can measure and monitor various medical parameters

in the human body. These applications can aim at monitoring a patient's health when they are not in hospital or when they are alone. Subsequently, they can provide real time feedback to the doctor, relatives, or the patient. McGrath and Scanaill have described in detail the different sensors that can be worn on the body for monitoring a person's health.

There are many wearable sensing devices available in the market. They are equipped with medical sensors that are capable of measuring different parameters such as the heart rate, pulse, blood pressure, body temperature, respiration rate, and blood glucose levels. These wearables include smart watches, wristbands, monitoring patches, and smart textiles.

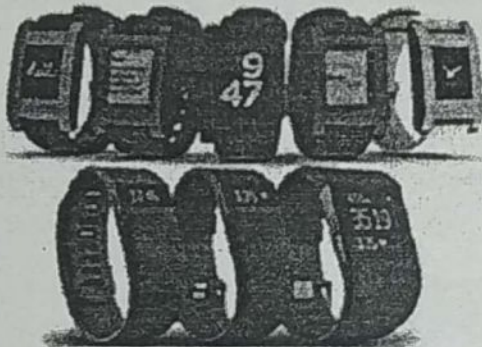


Fig. 1 : Smart watches and fitness trackers

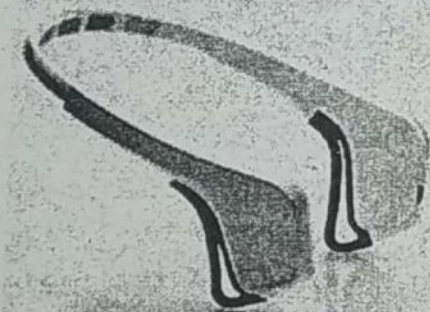


Fig. 2 : Brain sensing headband with embedded neurosensors

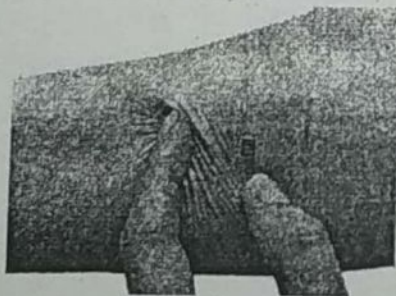


Fig. 3 : Embedded skin patches

Moreover, smart watches and fitness trackers are becoming fairly popular in the market as companies such as Apple, Samsung, and Sony are coming up with very innovative features. For example, a smart watch includes features such as connectivity with a smartphone, sensors such as an accelerometer, and a heart rate monitor (see figure 1).

Another novel IoT device, which has a lot of promise are monitoring patches that are pasted on the skin. Monitoring patches are like tattoos. They are stretchable and disposable and are very cheap. These patches are supposed to be worn by the patient for a few days to monitor a vital health parameter continuously. All the electronic components are embedded in these rubbery structures. They can even transmit the sensed data wirelessly. Just like a tattoo, these patches can be applied on the skin as shown in figure 3. One of the most common applications of such patches is to monitor blood pressure.

A very important consideration here is the context. The data collected by the medical sensors must be combined with contextual information such as physical activity. For example, the heart rate depends on the context. It increases when we exercise. In that case, we cannot infer abnormal heart rate. Therefore, we need to combine data from different sensors for making the correct inference.

Q.34 Write detail note on humidity sensors, classification and explain its working.

Ans. Humidity Sensor : Humidity sensor is one of the most important devices that has been widely in consumer, industrial, biomedical, and environmental etc. applications for measuring and monitoring humidity.

Humidity is defined as the amount of water present in the surrounding air. This water content in the air is a key factor in the wellness of mankind. For example, we will feel comfortable even if the temperature is 0°C with less humidity i.e. the air is dry.

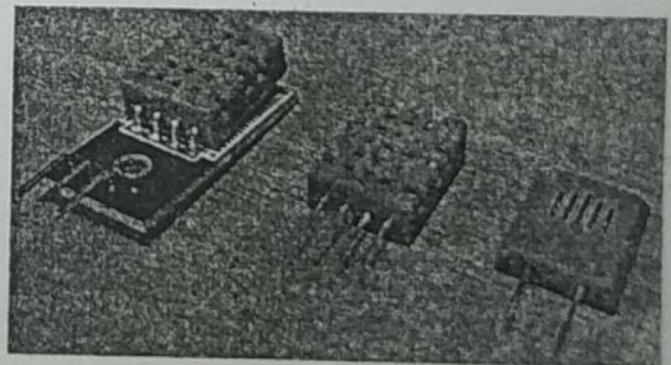


Fig.

But if the temperature is 100°C and the humidity is high i.e. the water content of air is high, then we will feel quite uncomfortable. Humidity is also a major factor for operating sensitive equipment like electronics, industrial equipment, electrostatic sensitive devices and high voltage

devices etc. Such sensitive equipment must be operated in a humidity environment that is suitable for the device.

Hence sensing, measuring, monitoring and controlling humidity is a very important task. Some of the important areas of application for sensing, measuring and controlling humidity are mentioned below :

Domestic: Sensing and controlling humidity in our homes and offices is important as higher humidity conditions will affect the blood flow. Other areas include cooking, indoor plantation etc.

Industrial: In industries like refineries, chemical, metal, or other industries where furnaces are used, high humidity will reduce the amount of oxygen in the air and hence reduces the firing rate. Other industries like food processing, textile, paper etc. also need control of humidity.

Agriculture: Irrigation techniques like drip irrigation need accurate moisture content for plants. Also, the moisture in the soil plays an important role in the proper growth of the plant. Other areas where humidity control is required is indoor vegetation.

Electronics and Semiconductor: Almost all electronic devices are rated with a range of humidity values in which they work as expected. Generally, this value will be something like 10% – 50% humidity. Semiconductor fabs (Fabrication Plants) should maintain very precise temperature and humidity values as even minute difference can show a huge impact in the production.

Medical: Medical equipment like ventilators, incubators, sterilizers etc. need humidity control. It is also used in pharmaceutical plants and biological processes.

All the above mentioned and many other applications need sensing of humidity and is done using humidity sensors.

Classification and Working Principles of Humidity Sensors : Humidity sensors are very important devices that help in measuring the environmental humidity. Technically, the device used to measure the humidity of the atmosphere is called hygrometer. Humidity sensors or hygrometers can be classified based on the type of humidity it is used for measuring.

Absolute Humidity (AH) Sensors or Relative Humidity (RH) Sensors : Humidity sensors can also be classified based on the parameter used for measuring humidity i.e. capacitive humidity sensors, electrical conductivity (or resistive) humidity sensors and thermal conductivity humidity sensors.

There are other types of humidity sensors or hygrometers like optical hygrometer, oscillating hygrometer and gravimetric hygrometer.

Let us see about different types of humidity sensors or hygrometers along with their working principles.

Capacitive Humidity Sensors: Humidity sensors based on capacitive effect or simply capacitive humidity sensors are one of the basic types of humidity sensors available.

They are often used in applications where factors like cost, rigidity and size are to be concern. In capacitive relative humidity (RH) sensors, the electrical permittivity of the dielectric material changes with change in humidity.

Working of Capacitive RH Sensors: A simple capacitive RH sensor can be made from an air filled capacitor as the moisture in the atmosphere changes its permittivity. But for practical applications, air as a dielectric is not feasible.

Hence, the space between the capacitor plates is usually filled with an appropriate dielectric material (isolator), whose dielectric constant varies when it is subjected to change in humidity.

The common method of constructing a capacitive RH sensor is to use a hygroscopic polymer film as dielectric and depositing two layers of electrodes on the either side.

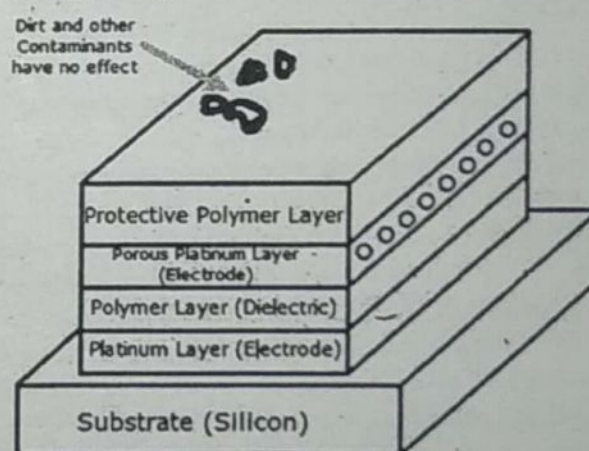


Fig. 1

Another way to use the capacitive RH sensors is to observe the changes in the frequency of the oscillator constructed using a capacitor with RH sensitive test subject as dielectric. This setup is often employed in pharmaceutical products.

The test samples like medical tablets are placed between two plates (which form the capacitor electrodes) to form a capacitor in the LC oscillator circuit. The frequency of the oscillator changes with humidity surrounding the test sample.

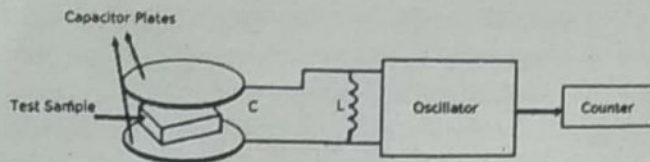


Fig. 2

Let us see the construction of a thin thermostat polymer film based capacitive RH Sensor. It is fabricated on a silicon substrate. On this substrate, two metal electrodes made of either aluminium, platinum or chromium are deposited. The shape of these electrodes is carved out such that, the electrodes form an interdigitized pattern.

On top of this layer, a dielectric layer is deposited. The following image shows a top and cross section view of the capacitive humidity sensor. Note that two temperature sensitive resistors are deposited on the same substrate to provide temperature compensation.

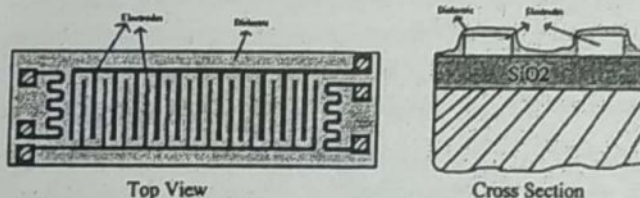


Fig. 3

Advantages of Capacitive Humidity Sensors :

- (i) The output voltage is near linear.
- (ii) They provide stable results over long usage.
- (iii) Can detect wide range of RH.

Disadvantages of Capacitive Humidity Sensors : The distance from the sensor and signalling circuit is very limited.

Applications of Capacitive Humidity Sensors : Capacitive humidity sensors are used in a wide range of applications including but not limited to:

- (i) HVAC Systems
- (ii) Printers and Fax Machines
- (iii) Weather Stations
- (iv) Automobiles
- (v) Food Processing
- (vi) Refrigerators, Ovens and Dryers

Resistive Humidity Sensors (Electrical Conductivity Sensors): Resistive humidity sensors are another important type of humidity sensors that measure the resistance (impedance) or electrical conductivity. The principle behind

resistive humidity sensors is the fact that the conductivity in non-metallic conductors is dependent on their water content.

Working of Resistive Humidity Sensors: The resistive humidity sensor is usually made up of materials with relatively low resistivity and this resistivity changes significantly with changes in humidity. The relationship between resistance and humidity is inverse exponential. The low resistivity material is deposited on top of two electrodes.

The electrodes are placed in interdigitized pattern to increase the contact area. The resistivity between the electrodes changes when the top layer absorbs water and this change can be measured with the help of a simple electric circuit.

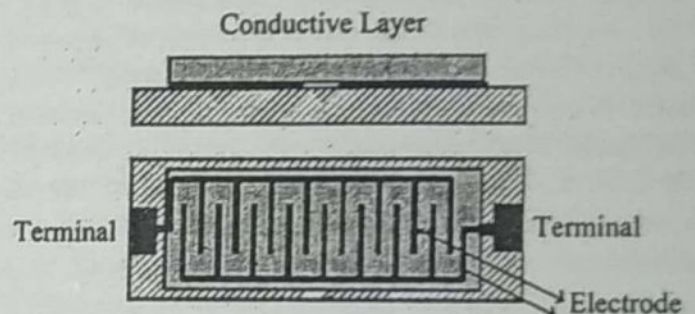


Fig. 4

Some of the commonly used materials are salt, specially treated substrates, solid polyelectrolytes and conductive polymers. Modern resistive humidity sensors are coated with ceramic substance to provide extra protection. The electrodes in the sensor are usually made of noble metals like gold, silver or platinum.

Advantages of Resistive Humidity Sensors :

- (i) Low Cost
- (ii) Small Size
- (iii) The distance between the sensor and signal circuit can be large (suitable for remote operations).
- (iv) Highly interchangeable as there are no calibration standards.

Disadvantages of Resistive Humidity Sensors :

- (i) Resistive humidity sensors are sensitive to chemical vapors and other contaminants.
- (ii) The output readings may shift if used with water soluble products.

Applications of Resistive Humidity Sensors: Resistive or electrical conductive humidity sensors are low cost sensors with relatively small size. They are often used in several industrial, domestic or residential and commercial applications.

Thermal Conductivity Humidity Sensors: Thermal conductivity humidity sensors are also known as absolute humidity (AH) sensors as they measure the absolute humidity. Thermal conductivity humidity sensors measure the thermal conductivity of both dry air as well as air with water vapor. The difference between the individual thermal conductivities can be related to absolute humidity.

Working of Thermal Conductivity Humidity Sensors: The best component to accomplish thermal conductivity based humidity sensor is thermistor. Hence, two tiny thermistors with negative temperature coefficient are used to form a bridge circuit.

In that, one thermistor is hermetically sealed in a chamber filled with dry nitrogen while the other is exposed to open environment through small venting holes. When the circuit is powered on, the resistance of the two thermistors are calculated and the difference between those two values is directly proportional to absolute humidity (AH).

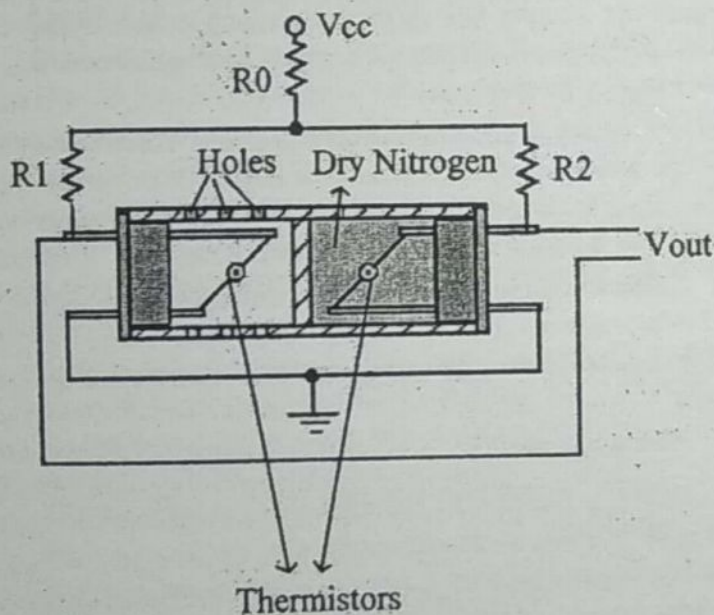


Fig. 5

Advantages of Thermal Conductivity Humidity Sensors :

- (i) Suitable for high temperature environments and high corrosive situations.
- (ii) Very durable.
- (iii) Higher resolution compared to other types.

Disadvantage of Thermal Conductivity Humidity Sensors : Exposure to any gas with thermal properties different than nitrogen might affect reading measurement.

Applications of Thermal Conductivity Humidity Sensors : Some of the common applications of thermal conductivity humidity sensors are:

- (i) Drying kilns
- (ii) Pharmaceutical plants
- (ii) Owens
- (iv) Clothes dryers and drying machines
- (v) Food dehydration

Q.35 Explain the working of ultrasonic sensors. Define proximity sensor and also write advantages and disadvantages of ultrasonic sensors.

Ans. Ultrasonic Sensors : Ultrasonic transducers convert ultrasound waves to electrical signals and vice versa. These devices work on a principle similar to that used by transducers in radar and sonar systems, which evaluate the attributes of the target object by processing the echo signals from radio or sound waves, respectively. Ultrasonic sensors consist of two parts: a transmitter and receiver, which create a transducer that converts ultrasound waves into electrical signals (A/C) or vice versa. The transceiver vibrates and creates an ultrasonic wave that is transmitted and travels until it hits an object and is reflected back to the receiver. The interval between the signal being sent and received is typically referred to as time-of-flight (TOF) and depends on the distance the ultrasonic wave travels until it is reflected. The basic equation: time is equal to distance divided by speed, can be used to measure fluid level, fluid identification/concentration, and distance.

Ultrasonic Transducer Technology: The transceiver vibrates and creates an ultrasonic wave using piezoelectric transducers or capacitive transducer technologies. Piezoelectric crystals change size and shape when a voltage is applied: A/C voltage makes them oscillate at the same frequency and produce ultrasonic sound. Capacitive transducers use electrostatic fields between a conductive diaphragm and a backing plate. A single ultrasonic transducer can both generate and receive a signal, but the two functions are often separated in order to optimize the performance of each task.

Proximity Sensor: A proximity sensor is a sensor that detects the presence of nearby objects without requiring physical contact. Ultrasonic sensors are typically used as a proximity sensor, setting a threshold distance which can determine whether an object is an obstacle. This kind of proximity sensor is commonly used in the robotics industry.

Distance Measurement: Ultrasonic sensors are ideal tools for measuring distance without requiring contact with the

object and are an efficient method of precisely measuring small distances. As the distance from the object is proportional to the time interval between transmitting and receiving signals, a simple analysis of this data can reveal changes in the sensor's distance to the object.

Liquid Level: There is a wide range of applications for liquid level sensors, each with its specific characteristics. The most typical application involves measurement of the tank liquid level of certain liquids, where the tank height is known. In this case, sound waves hit the surface of the liquid, and the sound echo signals are reflected back towards the sensor. The time that it takes for the sound wave to return to the sensor is directly proportional to the distance between the piezoelectric sensor and the liquid in the tank. This time period is measured by the sensor which is then used to calculate the level of liquid in the tank. The speed of sound waves can sometimes be affected by the variations in temperature and the sensor design should accommodate these variations,

Fluid Identification/Concentration: Fluid identification/concentration involves changes in the propagation velocity of sounds between different liquids. The sensor measures the TOF of a known distance e.g., tank width, and the microcontroller calculates the fluid speed of sound. This value can be compared to the values in a look-up-table, which is used to identify the liquid.

Advantages of Ultrasonic Sensors:

- (i) Ultrasonic sensors produce ultrasonic frequencies that humans cannot hear, making them ideal for use in environments that require low noise levels.
- (ii) An ultrasonic sensor response is not dependent upon the surface color or optical reflectivity of the object e.g., a glass plate or a shiny aluminum plate.
- (iii) These sensors don't require much electricity, are simple in design, and are relatively inexpensive.
- (iv) Ultrasonic sensors with digital (on/off) outputs have excellent repeat sensing accuracy. It is possible to ignore immediate background objects, even at long sensing distances because switching hysteresis (the physical property value lags behind changes in the causation effect) is relatively low.

Disadvantages of Ultrasonic Sensors:

- (i) Ultrasonic sensors have a minimum sensing distance.
- (ii) Changes in the environment, such as temperature, pressure, humidity, air turbulence, and airborne particles affect ultrasonic responses.

- (iii) Targets with low density, such as foam and cloth, tend to absorb sound energy and these materials may be difficult to sense at long ranges.
- (iv) Ultrasonic sensors must be in the direct line of sight of the surface of the object in order to receive an adequate sound echo. Additionally, the reliability of these sensors requires a minimum object surface area.
- (v) Smooth surfaces reflect sound waves more efficiently than rough surfaces.

Q.36 Write different types of temperature sensors.

Ans. Temperature sensors are available of various types, shapes, and sizes. The two main types of temperature sensors are:

1. Contact Type Temperature Sensors: There are a few temperature meters that measure the degree of hotness or coolness in an object by being in direct contact with it. Such temperature sensors fall under the category contact-type. They can be used to detect solids, liquids or gases over a wide range of temperatures.

2. Non-Contact Type Temperature Sensors: These types of temperature meters are not in direct contact of the object rather, they measure the degree of hotness or coolness through the radiation emitted by the heat source.

There are different types of temperature sensors that have sensing capacity depending upon their range of application. Different types of temperature sensors are as follows:

- (i) Thermocouples
- (ii) Resistor temperature detectors
- (iii) Thermistors
- (iv) Thermometers
- (v) Semiconductors
- (vi) Infrared sensors

(i) Thermocouples: Thermocouple sensor is the most commonly used temperature sensor and it is abbreviated as TC. This sensor is extremely rugged, low-cost, self-powered and can be used for long distance. There are many types of temperature sensors that have a wide range of applications.

A thermocouple is a voltage device that indicates temperature by measuring a change in the voltage. It consists of two different metals: opened and closed. These metals work on the principle of thermo-electric effect. When two dissimilar metals produce a voltage, then a thermal difference

exists between the two metals. When the temperature goes up, the output voltage of the thermocouple also increases.

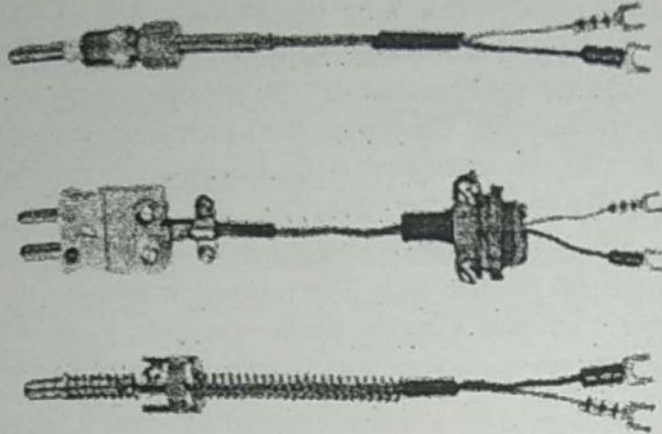


Fig. 1 : Thermocouples

This thermocouple sensor is usually sealed inside a ceramic shield or a metal that protects it from different environments. Some common types of thermocouples include K, J, T, R, E, S, N and B. The most common type of thermocouples is J, T and K type thermocouples, which are available in pre-made forms.

The most important property of the thermocouple is nonlinearity – the output voltage of the thermocouple is not linear with respect to temperature. Thus, to convert an output voltage to a temperature, it requires mathematical linearization.

(ii) **Resistor Temperature Detector (RTD):** RTD sensor is one of the most accurate sensors. In a resistor temperature detector, the resistance is proportional to the temperature. This sensor is made from platinum, nickel, and copper metals. It has a wide range of temperature measurement capabilities as it can be used to measure temperature in the range between -270°C to $+850^{\circ}\text{C}$. RTD requires an external current source to function properly. However, the current produces heat in a resistive element causing an error in the temperature measurements. The error is calculated by this formula:

$$\Delta T = P \times S$$

Where, 'T' is temperature. 'P' is one squared power produced and 'S' is a degree C/mill watt.

There are different types of techniques to measure temperature by using this RTD. They are two wired, three-wired and four-wired method. In a two-wired method, the current is forced through the RTD to measure the resulting voltage. This method is very simple to connect and implement; and, the main drawback is the lead resistance is the part of the measurement which leads to erroneous measurement.

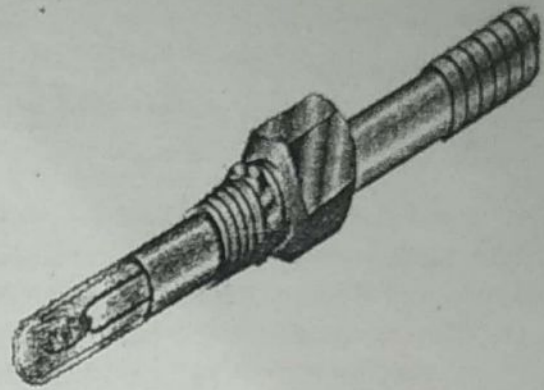


Fig. 2 : Resistor Temperature Detector (RTD)

Three-wired method is similar to the two-wired method, but the third wire compensates for the lead resistance. In a four-wired method, the current is forced on one set of the wires and the voltage is sensed on the other set of wires. This four-wired method completely compensates for the lead resistance.

(iii) **Thermistors:** Another type of sensor is a thermistor temperature sensor, which is relatively inexpensive, adaptable, and easy to use. It changes its resistance when the temperature changes like RTD sensor.



Fig. 3 : Thermistors

Thermistors are made from manganese and oxides of nickel, which make them susceptible to damages. So, these materials are called ceramic materials. This thermistor offers higher sensitivity than the resistor temperature detectors. Most of the thermistors have a negative temperature coefficient. It means, when the temperature increases the resistance decreases.

(iv) **Thermometers:** A thermometer is a device used to measure the temperature of solids, liquids, or gases. The name

thermometer is a combination of two words: thermo – means heat, and meter means to measure. Thermometer contains a liquid, which is mercury or alcohol in its glass tube. The volume of the thermometer is linearly proportional to the temperature – when the temperature increases, the volume of the thermometer also increases.

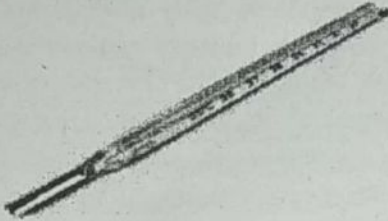


Fig. 4 : Thermometers

When the liquid is heated it expands inside the narrow tube of the thermometer. This thermometer has a calibrated scale to indicate the temperature. The thermometer has numbers marked alongside the glass tube to indicate the temperature when the line of mercury is at that point. The temperature can be recorded in these scales: Fahrenheit, Kelvin or Celsius. Therefore, it is always desirable to note for which scale the thermometer is calibrated.

(v) **Semiconductor Sensors:** Semiconductor sensors are the devices that come in the form of ICs. Popularly, these sensors are known as an IC temperature sensor. They are classified into different types: Current output temperature sensor, Voltage output temperature sensor, Resistance output silicon temperature sensor, Diode temperature sensors and Digital output temperature sensor. Present semiconductor temperature sensors offer high linearity and high accuracy over an operating range of about 55°C to $+150^{\circ}\text{C}$. However, AD590 and LM35 temperature sensors are the most popular temperature sensors.

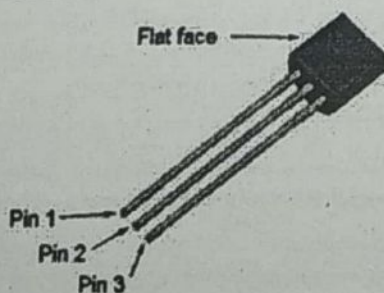


Fig. 5 : Semiconductor Sensors

(vi) **IR sensor:** IR sensor is an electronic instrument which is used to sense certain characteristics of its surroundings by either emitting or detecting IR radiation. These sensors are non-contacting sensors. For example, if you hold an IR sensor in front of your desk without establishing any contact, the sensor detects the temperature of the desk based on the merit

of its radiation. These sensors are classified into two types such as thermal infrared sensors and quantum infrared sensors.

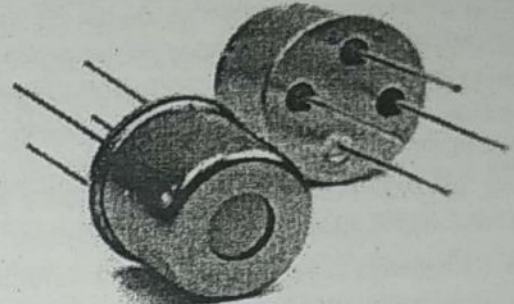


Fig. 6 : IR Sensor

Thus, this is all about different types of temperature sensors. The cost of the temperature sensor depends on the type of work it is intended for. However, the accuracy of the sensor will decide the price. So, the cost depends on the accuracy of the temperature sensor. Present temperature sensors intended at reducing the cost as well as efficiency.

Q.37 Explain the following with respect to the TinyOS:

- (i) *Programming model and development environment*
- (ii) *Scheduling*
- (iii) *Memory management and performance*
- (iv) *Communication and networking protocols support*
- (v) *Simulation support*
- (vi) *Security*

Ans.(i) Programming Model and Development Environment : TinyOS supports an event-driven concurrency model which consists of split-phase interfaces, deferred computation, and asynchronous events. TinyOS is programmed in NesC for memory limitations of sensor networks which are similar but not compatible with the C language. It allows writing pieces of reusable code which explicitly indicates their dependencies. Also, TinyOS uses a mechanism called trickle. Trickle is an algorithm used for propagating and maintaining code updates when needed. Trickle applies a polite gossip policy, where nodes occasionally broadcast code to all neighbour nodes, and remain silent. When a node hears an older summary of its own, it broadcasts an update rather than sending a network signal with packets. Then, the algorithm manages the process of sending, so each node only hears a small trickle of packets which is just enough to stay up to date. Trickle propagates new code within

seconds and makes the maintenance cost less in terms of time (propagation of new code to all neighbours nodes). The primary challenge in TinyOS development is the creation of flexible and reusable components.

(ii) **Scheduling** : The task scheduler in TinyOS is a simple non-preemptive first-in-first-out (FIFO) scheduler using a bounded size scheduling data structure. The TinyOS scheduler sets the processor to sleep when the tasks are completed, to maximise CPU utilization as well as the OS performance.

(iii) **Memory Management and Performance** : TinyOS uses static memory allocation with memory protection. There are no concepts of dynamic memory allocation such as hidden heaps, dynamic memory, or function pointers because TinyOS programmes are organised in components and are written in NesC language. TinyOS has a small footprint as it uses a non-preemptive FIFO task scheduling. It applies synchronisation clock in software, which increases the number of entries in the task queue at compile time when the system begins with 1, 32 kHz, or 1 MHz.

(iv) **Communication and Networking Protocols Support** : TinyOS has built-in support for common network protocols such as transmission control protocol (TCP), user datagram protocol (UDP), ICMPv6, IPv6, IPv6 over low-power WPAN (6LoWPAN), IPv6 routing protocol for low-power and lossy networks (RPL), and constrained application protocol (CoAP), in addition to hydrogen routing protocol that is used for reliable communication.

(v) **Simulation Support** : To validate the analysis model of TinyOS applications, a TinyOS simulation (TOSSIM) simulation environment has been developed. TinyOS simulation (TOSSIM) provides a high flexibility simulation of TinyOS applications which work by replacing components with simulation implementations. Moreover, TOSSIM provides developers an integrated environment of the network and troubleshooting capabilities. Server-side applications can be connected to a TOSSIM proxy only if it is a real sensor network. Hence, facilitating the transition between the simulation and real deployments. TOSSIM also provides support integration for troubleshooting and debugging applications directly on the mote. Unfortunately, TOSSIM does not support gathering power measurements.

(vi) **Security** : TinyOS uses TinySec library which was developed using the NesC programming language and implemented by the link layer to provide confidentiality, message authentication, integrity, and semantic security. The default block cipher encryption in TinySec is Skipjack algorithm that is used with cipher block chaining (CBC-CS)

method. Skipjack has an 80 bit key length that provides immunity to brute force attacks. Skipjack generates message authentication code (MAC) method which utilises CBC-MAC. However, CBC-MAC has security lacks since it furnishes semantic security with an 8 B introduction vector which includes only a 2 B counter overhead per packet. TinySec holds <10% energy, inactivity, and transfer speed overhead.

Q.38 Write detailed note on Contiki OS.

Ans. Contiki OS : Contiki is an open-source non-Linux-based OS for low-end IoT devices designed especially for IoT. It is lightweight, highly portable, and multitasking OS that runs on tiny low-power microcontrollers with minimal memory. Contiki OS is written in C programming language. It uses 2 kB of RAM and 40 kB of read-only memory (ROM). Nowadays, Contiki can be run on various hardware platforms such as Alf and Vegard RISC processor (AVR), MSP430, and Z80.

Architecture and Kernel Models : In contrast to TinyOS, Contiki OS has a modular architecture. The core of Contiki OS mainly consists of multiple lightweight event schedulers and a polling mechanism. The event schedule is responsible for dispatching events to run processes and periodically calls processes polling handlers, which identifies the action of the polled process. On the other hand, the polling mechanism identifies high priority events. Polling mechanism is used by processes that operate near the hardware to check the status updates of hardware devices. All processes that implement a poll handler are requested in order of their priority. Fig. shows the architecture of Contiki OS. Contiki OS contains sensor data handling, communication protocols, and device driver services. Each service has its interface and implementation.

Programming Model and Development Environment : Unlike TinyOS, the programming models in Contiki support both multithreading and event-driven using protothreads. The main advantage of protothreads is their very minimal memory overhead with no extra stack for a thread. Since events run to completion, Contiki does not allow interruption of handlers to post new events, and it does not allow process synchronisation. Programming models with Contiki are defined by events in a way that all tasks are executed in the same context. Protothreads mechanism runs on top of the event-driven kernel. A protothread process is invoked whenever a process receives an event, and the protothreads mechanism

decides which memory should be implemented by system libraries which are connected with programmes. Programmes can be connected with libraries

d. Contiki is

in three ways. The first way, the programmes can be statically connected with libraries that are part of Contiki core.

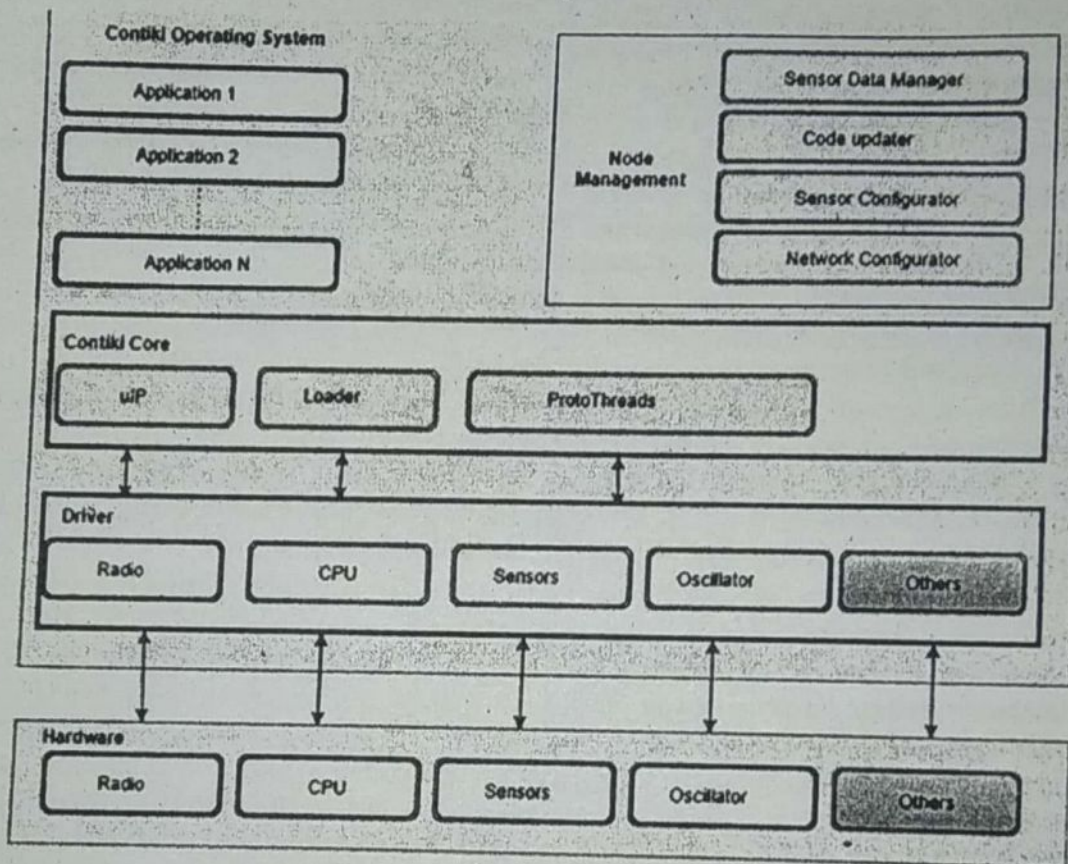


Fig. : Architecture of Contiki OS

Second, programmes can be statically linked with libraries that are part of the loadable programme. Third, programmes can call services using a specific library. Libraries that are applied as services can be replaced dynamically at the running time. Consider a programme that uses the *memcpy()* and *atoi()* functions to copy memory and to convert strings to integers, respectively. The *memcpy()* function is a frequently used C library function; whereas *atoi()* is used less often. Therefore, in this example, *memcpy()* has been included in the Contiki core but not *atoi()*. The *memcpy()* function will be linked against its static address in the core when the programme is linked to produce a binary. The object code for the part of the C library that implements the *atoi()* function must, however, be included in the binary programme. Moreover, Contiki uses loadable modules to perform dynamic code reprogramming and upgrading. With loadable modules, only specific parts of the codes need to be modified when a

single programme is changed. Besides, Contiki provides a command-line shell which is useful during development and debugging of Contiki systems.

Scheduling : The scheduling used in Contiki OS is similar to TinyOS, in which both use FIFO scheduling strategy. In Contiki, all event-driven scheduling is done at a single level and events (preemptive multitasking); events are executed as they arrive.

Memory Management and Performance : Unlike TinyOS, Contiki supports dynamic allocation or deallocation of memory through *memb()* and *mmem()* as well as *malloc()*. The *memb()* memory block allocator is the most frequently used. The *mmem()* managed memory allocator is used infrequently and it uses the standard C library *malloc()* heap memory allocator. In addition, Contiki uses Contiki coffee file system technique for data storage inside the sensor network. It allows multiple files to exist on the same physical onboard flash memory.

Communication and Networking Protocols Support: Contiki OS supports many protocols such as CoAP and the message queue telemetry transport (MQTT). In addition to that, the two main communication stacks are uIP and Rime stack that consists of a set of custom lightweight protocols for power constrained wireless networks. Contiki supports a full IP network stack with standard IP protocols such as UDP, TCP, and HTTP. Also, it has support for 6LoWPAN adaptation layer, the RPL IPv6 multi-hop routing protocol, and the CoAP RESTful application-layer protocol.

Simulation Support: Cooja simulator supports Contiki, which is a useful tool for Contiki OS application development. Cooja makes simulation colossally less demanding by providing a simulation environment to allow testing of code before running it on the target hardware devices.

Security: Contiki OS uses ContikiSec transport layer security (TLS)/datagram transport layer security (DTLS), which is a secure network layer, and contains three modes: authentication, confidentiality, and integrity in communication. ContikiSec uses low-energy utilisation and security while complying with a little memory footprint.

Power Consumption: Contiki is intended to run on low-power devices that may need to keep running for quite long time on batteries. To help the improvement of low-power devices power consumption, Contiki provides software-based power profiling mechanism for estimating the system power utilization and for knowing where the power was consumed giving power awareness.

Supporting Multimedia: Contiki supports full IP network stack protocols such as UDP, TCP, and HTTP that are used to stream multimedia contents. The frameworks available for video codecs and multimedia streaming are limited in this OS and has no extended support. Moreover, RTP protocol is not found in the base of Contiki.

More about Contiki OS: One of the essential features of Contiki is dynamic loading, which is the ability to link modules at run time. Contiki transferred nodes can be battery-operated because of the ContikiMAC radio duty cycling mechanism which allows nodes to sleep between each relayed message. Unlike TinyOS that has no blocking operations, Contiki provides some conditional blocking of functions in a sequential instruction block.

Q.39 Explain RIOT OS in detail.

Ans. Real-time OS for IoT (RIOT) OS: The RIOT is known as the friendly OS for the IoT. RIOT is an open-

source non-Linux-based OS specialised for low-end IoT devices with a minimum of 1.5 kB of RAM and 5 kB of ROM. RIOT provides a uniform abstraction over the details of different IoT hardware. It was developed by a grassroots community using C programming language. RIOT can run on various platforms including embedded systems, and it is easy to use. It supports many functionalities such as interruption handling, memory management, IPC, and synchronisation. Moreover, RIOT has many advantages such as reliability, predictability, performance, and scalability.

Architecture and Kernel Models: In contrast to the other OSs such as TinyOS or Contiki. RIOT has a microkernel architecture, which has been designed to work on several IoT platforms with different CPU architectures (32 bit, 16 bit, 8 bit) such as ARMv7, ARM Cortex-M0+, MSP430, and some recent AVR microcontrollers. The microkernel architecture of RIOT OS was developed using C++, and it supports full multithreading that provides a developer-friendly API and allows C++ and ANSI C application programming. RIOT kernel will never crash because it supports error device drivers. The architecture design of RIOT also contains POSIX compliance. Fig. shows the structure of RIOT, which is divided into four layers. The first layer is the kernel; which consists of the scheduler, inter-process communication, threading, thread synchronisation, supporting data structures and type definitions. The second layer is platform specific code (CPU boards), which contains the configuration for that particular CPU. The third layer is device drivers, which consist of the drivers for external devices such as network interfaces, sensors, and actuators. The fourth layer comprises of libraries, network code, and applications for demonstrating features and testing. Moreover, this layer includes a collection of scripts for various tasks as well as predefined environment documentation (doc).

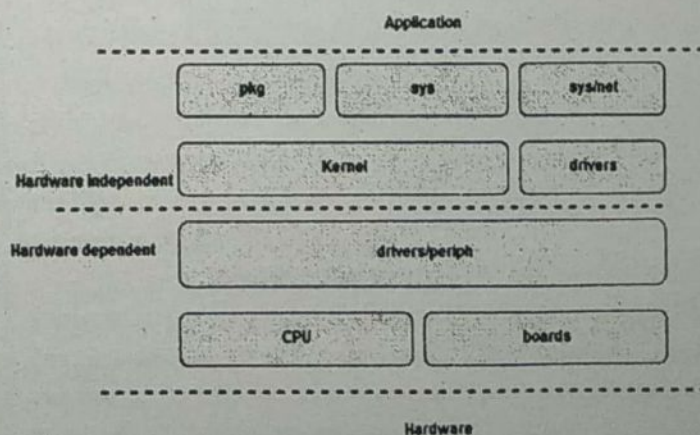


Fig. : Architecture of RIOT OS

Scheduling : Together with Contiki, RIOT implements preemptive priority-based and tickless scheduling, where each task has a priority in execution that helps the scheduler to select the highest priority task to run on CPU. RIOT tasks with the highest priority are executed first, and if there are more than one high priority tasks, a round-robin (RR) mechanism will be used.

Memory Management and Performance : In RIOT OS, both dynamic and static memory allocations are provided for applications. RIOT OS does not have a memory management unit (MMU) or floating point unit. However, it has a low memory footprint in the order of a few kB.

Communication and Networking Protocols Support: RIOT OS supports several networking protocols including TCP/IP v4 and v6 and the latest standards for connecting constrained systems to the internet engineering taskforce (IETF) 6LoWPAN. In addition to that, RIOT has built-in support for other IoT-related network protocols such as CoAP and RPL.

Simulation Support : At the time of writing this paper, RIOT OS does not have a simulator. Rather, we can have a full-scale simulation for RIOT applications from Contiki-Cooja simulator for IoT.

Programming Model and Development Environment: RIOT is similar to Contiki that it also supports preemptive multithreading. RIOT is developed using standard programming languages such as ANSI C and C++. With RIOT, developers can code the application once and run it on various IoT hardware devices. Moreover, RIOT provides common programmer APIs such as Berkeley software distribution (BSD) sockets or POSIX thread (pthread) functionalities. Besides, RIOT can run and debug the same as in Linux and MacOS using a set of popular debugging tools such as GNU Debugger (GDB) and Valgrind. The C++ programming capabilities used in RIOT allow RIOT to use powerful libraries such as the Wiselib, which contains algorithms for routing, clustering, time sync, localisation, and security. RIOT has other programming features such as dynamic linking support, Python interpreter, and energy profiler. Also, RIOT provides virtualisation, where the code and application can run as a simple Unix process. RIOT uses Wireshark for packet analysing.

Security : RIOT supports powerful attack detection capabilities called secure cyber-physical ecosystem (CPS). CPS is a system that interacts, monitors, and controls smart objects through complicated processes. When an attack is detected, then the reaction to it occurs.

Power Consumption : The simplicity of microkernel architecture of RIOT is the main characteristic to enable maximum energy efficiency. RIOT context switching can happen in two situations. The first situation is when a corresponding kernel operation gets called by itself such as a mutex locking. The second situation is when an interruption happens in a thread switch.

Q.40 Explain Lite OS in detail.

Ans. Lite OS : LiteOS is an open-source Linux-based lightweight OS designed to run on low-power devices. This makes LiteOS suitable for a wide range of areas including wearable, smart homes, connected vehicles, and microcontrollers. LiteOS can be installed on devices that run by Google Android OS, and it can connect with other third-party devices. It is developed purposely to provide a Unix-like OS for IoT developers and to provide programmers with familiar programming paradigms such as a hierarchical file system developed using LiteC programming language and a Unix-like shell.

Architecture and Kernel Models : In contrast to RIOT, LiteOS has a modular architecture divided into three subsystems; LiteShell, LiteFS, and the kernel as shown in Fig. LiteShell is a Unix-like shell that provides support for shell commands such as file management, process management, and debugging. LiteShell resides on a base station or a PC. This leverage allows more complex commands as the base station, or PC has abundant resources. The LiteShell can only be used with user intervention. Some local processing is done on the user command by the shell and then transmitted wirelessly to the intended IoT node. The IoT node does the required processing of the command and sends a response back which is then displayed to the user. When a mote does not carry out the commands, an error code is returned. The second architectural component of LiteOS is its file system, LiteFS, which consists of sensor nodes as a file and it mounts a sensor network as a directory and then lists all one hop sensor nodes as a file. A user on the base station can use this directory structure just as the traditional Unix directory structure and can also use legitimate commands. The third subsystem of LiteOS is the kernel which resides on the IoT node. The kernel supports concurrency multithreading, dynamic loading, and uses RR and priority scheduling, which allows developers to register event handlers through callback functions.

Programming Model and Development Environment: LiteOS is a multitasking OS that supports multithreading. In LiteOS, processes run applications in separate threads. Each thread has its allocated memory which helps in protecting the memory. LiteOS also provides support for event handling. Also, it supports dynamic reprogramming and replacement mechanism through the user application. Reprogramming can be performed either if the source code of the OS is available or not. If it is available, it will be easily recompiled with new memory settings, and all pointers of the old version will be redirected, whereas if the source code is not available, it uses a differential patching mechanism to upgrade the older version. Also, LiteOS supports online debugging including variable watches and a vast number of breakpoints. Additionally, it contains extensive development libraries.

Scheduling: LiteOS implements both priority-based and RR scheduling in the kernel. The task to be executed is chosen from the ready queue using priority-based scheduling. When a task requires a resource that is not available currently, the task allows interrupts and goes to sleep mode.

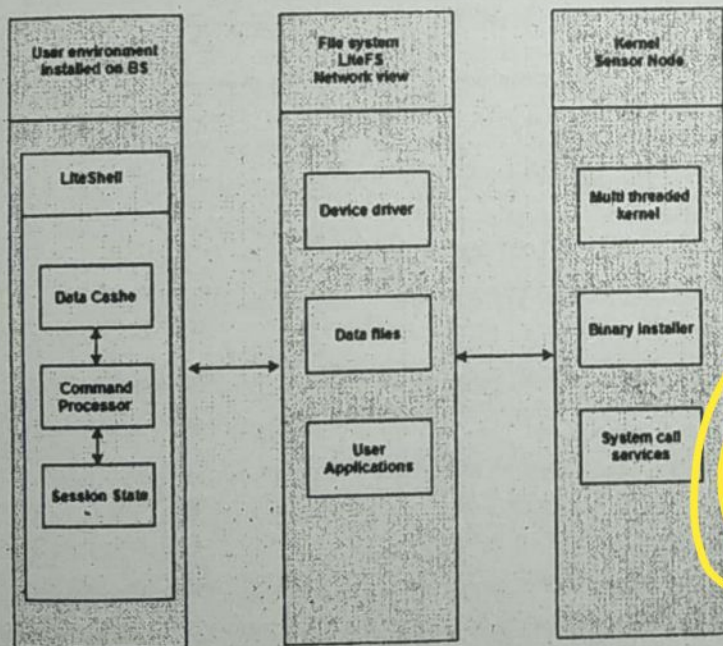


Fig. : Architecture of LiteOS

Memory Management and Performance: LiteOS implements dynamic memory allocation with an almost zero overhead through system calls using *malloc()* and *free()* API functions. The *malloc()* function allocates memory using a pointer. If the size of memory is zero, then *malloc()* returns to either NULL or a unique pointer value that can be passed to *free()* function. The *free()* function frees the memory space, which must have been returned by a previous call to *malloc()* function. This enables adapting the size of dynamic memory as required by an application.

Communication and Networking Protocols Support: LiteOS does not have any built-in networking protocols that support real-time applications. LiteOS provides support for long-distance connection based on technologies such as long-term evolution (LTE) and NodeB (NB)-IoT, and short-distance connection based on communication protocols such as ZigBee and 6LoWPAN emulate LiteOS on physical IoT devices. AVRORA is a set of simulation tools for programmes written for the AVR microcontroller. AVRORA contains an adaptable system for simulating and prototyping programmes, which allows Java API experimentation, profiling, and investigation.

Security: In terms of security, LiteOS provides independent user space and application separation through a set of system calls. The authentication mechanism is needed between the base station and mounted nodes, especially low-cost authentication mechanisms. To ensure the security of communications between the sensors and systems, LiteOS has a security component by embedding Hi3519 chip Huawei Lite OS that can be implemented to security cameras and portable high-definition camera.

Power Consumption: LiteOS supports ultra-low-power consumption; it can be used to run MicaZ nodes having 128 B of flash memory, 4 kB of RAM, and 8 MHz CPU. LiteOS battery can power a device for five years or more.

Supporting Multimedia: LiteOS does not support any implementation of networking protocols that support multimedia applications.

Q.41 Write the difference between arduino and raspberry Pi.

Ans. Raspberry Pi and Arduino are quite different boards. Each board has its own advantages and disadvantages.

Arduino was invented by Massimo Benzi in Italy. Arduino was a simple hardware prototyping tool. While raspberry pi as invented by Eben Upton at the University of Cambridge in the United Kingdom for improving the programming skills of his students.

These both teaching tools are suitable for beginners, hobbyists. The main difference between them is Arduino is microcontroller board while raspberry Pi is a mini computer. Thus Arduino is just a part of raspberry Pi. Raspberry Pi is good at software applications, while Arduino makes hardware projects simple.

| S.No. | Raspberry Pi | Arduino |
|-------|--|---|
| 1. | It is a mini computer with Raspbian OS. It can run multiple programs at a time. | Arduino is a microcontroller, which is a part of the computer. It runs only one program again and again. |
| 2. | It is difficult to power using a battery pack. | Arduino can be powered using a battery pack. |
| 3. | It requires complex tasks like installing libraries and software for interfacing sensors and other components. | It is very simple to interface sensors and other electronic components to Arduino. |
| 4. | It is expensive. | It is available for low cost. |
| 5. | Raspberry Pi can be easily connected to the internet using Ethernet port and USB Wi-Fi dongles. | Arduino requires external hardware to connect to the internet and this hardware is addressed properly using code. |
| 6. | Raspberry Pi did not have storage on board. It provides an SD card port. | Arduino can provide onboard storage. |
| 7. | Raspberry Pi has 4 USB ports to connect different devices. | Arduino has only one USB port to connect to the computer. |
| 8. | The processor used is from ARM family. | Processor used is from AVR family Atmega328P. |
| 9. | This should be properly shutdown otherwise there is a risk of files corruption and software problems. | This is a just plug and play device. If power is connected it starts running the program and if disconnected it simply stops. |
| 10. | The recommended programming language is python but C, C++, python, ruby are pre-installed. | Arduino uses Arduino, C/C++. |

ARCHITECTURE AND REFERENCE MODEL

3

IMPORTANT QUESTIONS

PART-A

Q.1 What do you mean by REST?

Ans. REST stands for Representational State Transfer. REST is a language and operating system independent architecture for designing network applications using simple HTTP to connect between machines.

Q.2 Define IoT reference architecture.

Ans. IoT Reference Architecture: It converge the focus on abstracted mechanisms rather than a consolidated applications, according to the concern of IoT stakeholders. All the abstracted perspectives are used according to the general literature and standards. The reference model can be used for deciding the working boundary for the researchers and analyzers. So that can limit the work and to give proper shape in their proposed schemes and architectures. In other words, by reference model organization of IoT standard can easily be done.

Q.3 What are the requirements of reference architecture?

Ans. Requirements of Reference Architecture : The requirements of reference architecture are as follows :

- (i) Connectivity and communications
- (ii) Device management

- (iii) Data collection, analysis and actuation
- (iv) Scalability
- (v) Security
- (vi) HA
- (vii) Predictive analysis

Q.4 Explain why reference architecture is good for IoT.

Ans. IoT devices are very commonly used for collecting and analyzing personal data. A model for managing the identity and access control for IoT devices and the data they publish and consume is a key requirement.

Q.5 What is the aim of IoT reference model?

Ans. Aims of IoT Reference Model : The IoT reference model aims at establishing a common grounding and common language for IoT architecture and systems.

Q.6 Define IoT reference model.

Ans. IoT Reference Model : The IoT reference model provides the highest abstraction level for the definition of the IoT Architectural Reference Model. It promotes a common understanding of the IoT domain. The description of the IoT reference model includes a general discourse on the IoT domain (i) an IoT domain model as a top-level description (ii) an IoT information model explaining how IoT knowledge is going to be modelled (iii) an IoT communication model in order to understand specifics about communication between many heterogeneous IoT devices and the internet as a whole.

of the IoT Domain Model is modelled, which is explicitly gathered, stored and processed in an IoT system, e.g. information about Devices, IoT Services and Virtual Entities. The IoT Functional Model identifies groups of functionalities, of which most are grounded in key concepts of the IoT Domain Model. A number of these Functionality Groups (FG) build on each other, following the relations identified in the IoT Domain Model. The Functionality Groups provide the functionalities for interacting with the instances of these concepts or managing the information related to the concepts, e.g. information about Virtual Entities or descriptions of IoT Services. The functionalities of the FGs that manage information use the IoT Information Model as the basis for structuring their information.

A key functionality in any distributed computer system is the communication between the different components. One of the characteristics of IoT systems is often the heterogeneity of communication technologies employed, which often is a direct reflection of the complex needs such systems have to meet. The IoT Communication Model introduces concepts for handling the complexity of communication in heterogeneous IoT environments. Communication also constitutes one FG in the IoT Functional Model.

Finally, Trust, Security and Privacy (TSP) are important in typical IoT use – case scenarios. Therefore, the relevant functionalities and their interdependencies and interactions are introduced in the IoT TSP Model. As in the case of communication, security constitutes one FG in the Functional Model.

Q.9 Write short note on REST.

Ans. REST: Representational state transfer (REST) or RESTful Web services are one way of providing interoperability between computer systems on the Internet. REST-compliant Web services allow requesting systems to access and manipulate textual representations of Web resources using a uniform and predefined set of stateless operations. Other forms of Web service exist, which expose their own arbitrary sets of operations such as WSDL and SOAP. REST is a set of principles that define how web standards, such as HTTP and URIs, are supposed to be used. To sum up, the principles can generally summarize into five keys: given every resource an ID, link resource together, use standard methods, resources with multiple representations, and communication stateless.

(a) **Give every "resource" an ID :** The resource can be everything connected through Internet. Everything should be identifiable by getting an ID - on the web, there is a unified concept for IDs: the URL. The URIs consist of a global namespace where every resource use uniform naming scheme. The benefit is that it can be applied both to browser in the web and to M2M communication.

(b) **Hypermedia as engine of application state:** The formal description is colloquially known as linking things together. The core is the concept of hypermedia, in other words: the idea of link. The benefit of link is that the links can link to resources from different application, different server, even different companies, because the naming scheme is a global standard, all of the resources in the web can be linked to each other.

(c) **Use standard methods:** The browser knows what to do with URIs because every resource supports the same interfaces, the same operations. HTTP calls verbs, the set of stand methods include GET, POST, PUT, DELETE, HEAD and OPTIONS. For clients to be able to interact with resources, they should implement the default application protocol (HTTP) correctly, such as GET, PUT, POST, DELETE methods. Using standard methods is important because it makes application become a part of the web and uniform interface also enables other component that support HTTP protocol to interact with the REST application.

Q.10 Explain cross platform compatibility and lack of skill set in development challenge.

Ans. Cross-Platform Compatibility (Hardware and Devices): IoT applications must be developed keeping in mind the technological changes of the future. Thus, IoT development requires a balance of hardware and software functions. Development teams may concentrate on designing the device that can achieve the finest performance but it might confine updating the product. Also, at the same time, the device allows using new functions, features, help in bug fixing as it has a heavy operating system but may cause a reduction in performance. Hence, it is a challenge for IoT application developers to ensure that device and IoT platform delivers the best performance despite heavy OS, device updates and bug fixings. Many vendors supply with SDKs and APIs for developers to add new functionalities to their already developed application. IoT applications run on the

PART-B

Q.7 Write short note on resource identifier.

Ans. Resource Identifiers : In order to ensure that an application is handling the correct resource, a mechanism to identify a resource univocally in the network is necessary. uniform resource identifiers (URIs), defined in RFC 3986, serve this specific need.

A URI identifies a resource univocally. A URI can be used to address a resource, so that it can be located, retrieved, and manipulated. There is a 1 : N relationship between a resource and URIs: a resource can be mapped to multiple URIs, but a URI points exactly to one resource. URIs can be of two kinds:

- A uniform resource name (URN) specifies the name of a resource (e.g., urn : ietf : rfc : 2616);
- A uniform resource locator (URL) specifies how to locate the resource, (e.g., http://example.com/books/123).

All URIs take the following form: scheme: scheme – specific – part. The scheme part defines how the rest of the URI is to be interpreted – it typically serves as an indication of the communication protocol that should be used to target the resource. For instance, URNs use the urn scheme, while web resources use the http scheme. URLs include all the information needed to successfully address the resource. A URL has the form shown in following figure.

| | | | | | |
|---------|-------------|-------|---------|-------|----------|
| http:// | example.com | :8080 | /people | ?id=1 | #address |
| scheme | host | port | path | query | fragment |

Fig. : Generic URL structure.

The optional [username:password] part specifies credentials to use for authenticated access to the resource. The host and optional port include networking information needed to reach to the resource. The host can be either an IP address or a fully qualified domain name, which must be resolved using the DNS system. The path provides information to locate the resource inside the host. The query contains matching information to filter out the result. Finally, the fragment can be used to identify a specific portion of the resource. URIs should be opaque and not expose any specific

notion of the format used to represent the targeted resource. For example, http://example.com/people/123 is a good URI, while http://example.com/people/123.xml and http://example.com/people/123.json are not.

Q.8 Explain interaction of all sub models in IoT reference model.

Ans. Interaction of all Sub-models: The IoT reference model aims at establishing a common grounding and a common language for IoT architectures and IoT systems. It consists of the sub – models shown in Fig. The “yellow” arrows show how concepts and aspects of one model are used as the basis for another.

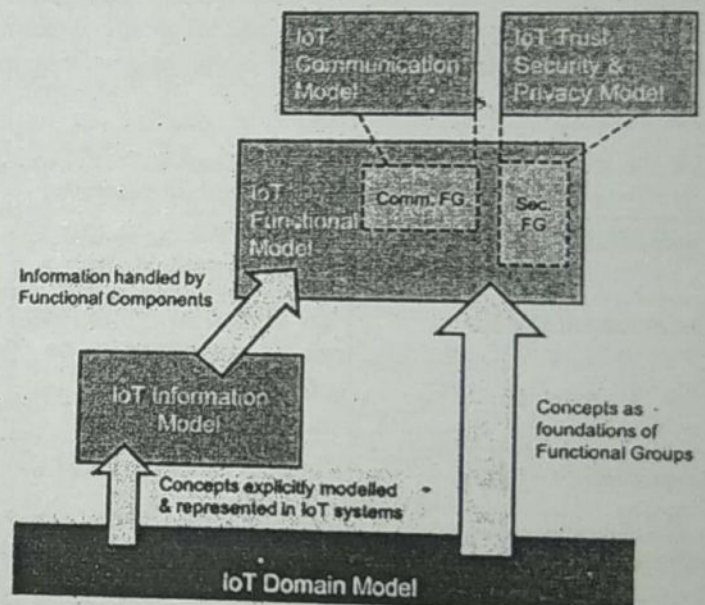


Fig. : Interaction of all sub-models in the IoT Reference Model. The sub – models are explained in the text body.

The foundation of the IoT reference model is the IoT Domain Model, which introduces the main concepts of the Internet of Things like Devices, IoT Services and Virtual Entities (VE), and it also introduces relations between these concepts. The abstraction level of the IoT Domain Model has been chosen in such a way that its concepts are independent of specific technologies and use-cases. The idea is that these concepts are not expected to change much over the next decades or longer.

Based on the IoT Domain Model, the IoT Information Model has been developed. It defines the structure (e.g. relations, attributes) of IoT related information in an IoT system on a conceptual level without discussing how it would be represented. The information pertaining to those concepts

web as well as mobile devices and hence need to be compatible.

As most of the IoT applications are integrated into the legacy systems, it is no less than a challenge for developers to make the system compliant to industry standards and protocols. While developing the IoT application, developers must ensure that the application can be seamlessly integrated without increasing difficulties in the IoT ecosystem (considering both hardware and software).

Lack of Skill Set: All of the development challenges above can only be handled if there is a proper skilled resource working on the IoT application development. IoT is a diverse field and thus count on resources that are well aware of software and hardware implementations. A right talent will always get you past the major challenges and will be an important IoT application development asset.

Q.11 Explain challenge, factors, resolution of the following with respect to design challenge in IoT.

- (i) Stability of network
- (ii) Power management
- (iii) Network failover and memory management

Ans. (i)

| Challenge | Factors | Resolution |
|---|---|---|
| <ul style="list-style-type: none"> Deploying IoT products in network unstable (ex: Poor GSM / GPRS signals) environment creates issues | <ul style="list-style-type: none"> Link speed Link stability SW state machine Asynchronous conditions | <ul style="list-style-type: none"> Boundary value tuning Algorithm enhancement Dynamic detection |

(ii)

| Challenge | Factors | Resolution |
|---|--|---|
| <ul style="list-style-type: none"> Overall power consumption by sensory nodes in IoT should be minimal, should consider fluctuations as well | <ul style="list-style-type: none"> Minimalistic power need Higher power backup Power fluctuations Power failure recovery | <ul style="list-style-type: none"> Component selection SW turning |

(iii)

| Challenge | Factors | Resolution |
|--|---|---|
| <ul style="list-style-type: none"> During network failover, internal non-volatile memory should store critical data | <ul style="list-style-type: none"> Network outage Fleet management Optimization Cost implications | <ul style="list-style-type: none"> Decision making in NVM Choosing right memory and data structure NOR instead of NAND Creating secondary partition for failover handling |

Q.12 Write short note on IoT security risks and challenges

Ans. Three categories of IoT risks include:

- (i) Risks that are typical in any Internet system
- (ii) Risks that are specific to IoT devices
- (iii) Safety to ensure no harm is caused by misusing actuators, for instance.

Traditional security practices such as locking down open ports on devices belong to the first category (for example, a fridge connected to the Internet in order to send alerts about the product inventory and temperature may use an unsecured SMTP server and can be compromised by a botnet). The second category includes issues specifically related to IoT hardware, e.g. the device may have its secure information compromised. For example, some IoT devices are too small to support proper asymmetric encryption. Furthermore, any device that can connect to the Internet has an embedded operating system deployed in its firmware and many of these embedded operating systems are not designed with security as their primary consideration.

In order to make IoT services available at low cost with a large number of devices communicating securely to each other, there are many security challenges to overcome. Some main challenges are as follows:

Scalability: Managing a large number of IoT nodes requires scalable security solutions.

Connectivity: In IoT communications, connecting various devices of different capabilities in a secure manner is another challenge.

End – to – End Security: End – to – end security measures between IoT devices and Internet hosts are equally important.

Authentication and Trust: Proper identification and authentication capabilities and their orchestration within a complex IoT environment are not yet mature. This prevents establishment of trust relationships between IoT components, which is a prerequisite for IoT applications requiring ad – hoc connectivity between IoT components, such as Smart City scenarios. Trust management for IoT is needed to ensure that data analytics engines are fed with valid data. Without authentication it is not possible to ensure that the data flow produced by an entity contains what it is supposed to contain.

Identity Management: Identity management is an issue as poor security practices are often implemented. For example, the use of clear text/Base64 encoded IDs/passwords with devices and machine-to-machine (M2M) is a common mistake. This should be replaced with managed tokens such

as JSON Web Tokens (JWT) used by OAuth/OAuth2 authentication and authorization framework (the Open Authorization).

Attack – Resistant Security Solutions: Diversity in IoT devices results in a need for attack – resistant and lightweight security solutions. As IoT devices have limited compute resources, they are vulnerable to resource enervation attacks.

Q.13 Explain various security threats and attacks in IoT devices in tabular form.

Ans. To emphasize security risks in IoT, its acronym has been presented as Interconnection of Threats (IoT). Indeed, IoT devices are particularly vulnerable to physical attacks, software attacks, side – channel attacks, and so on as presented in below table.

Table : Security Threats to IoT Devices

| Threats | Attack Procedure | Security Requirement | Examples |
|-----------------------|--|--------------------------|---|
| Physical attacks | Tamper with the hardware and other components. | Tamper resistance | Layout reconstruction, micro-probing |
| Environment attacks | The device encryption key can be discovered by the attacker by recovering the encryption information. | Secure encryption scheme | Timing attack, side – channel attack, fault analysis attack |
| Cryptanalysis attacks | Find ciphertext to break the encryption. | Secure encryption scheme | Known-plaintext attack, chosen plaintext attack |
| Software attacks | Exploit vulnerabilities in the system during its own communication interface and inject malicious codes. | Proper antivirus update | Trojan horse, worms, or viruses |

Current IoT platforms are built using technology solutions from a wide variety of vendors. Some of these platforms are an eclectic mix of components repurposed from existing solutions for use in specifically designed platforms with the hope that the components will work together in a secure way. Security measures within the IoT components, if any, have not been designed to take into account the dependencies resulting from the IoT connectivity capabilities. For example, industrial devices often do not have proper authentication mechanisms because they have been designed to be used in physically protected and isolated environments. Another example is the challenge of providing software updates or

security patches in a timely manner to end nodes without impairing functional safety.

Q.14 Explain attack taxonomy according to the IoT process phases.

Ans. In general, an IoT process can be considered as a five phase sequence, from data collection to data delivery to the end users. Table demonstrates the variety of attacks categorized for the five phases of IoT: data perception, storage, intelligent processing, data transmission and end – to – end delivery.

Table : Attack Taxonomy According to the IoT Process Phases

| Phase | Attack/Threat | Description |
|---|--|---|
| Data Perception: Various types of data collectors can be used. The device may be a static body (body sensors or RFID tags) or a dynamic vehicle (sensors and chips). | Data leakage or breach, data sovereignty, data loss, data authentication. | Data leakage can be internal or external, intentional or unintentional, involving hardware or software. |
| Storage: If the device has its own local memory, data can be stored. In the case of stateless devices, the data can be stored in the cloud. | Attack on availability, access control, integrity, denial of service, impersonation. | Availability is one of the primary security concerns. Distributed denial of service (DDoS) is an overload condition that is caused by a huge number of distributed attackers. |
| Intelligent Processing | Attack on authentication | An IoT solution provides data analysis and intelligent services in real time. |
| Data Transmission | Channel security, session hijack. Routing protocols, flooding | Threats in transmission, such as interrupting, blocking, data manipulation, forgery, etc. |
| End – to – End Delivery | Man or machine. Maker or hacker. | Delivery of processed data on time without errors or alteration. |

Q.15 Write short note on other challenges of IoT.

Ans. Other Challenges: There are many securities and privacy related issues in IoT design. These issues must be address during the development, and adopting IoT infrastructure and devices. But other than the privacy and security, there are some other challenges which must be taken in consideration. In the subsequent topic, some other challenges that are having equal importance.

Meeting Customer Expectations: Success of IoT implementation is directly proportional to the how clearly and effectively problems statements are defined. However most of the IoT service providers are lacking in it. Customer satisfaction, effectiveness of service and productivity are the alternatives which will influence the customer to adapt IoT services:

Analytics Challenges: The real value of the IoT solution is achieved using actionable insights derived from the IoT information collected. This requires a superior analytics platform that can based the ginormous amount of data that needs to be added to the solution.

Data analytics partners must realize that they must involve data processing, cleaning and representation while designing the IoT application. Therefore, leaving enough space for real-time extensibility to an IoT solution that will help solve the crucial challenge of implementing IoT.

Waiting for Governmental Regulation: While some companies are embracing IoT in actual time, others are

reluctant. In several instances, these companies are waiting with fresh norms and guidelines for public authorities to intervene. Some of the government regulations are shown in fig.

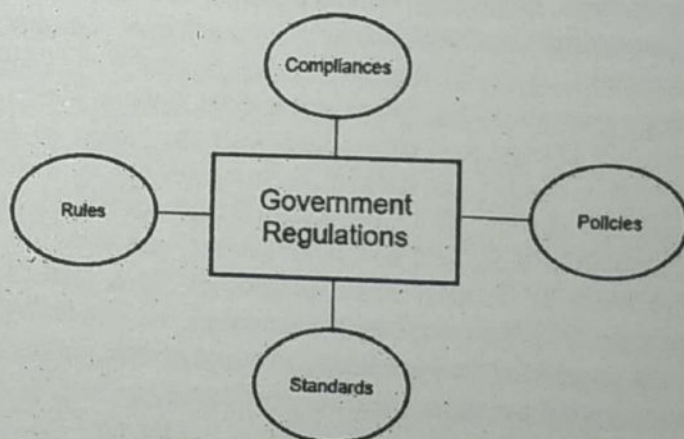


Fig. : Some of the government regulations

In the emerging information system and an autonomous system like IoT, Trust is an important and critical issue. Although the generalize concept of trust is very rich still it can be summarized in the following five properties:

- **Objective property of trustee:** Reliability and security of trustee.
- **Subjective property of trustee:** Goodness, kindness, and honesty.
- **Objective property of trustor:** Criteria and policies for the trust decision.
- **Subjective property of trustor:** Willingness and trustworthiness.

- The context in which trust relation lies: Environment, location, time device is being used, and its operation mode, and many more.

IoT trust management systems should assist detect malicious nodes by supporting other security mechanisms and protocols like system for intrusion detection, protocols and mechanism for authentication check, mechanisms to check and preserve privacy, and other important management systems.

Q.16 Write short note on trust issues and trust related attacks.

Ans. Trust Issues and Trust-related Attacks: Indeed, most trust management schemes assume collaboration between distributed entities. Cooperation will merely be broken by inconsiderate behaviors and exploited by malicious aggressors to trigger later trust-related attacks. Following are the various types of trust-related attacks:

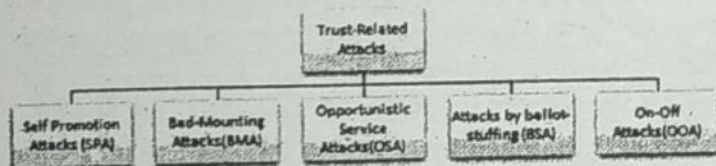


Fig. : Trust related attacks.

- **Self – promotion – attacks (SPA):** In such types of attacks, malicious nodes manipulate their name by offering sensitive recommendations.
- **Bad – mouthing attacks (BMA):** In these attacks, malicious node manipulates the name of another trusted node creating unhealthy suggestions.
- **Attacks by ballot – stuffing (BSA):** These kinds of attacks are also known as attacks by Good Mouthing. Some malicious nodes will operate these attacks to trigger the attack. In reality, by offering reasonable suggestions for it, a malicious node manipulates the name of another malicious node.
- **Opportunistic – service attacks (OSA):** In this type of attack, a malicious node tries to become timely by offering authentic services to keep its name high. The basic objective is to cheat various malicious nodes to maintain unhealthy-mouthing and sensitive-mouthing attacks.
- **On – off attacks (OOA):** In this type of attack, honest and unhealthy service is provided by the malicious node, or else. The goal is to remain sensitive to its name, and it will compromise the network by offering an honest

suggestion for malicious nodes or an unhealthy-recommendation for trusted node. Finding such attacks seems like a much difficult task.

Q.17 Explain the three layer and five layer architecture of internet of things.

Ans. There is no single consensus on architecture for IoT, which is agreed universally. Different architectures have been proposed by different researchers.

Three- and Five-Layer Architectures: The most basic architecture is a three – layer architecture as shown in fig. It was introduced in the early stages of research in this area. It has three layers, namely, the perception, network, and application layers.

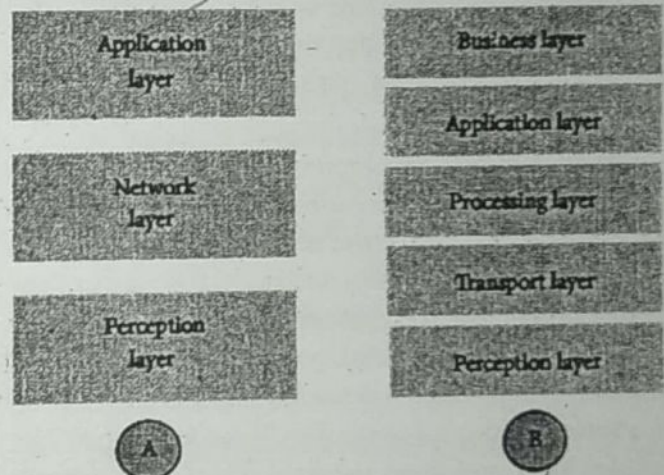


Fig. : Architecture of IoT (A: three layers) (B: five layers).

- The perception layer is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment.
- The network layer is responsible for connecting to other smart things, network devices and servers. Its features are also used for transmitting and processing sensor data.
- The application layer is responsible for delivering application specific services to the user. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities and smart health.

The three-layer architecture defines the main idea of the Internet of Things, but it is not sufficient for research on IoT because research often focuses on finer aspects of the Internet of Things. That is why, we have many more layered architectures proposed in the literature. One is the five –

layer architecture, which additionally includes the processing and business layers. The five layers are perception, transport, processing, application, and business layers (see Figure). The role of the perception and application layers is the same as the architecture with three layers. The function of the remaining three layers are as follows :

- (i) The transport layer transfers the sensor data from the perception layer to the processing layer and vice versa through networks such as wireless, 3G, LAN, Bluetooth, RFID, and NFC.
- (ii) The processing layer is also known as the middleware layer. It stores, analyzes and processes huge amounts of data that comes from the transport layer. It can manage and provide a diverse set of services to the lower layers. It employs many technologies such as databases, cloud computing and big data processing modules.
- (iii) The business layer manages the whole IoT system, including applications, business and profit models and users privacy.

Another architecture proposed by Ning and Wang is inspired by the layers of processing in the human brain. It is inspired by the intelligence and ability of human beings to think, feel, remember, make decisions and react to the physical environment. It is constituted of three parts. First is the human brain, which is analogous to the processing and data management unit or the data center. Second is the spinal cord, which is analogous to the distributed network of data processing nodes and smart gateways. Third is the network of nerves, which corresponds to the networking components and sensors.

Q.18 What is cloud and fog based IoT architecture? Explain.

Ans. In particular, we have been slightly vague about the nature of data generated by IoT devices and the nature of data processing. In some system architectures the data processing is done in a large centralized fashion by cloud computers. Such a cloud centric architecture keeps the cloud at the center, applications above it, and the network of smart things below it. Cloud computing is given primacy because it provides great flexibility and scalability. It offers services such as the core infrastructure, platform, software, and storage. Developers can provide their storage tools, software tools, data mining, and machine learning tools, and visualization tools through the cloud.

Lately, there is a move towards another system architecture, namely, fog computing, where the sensors and network gateways do a part of the data processing and analytics. A fog architecture presents a layered approach as shown in fig., which inserts monitoring, preprocessing, storage, and security layers between the physical and transport layers. The monitoring layer monitors power, resources, responses, and services. The preprocessing layer performs filtering, processing, and analytics of sensor data. The temporary storage layer provides storage functionalities such as data replication, distribution, and storage. Finally, the security layer performs encryption/decryption and ensures data integrity and privacy. Monitoring and preprocessing are done on the edge of the network before sending data to the cloud.

Often the terms "fog computing" and "edge computing" are used interchangeably. The latter term predates the former and is construed to be more generic. Fog computing originally termed by Cisco refers to smart gateways and smart sensors, whereas edge computing is slightly more penetrative in nature. This paradigm envisions adding smart data preprocessing capabilities to physical devices such as motors, pumps, or lights. The aim is to do as much of preprocessing of data as possible in these devices, which are termed to be at the edge of the network. In terms of the system architecture, the architectural diagram is not appreciably different from fig.

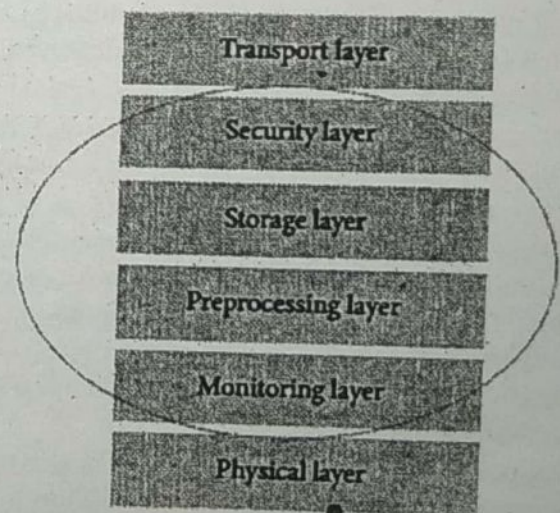


Fig. : Fog architecture of a smart IoT gateway

Finally, the distinction between protocol architectures and system architectures is not very crisp. Often the protocols and the system are codesigned. We shall use the generic 5-layer IoT protocol stack for both the fog and cloud architectures.

Q.19 Explain domain model in detail.

Ans. Domain Model : The IoT-A project defines a domain model as a description of concepts belonging to a particular area of interest. The domain model also defines basic attributes of these concepts, such as name and identifier. Furthermore, the domain model defines relationships between concepts, for instance "Services expose Resources". Domain models also help to facilitate the exchange of data between domains model also provides a common lexicon and taxonomy of the IoT domain. The main purpose of a domain model is to generate a common understanding of the target domain. Such a common understanding is important, not just project-internally, but also for the scientific discourse. Only with a common understanding of the main concepts it becomes possible to argue about architectural solutions and to evaluate them. As has been pointed out in literature, the IoT domain suffers already from an inconsistent usage and understanding of the meaning of many central terms.

The domain model is an important part of any reference model since it includes a definition of the main abstract concepts (abstractions), their responsibilities, and their relationships. Regarding the level of detail, the domain model should separate out what does not vary much from what does. For example, in the IoT domain, the device concept will likely remain relevant in the future, even if the types of devices used will change over time and/or vary depending on the application context. For instance, there are many technologies to identify objects: RFID, bar codes, image recognition etc. But which of these will still be in use 20 years from now? And which is the best-suited technology for a particular application? Since no one has the answers to such and related questions, the IoT Domain Model does not include particular technologies, but rather abstractions thereof.

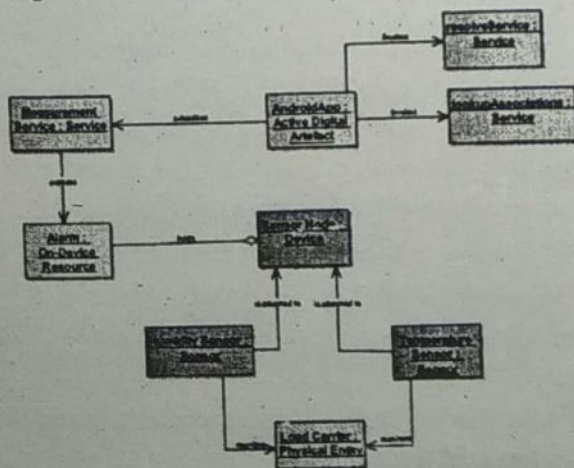


Fig. : Instantiated IoT domain Model for the "Red Thread" example

As we can see in fig., the important entities that are relevant for our use case are depicted with blocks of different colours. For instance, there is our truck driver "Ted" represented by as a yellow box (viz. instance), and the temperature sensor (that triggers an alarm after "Ted" had turned off the engine of the truck) is represented as a blue instance. Already at this stage we can easily deduct that there is some colour coding involved that reflects an aspect of the respective entity.

In addition to the coloured boxes, the diagram also shows arrows with verbs that connect the boxes. If we look very closely to the arrows, we see that they have different terminators such as diamond shapes or traditional arrow shapes. These shapes illustrate different kinds of relationships between the objects that are connected by them. In a similar way as the category names and the colour coding of the objects are related to each other, the verbs indicate information about the relationships shown with the arrows.

PART-C

Q.20 What are the requirements of reference architecture? Explain in detail.

Ans. There are some specific requirements for IoT that are unique to IoT devices and the environments that support them, e.g. many requirements emerge from the limited formfactors and power available to IoT devices. Other requirements come from the way in which IoT devices are manufactured and used. The approaches are much more like traditional consumer product designs than existing Internet approaches. Of course there are a number of existing best practices for the server-side and Internet connectivity that need to be remembered and factored in.

1. Connectivity and Communications: Existing protocols, such as HTTP, have a very important place for many devices. Even an 8-bit controller can create simple GET and POST requests and HTTP provides an important unified (and uniform) connectivity. However, the overhead of HTTP and some other traditional Internet protocols can be an issue for two main reasons. Firstly, the memory size of the program can be an issue on small devices. However, the bigger issue

is the power requirements. In order to meet these requirements, we need a simple, small and binary protocol.

In addition, there are devices that connect directly and those that connect via gateways. The devices that connect via a gateway potentially require two protocols: one to connect to the gateway, and then another from the gateway to the cloud.

2. Device Management: While many IoT devices are not actively managed, this is not necessarily ideal. The following list covers some widely desirable requirements:

- The ability to disconnect a rogue or stolen device
- The ability to update the software on a device
- Updating security credentials
- Remotely enabling or disabling certain hardware capabilities
- Locating a lost device
- Wiping secure data from a stolen device
- Remotely re-configuring Wi-Fi, GPRS, or network parameters

3. Data Collection, Analysis, and Actuation: A few IoT devices have some form of UI, but in general IoT devices are focused on offering one or more sensors, one or more actuators, or a combination of both. The requirements of the system are that we can collect data from very large numbers of devices, store it, analyze it and then act upon it.

The reference architecture is designed to manage very large numbers of devices. If these devices are creating constant streams of data, then this creates a significant amount of data. The requirement is for a highly scalable storage system, which can handle diverse data and high volumes.

The action may happen in near real time, so there is a strong requirement for real-time analytics. In addition, the device needs to be able to analyze and act on data. In some cases this will be simple, embedded logic. On more powerful devices we can also utilize more powerful engines for event processing and action.

4. Scalability: Any server-side architecture would ideally be highly scalable, and be able to support millions of devices all constantly sending, receiving, and acting on data. However, many "high-scalability architectures" have come with an equally high price – both in hardware, software, and in complexity. An important requirement for this architecture is

to support scaling from a small deployment to a very large number of devices. Elastic scalability and the ability to deploy in a cloud infrastructure are essential. The ability to scale the serverside out on small cheap servers is an important requirement to make this an affordable architecture for small deployments as well as large ones.

5. Security: Security is one of the most important aspects for IoT. IoT devices are often collecting highly personal data, and by their nature are bringing the real world onto the Internet (and viceversa). This brings three categories of risks:

- Risks that are inherent in any Internet system, but that product/IoT designers may not be aware of.
- Specific risks that are unique to IoT devices.
- Safety to ensure no harm is caused by, for instance, misusing actuators.

The first category includes simple things such as locking down open ports on devices (like the Internet-attached fridge that had an unsecured SMTP server and was being used to send spam).

The second category includes issues specifically related to IoT hardware, e.g. the device may have its secure information read. For example, many IoT devices are too small to support proper asymmetric encryption. Another specific example is the ability for someone to attack the hardware to understand security. Another example – university security researchers who famously reverse-engineered and broke the Mifare Classic RFID card solution. These sort of reverse engineering attacks are an issue compared with pure web solutions where there is often no available code to attack (i.e. completely server-side implementation).

Two very important specific issues for IoT security are the concerns about identity and access management. Identity is an issue where there are often poor practices implemented. For example, the use of clear text/ Base64 encoded user IDs/passwords with devices and machine-to-machine (M2M) is a common mistake. Ideally these should be replaced with managed tokens such as those provided by OAuth/OAuth2.

Another common issue is to hard-code access management rules into either client- or server-side code. A much more flexible and powerful approach is to utilize models such as "Attribute Based Access Control" and "Policy Based

and the cloud. The most wellknown three potential protocols are :

- HTTP/HTTPS (and RESTful approaches on those)
- MQTT 3.1/3.1.1
- Constrained application protocol (CoAP)

Many small devices such as 8-bit controllers can only partially support the protocol - for example enough code to POST or GET a resource. The larger 32-bit based devices can utilize full HTTP client libraries that properly implement the whole protocol.

There are several protocols optimized for IoT use. The two best known are MQTT and CoAP. MQTT was invented in 1999 to solve issues in embedded systems and SCADA. MQTT is a publish-subscribe messaging system based on a broker model. The protocol has a very small overhead (as little as 2 bytes per message), and was designed to support lossy and intermittently connected networks. MQTT was designed to flow over TCP. In addition there is an associated specification designed for ZigBee-style networks called MQTT-SN (Sensor Networks).

CoAP is a protocol from the IETF that is designed to provide a RESTful application protocol modeled on HTTP semantics, but with a much smaller footprint and a binary rather than a text-based approach. CoAP is a more traditional client-server approach rather than a brokered approach. CoAP is designed to be used over UDP.

The reasons to select MQTT and not CoAP at this stage are :

- Better adoption and wider library support for MQTT;
- Simplified bridging into existing event collection and event processing systems, and
- Simpler connectivity over firewalls and NAT networks

However, both protocols have specific strengths (and weaknesses) and so there will be some situations where CoAP may be preferable and could be swapped in.

One important aspect with IoT devices is not just for the device to send data to the cloud/ server, but also the reverse. This is one of the benefits of the MQTT specification; because it is a brokered model, clients connect an outbound connection to the broker, whether or not the device is acting as a publisher or subscriber. This usually avoids firewall problems because this approach works even behind firewalls or via NAT.

In the case where the main communication is based on HTTP, the traditional approach for sending data to the device would be to use HTTP Polling. This is very inefficient

and costly, both in terms of network traffic as well as power requirements. The modern replacement for this is the WebSocket protocol that allows an HTTP connection to be upgraded into a full two-way connection. This then acts as a socket channel (similar to a pure TCP channel) between the server and client. Once that has been established, it is up to the system to choose an ongoing protocol to tunnel over the connection.

For the reference architecture using MQTT as a protocol with WebSockets. In some cases, MQTT over WebSockets will be the only protocol. This is because it is even more firewall-friendly than the base MQTT specification as well as supporting pure browser/JavaScript clients using the same protocol.

Note that while there is some support for WebSockets on small controllers, such as Arduino, the combination of network code, HTTP and WebSockets would utilize most of the available code space on a typical Arduino 8-bit device. Therefore, we use of WebSockets on the larger 32-bit devices.

3. The Aggregation/Bus Layer: An important layer of the architecture is the layer that aggregates and brokers communications. This is an important layer for three reasons:

1. The ability to support an HTTP server and/or an MQTT broker to talk to the devices;
2. The ability to aggregate and combine communications from different devices and to route communications to a specific device (possibly via a gateway)
3. The ability to bridge and transform between different protocols, e.g. to offer HTTPbased APIs that are mediated into an MQTT message going to the device.

The aggregation/bus layer provides these capabilities as well as adapting into legacy protocols. The bus layer may also provide some simple correlation and mapping from different correlation models (e.g. mapping a device ID into an owner's ID or vice-versa).

Finally the aggregation/bus layer needs to perform two key security roles. It must be able to act as an OAuth2 Resource Server (validating Bearer Tokens and associated resource access scopes). It must also be able to act as a policy enforcement point (PEP) for policy-based access. In this model, the bus makes requests to the identity and access management layer to validate access requests. The identity and access management layer acts as a policy decision point (PDP) in this process. The bus layer then implements the results of these calls to the PDP to either allow or disallow resource access.

Access Control". The most well known of these approaches is that provided by the XACML standard. Such approaches remove access control decisions from hard-coded logic and externalize them into policies, which enabled the following:

- More powerful and appropriate decisions;
- Can potentially be based on contexts such as location, or which network is being used, or the time of day;
- Access control can be analyzed and audited; and
- Policies can be updated and changed, even dynamically, without recoding or modifying devices.

Our security requirements therefore should support :

- Encryption on devices that are powerful enough;
- A modern identity model based on tokens and not user ids/passwords;
- The management of keys and tokens as smoothly/remotely as possible; and
- Policy-based and user-managed access control for the system based on XACML.

Q.21 Explain the reference architecture for IoT with the help of block diagram.

Ans. The Architecture: The reference architecture consists of a set of components. Layers can be realized by means of specific technologies. There are also some cross-cutting/vertical layers such as access/identity management.

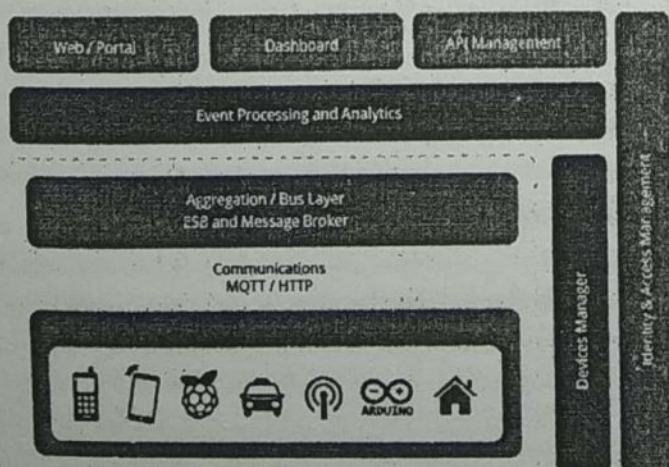


Fig. : Reference architecture for IoT

The layers are :

- Client/external communications - Web/Portal, Dashboard, APIs

- Event processing and analytics (including data storage)
- Aggregation/bus layer - ESB and message broker
- Relevant transports - MQTT/HTTP/XMPP/CoAP/AMQP, etc.

- Devices

The cross-cutting layers are :

- Device manager
- Identity and access management

1. The Device Layer: The bottom layer of the architecture is the device layer. Devices can be of various types, but in order to be considered as IoT devices, they must have some communications that either indirectly or directly attaches to the Internet. Examples of direct connections are :

- Arduino with Arduino Ethernet connection
- Arduino Yun with a Wi-Fi connection
- Raspberry Pi connected via Ethernet or Wi-Fi
- Intel Galileo connected via Ethernet or Wi-Fi
- Examples of indirectly connected device include
- ZigBee devices connected via a ZigBee gateway
- Bluetooth or Bluetooth Low Energy devices connecting via a mobile phone
- Devices communicating via low power radios to a Raspberry Pi

There are many more such examples of each type.

Each device typically needs an identity. The identity may be one of the following:

- A unique identifier (UUID) burnt into the device (typically part of the System-on-Chip, or provided by a secondary chip)
- A UUID provided by the radio subsystem (e.g. Bluetooth identifier, Wi-Fi MAC address)
- An OAuth2 Refresh/Bearer Token
- An identifier stored in nonvolatile memory such as EEPROM

For the reference architecture, every device has a UUID (preferably an unchangeable ID provided by the core hardware) as well as an OAuth2 Refresh and Bearer token stored in EEPROM.

The specification is based on HTTP; however, the reference architecture also supports these flows over MQTT.

2. The Communication Layer: The communication layer supports the connectivity of the devices. There are multiple potential protocols for communication between the devices

4. The Event Processing and Analytics Layer: This layer takes the events from the bus and provides the ability to process and act upon these events. A core capability here is the requirement to store the data into a database. This may happen in three forms. The traditional model here would be to write a serverside application, e.g. this could be a JAX-RS application backed by a database. However, there are many approaches where we can support more agile approaches. The first of these is to use a big data analytics platform. This is a cloud-scalable platform that supports technologies such as Apache Hadoop to provide highly scalable mapreduce analytics on the data coming from the devices. The second approach is to support complex event processing to initiate near real-time activities and actions based on data from the devices and from the rest of the system.

The following approaches are as follows :

- Highly scalable, column-based data storage for storing events
- Map-reduce for long-running batch-oriented processing of data
- Complex event processing for fast in-memory processing and near real-time reaction and autonomic actions based on the data and activity of devices and other systems
- In addition, this layer may support traditional application processing platforms, such as Java Beans, JAX-RS logic, message-driven beans, or alternatives, such as node.js, PHP, Ruby or Python.

5. Client/External Communication Layer: The reference architecture needs to provide a way for these devices to communicate outside of the device – oriented system. This includes three main approaches. Firstly, we need the ability to create web-based front – ends and portals that interact with devices and with the event-processing layer. Secondly, we need the ability to create dashboards that offer views into analytics and event processing. Finally, we need to be able to interact with systems outside this network using machine-to-machine communications (APIs). These APIs need to be managed and controlled and this happens in an API management system.

The recommended approach to building the web front end is to utilize a modular front-end architecture, such as a portal, which allows simple fast composition of useful UIs. Of course the architecture also supports existing Web server-side technology, such as Java Servlets/ JSP, PHP, Python, Ruby, etc.

The API management layer provides three main functions:

- The first is that it provides a developer-focused portal (as opposed to the user-focused portal previously mentioned), where developers can find, explore, and subscribe to APIs from the system. There is also support for publishers to create, version, and manage the available and published APIs;
- The second is a gateway that manages access to the APIs, performing access control checks (for external requests) as well as throttling usage based on policies. It also performs routing and load-balancing;
- The final aspect is that the gateway publishes data into the analytics layer where it is stored as well as processed to provide insights into how the APIs are used.

6. Device Management: Device management (DM) is handled by two components. A server – side system (the device manager) communicates with devices via various protocols and provides both individual and bulk control of devices. It also remotely manages software and applications deployed on the device. It can lock and/or wipe the device if necessary. The device manager works in conjunction with the device management agents. There are multiple different agents for different platforms and device types.

The device manager also needs to maintain the list of device identities and map these into owners. It must also work with the identity and access management layer to manage access controls over devices.

There are three levels of device: non-managed, semi-managed and fully managed (NM, SM, FM).

Fully managed devices are those that run a full DM agent. A full DM agent supports:

- Managing the software on the device
- Enabling/disabling features of the device (e.g. camera, hardware, etc.)
- Management of security controls and identifiers
- Monitoring the availability of the device
- Maintaining a record of the device's location if available
- Locking or wiping the device remotely if the device is compromised, etc.

Non-managed devices can communicate with the rest of the network, but have no agent involved. These may include 8-bit devices where the constraints are too small to support the agent. The device manager may still maintain information on the availability and location of the device if this is available.

Semi – managed devices are those that implement some parts of the DM (e.g. feature control, but not software management).

Q.22 What is social IoT? Explain its components.

Ans. The three main facets of an SIoT system are as follows :

- (i) The SIoT is navigable. We can start with one device and navigate through all the devices that are connected to it. It is easy to discover new devices and services using such a social network of IoT devices.
- (ii) A need of trustworthiness (strength of the relation – ship) is present between devices.
- (iii) We can use models similar to studying human social networks to also study the social networks of IoT devices.

Basic Components: In a typical social IoT setting, we treat the devices and services as bots where they can set up relationships between them and modify them over time. This will allow us to seamlessly let the devices cooperate among each other and achieve a complex task.

To make such a model work, we need to have many interoperating components. Following are the some of the major components in such a system.

- (1) **ID:** we need a unique method of object identification. An ID can be assigned to an object based on traditional parameters such as the MAC ID, IPv6 ID, a universal product code, or some other custom method.
- (2) **Meta-information:** Along with an ID, we need some meta-information about the device that describes its form and operation. This is required to establish appropriate relationships with the device and also appropriately place it in the universe of IoT devices.
- (3) **Security Controls:** This is similar to “friend list” settings on Facebook. An owner of a device might place restrictions on the kinds of devices that can connect to it. These are typically referred to as owner controls.
- (4) **Service Discovery:** Such kind of a system is like a service cloud, where we need to have dedicated directories that store details of devices providing certain kinds of services. It becomes very important to keep these directories up to date such that devices can learn about other devices.
- (5) **Relationship Management:** This module manages relationships with other devices. It also stores the types of devices that a given device should try to connect with based

on the type of services provided. For example, it makes sense for a light controller to make a relationship with a light sensor.

(6) Service Composition: This module takes the social IoT model to a new level. The ultimate goal of having such a system is to provide better integrated services to users. For example, if a person has a power sensor with her air conditioner and this device establishes a relationship with an analytics engine, then it is possible for the ensemble to yield a IoT of data about the usage patterns of the air conditioner. If the social model is more expansive, and there are many more devices, then it is possible to compare the data with the usage patterns of other users and come up with even more meaningful data. For example, users can be told that they are the largest energy consumers in their community or among their Facebook friends.

Q.23 Explain the design challenges in IoT.

Ans. Design Challenges : The design phase of IoT is a very vast field. This phase is having countless challenges. A diverse variety of IoT devices having different types of design challenges. The design phase of an embedded system also has many challenges. There are a variety of common challenges in designing embedded and IoT devices with some specification. Some of the design challenges are as follows.

Absence of Essential Flexibility for Running Applications over Embedded Systems : As the demand of connected devices is increasing day by day, embedded systems are proving efficient to work with totally different devices, due to its flexible nature; it is also able to adapt to different networking architectures so that new functionalities can be copied in the real-time environment. Fig. shows that flexible embedded systems can work with totally different devices:

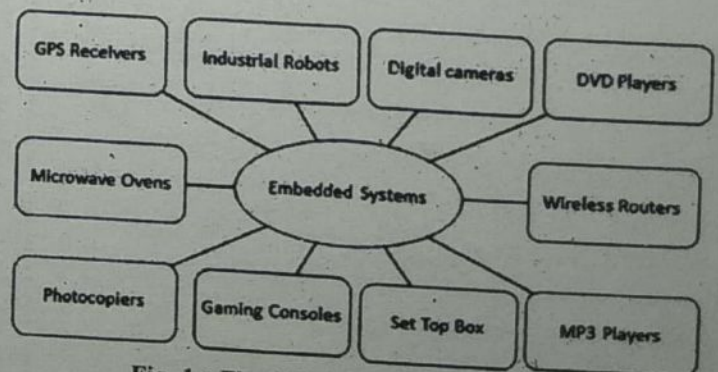


Fig. 1 : Flexibility of embedded systems

Developing flexible, embedded IoT systems is a challenge for designers. It is very difficult to design flexible IoT systems and devices that are having the capability to adapt the latest applications. Difficulties during developing flexible embedded IoT systems are:

- Problems in making certain smooth integration of the latest services.
- It's hard to adopt a new environment.
- Frequent hardware and package changes are difficult.
- There are issues in packaging tiny chip and the difficulty in integrating these tiny chips with low weight and a lesser amount of power consumption.
- Running different energy awareness operations becomes difficult.

Security Issues: IoT systems and devices must be secured robust and trustworthy. But Unstable performance in real-time embedded environment can be seen in entire IoT hardware products. Engineers face issues in ensuring the security of embedded parts as they operate in an extremely resources constrained and physically insecure environment. But nowadays due to the technological advancement, firm performance in real-time embedded environment can be seen in the latest IoT hardware products. The systems are designed to be robust and trustworthy. They are also made secure with cryptographic algorithms and various security procedures. Various types of security approaches are included to protect all parts of the embedded system from model to deployment.

High Power Dissipation : High power dissipation of hardware components like microprocessor and embedded chips is another challenge of the embedded and IoT system design. Hardware design uses more power to obtain the most effective performance from application and devices in a given period.

Deployment of an embedded system with a greater ratio of transistors and an adequate or sufficient amount of power consumption quantitative ratio is also one of the persistent challenges. High power dissipation coming up with low power embedded system is due to two reasons:

- There is an increase in the capacity distribution per semiconductor. Thus, to reduce the power consumption of overall embedded systems, engineers should use economic system design with technology alone.
- The main concentration of engineers consumes low power and delivers high performance, thus increasing

the frequency of the system, resulting in the burning of additional power. They should pay more attention to choices of styles.

Difficulties of Testing : To ensure that the product design is reliable, thorough testing, verification, and validation are other problems. Limitations related to the software updates make the testing process very challenging in finding bugs as would be prudent for a given software version. It also increases the importance of build and deployment procedure.

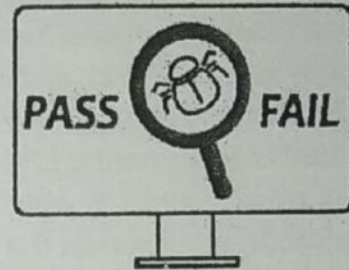


Fig. 2 : Testing of system for bugs

The tasks that can be performed during the hardware testing, verification, and validation are as follows:

- **Hardware Testing:** Wherever hardware tools are used by embedded developers, it is comparable to all types of testing. The system's performance, consistency and validations are tested as per merchandise demand, which refers to the embedded hardware.
- **Verification:** It is done to ensure that the verification has been carried out properly or not.
- **Validation:** In this, it is validated whether the merchandise matches the necessary quality standard and passes them.

Insufficient Practical Safety of Safety-critical Embedded Systems: Safety-critical embedded systems are very sensitive and special embedded system. Failure of such systems or applications based on such system can lead to serious injury, loss of life, significant property damage, or damage to the environment. Railways, automatic weapons, medical care devices, nuclear are the examples of fields where safety-critical system are being used.

It is highly considered that designing such systems is a very complex process because the designing process includes the combination of renowned design approaches and techniques in software and hardware to fulfill two different types of requirements, non-functional requirement, and functional requirement:

Functional Requirements: These are set of activities or operations are expected to perform by an embedded system.

Non-functional Requirements: These are the set of required attributes like small size, safety, low cost, ease of maintainability, high reliability and availability.

Q.24 Write detailed note on development challenges of IoT.

Ans. Development Challenges: The difficulty with IoT is that many companies focus exclusively on the growth of IoT while not assessing the first difficulties they face. Many of these companies have no background in IT trade or package growth, but most are committed to offering internet – connected equipment. Even enterprises that have package and hardware style expertise usually take IoT as alternative ancient computing technologies and build huge mistakes once developing IoT devices.

Connectivity: Connectivity is the original problem, that is, a way to connect computers to the internet as well as the platform for cloud computing. However, this can be determined to an excellent extent by the setting of the device application and also by the type of communication infrastructure that these devices provide. Connectivity of the various devices in an IoT ecosystem is shown in fig. 1.

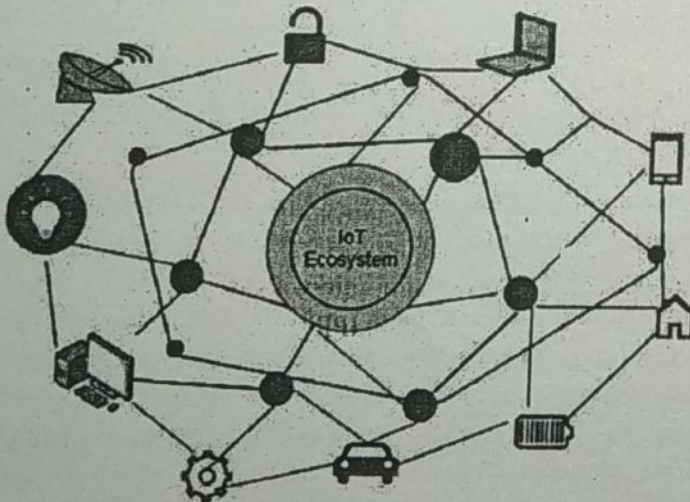


Fig. 1 : IoTecosystem connectivity

IoT utilizes a centralize client-server model in its current form to produce assorted servers, workstations and system resources. This can be quite cost-effective for current systems as the IoT remains in its infancy; however, what happens

once many billions of devices concurrently allude to the network?

According to Gartner's updated accounts, by 2020 more than 20 billion units will be connected to IoT. It is just a matter of your moment before consumers start to experience major IoT connectivity, effectiveness, and general performance bottlenecks.

For instance, if you want to create a smart home computer, such as an internet toaster, you will have access to a home Wi-Fi router or IoT router ZigBee/Z-Wave. Therefore, one or more transmitting media should be supported by your device. However, access to the Wi-Fi network is not available in some settings, such as IoT farming or smart cars, and the mobile network is also your first option.

Therefore, if you want to balance your alternative side and make style decisions, the prospects are endorsed, providing all options, and investment. For example, transmitting information to the cloud service through a cellular network would be expensive, but you will confirm picking up and running original mode or blockchain mode to create IoT system that is relatively less addicted to cloud computing.

You should also know, of course, that IoT remains an early technology undergoing major changes and modifications. There are many uncertainties and trends in this area. Consequently, techniques in use today may deteriorate in the longer term. On the contrary, IoT devices have an expanded life cycle compared to personal computers and intelligent phones that are simple to replace with fresh products. A smart refrigerator, for instance, should operate for at least 5 to 10 years. Therefore, you want to create a notion to verify that your device will retain its property and adapt to new techniques once IoT starts to take shape in the future.

Flexibility and Compatibility: Since the IoT pattern is constantly dynamic, therefore an IoT product must also support future technologies. However, it is compulsory to keep equilibrium between package and hardware once an IoT item is introduced.

Developing your device's dedicated hardware helps to achieve optimum performance, although product updates may also be prohibited. On the contrary, selecting relevant storage and computing resources and IoT – customized operating system may cause performance degradation, but it

allows you to expand your computer, use fresh features, and solve bug exploitation patches.

Some suppliers may attempt to give relevant APIs and SDKs whenever possible to allow development staff to add characteristics or functions to their IoT systems. An instance of honesty is Amazon Echo. This IoT instrument will execute the programming of the expansion in a thousand completely distinct directions.

Once the IoT product has been developed, you must ensure the compatibility and that your IoT device is embedded seamlessly with the IoT scheme of customers, without increasing the complexity or transportation of any setbacks to current knowledge. For this purpose, we want each package and hardware to be considered.

An optimal situation shows that customers do not have to be pressured into a brand – new implementation merely because they buy a smart device for their homes. The main examples are the Apple Home Kit and Samsung Smart-Things. Each allows personnel development to provide users with fresh IoT features in settings known to users.

Data Collection and Processing: In relation to safety and privacy, we would like to discover a joint technique for processing all the information gathered. To handle the scale of cloud storage and satisfy our platform requirements, we want to initially evaluate the amount of processed and picked up information. Data collection and processing is a continuous process, fig.2 illustrate the data collection and processing cycle.

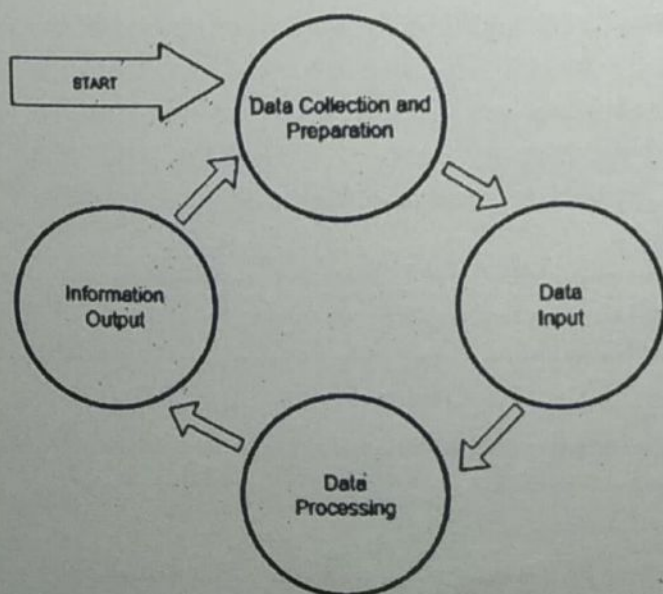


Fig. 2 : Data collection and processing cycle

Data collection and preparation is the first step in this contentious process. Prepared IoT information is as valuable as gold, but it is ineffective if it is stored on your server and not processed correctly. Therefore it is more essential that we are advancing in processing the information gathered.

Q.25 What are the various security challenges in IoT? Explain.

Ans. Various Security Challenges in IoT: The Internet of Things (IoT) will present new security challenges in cryptographic security, credentialing, and identity management. Currently available cryptographic techniques require further analysis to determine applicability in the Internet of Things. Credentialing presents significant challenges in the current Internet and these challenges will be exacerbated by the sheer number of devices and the expected limitations in user interfaces. Identity management is currently oriented towards either user or device identity; in the Internet of Things making an implicit or explicit mapping between IoT device identities and Internet user identities may be required. Network Security devices, such as firewalls and network guards, will be essential to meet security requirements. Security will be in tension with usability, privacy and devices constrained resources.

Current Internet security protocols rely on a well-known and widely trusted suite of cryptographic algorithms. The Advanced Encryption Standard (AES) block cipher for confidentiality; the Rivest – Shamir – Adelman (RSA) asymmetric algorithm for digital signatures and key transport; the Diffie – Hellman (DH) asymmetric key agreement algorithm; and the SHA-1 and SHA-256 secure hash algorithms. This suite of algorithms is supplemented by a set of emerging asymmetric algorithms, known as Elliptic Curve Cryptography (ECC). Adoption of the ECC algorithms has been slowed by significant IPR concerns, but publication of RFC 6090 and recent IPR disclosures may encourage adoption.

These cryptographic suites were designed with the expectation that significant resources (e.g., processor speed and memory) would be available. The applicability of these cryptographic techniques to the Internet of Things is unclear, and requires further analysis to ensure that algorithms can be successfully implemented given the constrained memory and processor speed expected in the IoT. For initial IoT

protocol development, developers are encouraged to look to AES-GCM, which is a combined mode supporting authentication and encryption, and the ECC asymmetric algorithms. As the resources available on common IoT devices becomes clearer, researchers may determine that these algorithm suites are not optimal, and research into more suitable cipher suites will advance. This will be a source of tension with implementers. Any techniques that are easy to implement in the IoT could be easy to break by users with more traditional computing resources. Regardless of the device footprint, we would suggest requiring at least 112 bit "security level" for all cryptographic techniques, which is the current baseline for less constrained devices. Assuming that an attacker will have the same limitations in resources is clearly an incorrect assumption.

To ensure that early adopters have security features available when needed, it is essential that IoT protocol suites specify a mandatory to implement but optional to use security solution. This will ensure security is available in all implementations, but configurable to use when not necessary (e.g., in closed environment). We expect initial deployments to proceed with security configured "off", but exploits that leverage such vulnerabilities will surely emerge in short order. The experience with home and small business WEP wireless deployments is informative: weak cryptography was rapidly discovered and exploited. Deploying the IoT without security will surely have the same result.

The most difficult aspect of cryptographic security is always key management. While many Internet protocols have been deployed with manual key management (i.e., "pre-shared keys") manual configuration of the number of devices in the IoT is unlikely to scale. In addition to the large number of devices, limited user interfaces will make it difficult to deploy meaningful security in this manner. Even if the devices can be manually keyed on initial deployment, automated re-key after deployment is essential.

Credentialing users and devices presents significant challenges in the current Internet, and these challenges will be exacerbated by the sheer number of devices and the expected limitations in user interfaces. Security techniques that combine automatic and manual techniques for initial deployment will likely be needed in the IoT. In particular, so-called "pairing protocols" such as those relied upon for Bluetooth security may need to be incorporated as a deployment

strategy. However, we envision that static keys will only support initial deployment, rather than be used as the traffic keys. Leap-of-Faith technologies, such as those employed in the "Better Than Nothing Security" (BTNS) IPsec profile, may also fill a key role in these protocols.

In the IoT, we expect that most devices will not be associated with a single person. A house only needs one toaster even if it serves a family of four. There may be a need to map device identities to groups of people (e.g., the adults in that family of four) in ways that are not commonly performed today.

Privacy issues are also expected to be significant. Our experiences with Smart Grid demonstrate the sensitivities of exposing electricity usage associated with a home or business. The IoT has the potential to expose the precise application of that energy demand, further violating the privacy expectations of the population. In combination with these privacy issues, compromises in the IoT protocol suites are likely to require establishing a security perimeter that monitors and restricts IoT devices. Older technologies from the military and intelligence communities, such as "network guards", once used to prevent information leakage may be needed once again.

In summary, the security challenges for the IoT are daunting. It is essential that early IoT protocols include mandatory to implement security features, even if those features stretch the capabilities of such devices. Automated key management is always a challenge, but it is even more critical that IoT protocols do not rely on pre-shared keys. Credentialing/registration of devices will also be a challenge; but pairing protocols are well-understood and provide one possible solution set. Privacy concerns may provide incentives for adoption for technologies designed to prevent information leakage in military/intelligence environments.

Q.26 Write detailed note on attack categorization according to the IoT architecture.

Ans. There are various IoT architecture models. In general, the IoT architecture is assumed to have four layers, presented in fig. We will briefly review the main security threats at the perception, network, and service layers. The most important security concerns in IoT presented as four-layer architecture (Fig.) are summarized in following table.

Table : Top Ten Vulnerabilities in IoT

| Security Concerns | Application & Interface Layer | Service Support Layer | Network Layer | Device Layer |
|---|-------------------------------|-----------------------|---------------|--------------|
| Insecure web interface | ✓ | ✓ | ✓ | |
| Insufficient authentication/authorization | ✓ | ✓ | ✓ | ✓ |
| Insecure network services | | ✓ | ✓ | |
| Lack of transport encryption | | ✓ | ✓ | |
| Privacy concerns | | ✓ | ✓ | ✓ |
| Insecure cloud interface | ✓ | | | |
| Insecure mobile Interface | ✓ | | ✓ | ✓ |
| Insecure security configuration | ✓ | ✓ | ✓ | |
| Insecure software/firmware | ✓ | | ✓ | |
| Poor physical security | | | ✓ | ✓ |

1. Security Threats at the Sensing/Perception Layer:

To fully implement IoT security, it must be designed and built into the devices themselves. This means that IoT devices must be able to prove their identity, maintain authenticity, sign and encrypt their data to maintain integrity, and limit locally stored data to protect privacy. The security model for devices must be strict enough to prevent unauthorized use but flexible enough to support secure ad hoc interactions with people and other devices on a temporary basis. For example, while unauthorized changing of the toll rate on a connected parking meter should be prevented, the meter should have a secure interface to reserve and pay for the parking spot for a limited duration.

(i) **Physical Damage:** Some attackers may lack technical knowledge and their attacks are limited by destroying devices. As device enclosures are often not tamperproof, the devices can be opened and their hardware can be accessed via probes and pin headers. Physical security requires designing tamper resistance into devices so that it is difficult to extract sensitive information such as personal data, cryptographic keys, or credentials. Many devices cannot protect their code and data from external access. As a result, an attacker can clone entire devices or manipulate their software and data. For example, to manipulate a glucometer so that it will provide incorrect readings. Another example is damage to hundreds of smart traffic light devices by thieves who stole the devices' SIM cards. The stolen cards were then used to make mobile phone calls in South Africa. The damage to the traffic light system resulted in many car crashes and a high cost to fix the entire system.

(ii) **Node Capture:** An active attacker can extract the information that the devices contain instead of destroying them.

(iii) **Sinkhole Attack:** If sensors are left unattended in the network for long periods, they become susceptible to sinkhole attack. In this attack, the compromised node extracts the information from all the surrounding nodes.

(iv) **Selective Forwarding Attack:** Malicious nodes may choose packets and drop them out, thereby selectively filtering certain packets and allowing the rest. Dropped packets may carry necessary sensitive data for further processing.

(v) **Witch Attack:** This attack occurs when a malicious IoT node takes advantage of failure of a legitimate node. When the legitimate node fails, the factual link takes a diversion through the malicious node for all its future communication, leading to data loss.

(vi) **HELLO Flood Attacks:** A malicious node initiates a HELLO flood attack by sending HELLO messages to all the neighbors that are reachable at its frequency level. Hence, it becomes a neighbor to all the nodes in the network. As the next step, this malicious node will broadcast a HELLO message to all its neighbors, affecting their availability. Flooding attacks cause non-availability of resources to legitimate users by distributing a huge number of nonsense requests to a certain service.

2. Security Threats at the Network and Service Support Layers: The service support layer (Fig.) represents the IoT management system and is responsible for onboarding devices and users, applying policies and rules, and orchestrating

automation across devices. Role-based access control to manage user and device identity and the actions they are authorized to take is critical at this layer. To achieve nonrepudiation, it is also important to maintain an audit trail of changes made by each user and device so that it is impossible to refute actions taken in the system. This monitoring data could also be used to identify potentially compromised devices when abnormal behavior is detected. Some typical attacks at the network and service support layer are as follows :

(i) **Man - in - the - Middle (MITM) Attack:** Man-in-the-middle attack is an example of the eavesdropping possible in the IoT. As device authentication involves exchange of device identities, identity theft is possible due to man-in-the-middle attack.

(ii) **Replay Attack:** During the exchange of identity-related information or other credentials in IoT, this information can be spoofed, altered or replayed. Replay attack is essentially a form of active man-in-the-middle attack.

(iii) **Denial of Service Attack:** As the IoT devices in IoT are resource constrained, they are vulnerable to resource usage attack. Attackers can send messages or requests to a specific device to consume its resources.

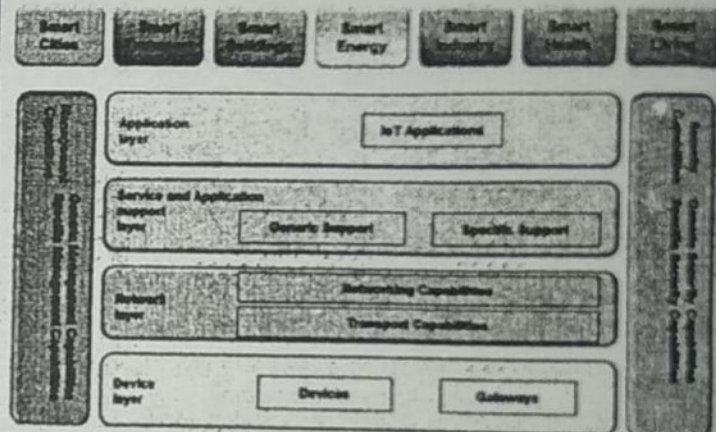


Fig. : IoT layered Architecture



IMPORTANT QUESTIONS

PART-A

Q.1 What are the important components that exist in the Internet of Things?

Ans. The important components that exist in the Internet of Things are as follows:

- (i) Hardware
- (ii) Software
- (iii) Verbal exchange infrastructure

Q.2 What is IoT testing?

Ans. IoT testing is a type of testing to check IoT devices. Today there is an increasing need to deliver better and faster services. There is a huge demand to access, create, use, and share data from any device. The thrust is to provide greater insight and control over various interconnected IoT devices. Hence, the IoT testing framework is important.

Q.3 What are the types of IoT?

Ans. There are two types of IoT:

- (i) **Internet of Things:** It creates a business that uses a gadgets to perform a task.
- (ii) **Industrial Internet of Things:** It creates business in the industry like agriculture.

Q.4 What is Thingful?

Ans. Thingful is a search engine for the Internet of Things. It allows secure interoperability between millions of IoT objects via the Internet. This IoT testing tool also to control how data is used and empowers to take more decisive and valuable decisions.

Q.5 What are the benefits of IoT?

Ans. The major benefits of IoT are:

- (i) **Improved customer engagement:** IoT improves customer experience by automating the action. E.g. Any issue in the car will be automatically detected by sensors. The driver, as well as the manufacturer, will be notified about it.
- (ii) **Technical optimization:** IOT has helped in improving technologies and improving them.

Q.6 What do you mean by M2M?

Ans. Machine-to-Machine (M2M) refers to networking of machines (or devices) for the purpose of remote monitoring and control and data exchange.

Q.7 What do you mean by Resource following in M2M system?

Ans. Resource Following: This procedure has been enormously changed the manner in which how we track an individual item ideal from the creation to convey. With the assistance of Internet of Things instruments and procedures, a great deal has been given to the web based business

associations. Utilizing these apparatuses and process, online business associations have use another methodology of advising their clients about their item whereabouts and the majority of this is going on effortlessly contrasted with the standard procedure that is actualized starting at now.

Q.8 Explain Protection telematics in M2M system.

Ans. Protection Telematics: Insurance organizations have possessed the capacity to tailor a variety of perspectives into thought while giving out any superior statements to their clients. In this way thinking about all the distinctive contributions to the application, the client will have the capacity to characterize a perfect measure of premium keeping all his past information into thought. Along these lines has upset the manner in which insurance agencies have been working up until this point.

Q.9 What do you mean by asset tracking?

Ans. Asset Tracking: This process has been tremendously changed the way how we track an individual object right from the production to deliver. With the help of Internet of Things tools and processes, a lot has been provided to the e-commerce organizations. Using these tools and process, e-commerce organizations have leverage a new approach of informing their customers about their product whereabouts and all of this is happening at a very low cost compared to the standard process that is implemented as of now.

Q.10 We have heard about IoT(Internet of Everything) and Machine-to-machine technology (M2M). Is it the same thing?

Ans. M2M is one of the driving components of the IoT ecosystem. In M2M, data is being acquired, logged, analysed and applied to specific vertical applications, narrowly designed for that solution, and the end-to-end solution has set interfaces and integrations.

IoT is a broader concept – “a catch-all technology bucket”. It makes uses of the data from various applications, including M2M applications, but with a more open architecture, and with use of other emerging technologies (Big Data / Analytics etc) use the insights gained from larger data sets, to improve, or create new business applications and models for industries.

Q.11 Where are the key areas a business can expect to gain value from M2M technology?

Ans. M2M is about sending business-critical information from sensors, meters or cameras in real time to your business applications.

Your application then translates the captured information into meaningful actions. This utilisation of key information (which you may not have access to before) can drive efficiency in your business. Outcomes can include increased sales, cost and service time reductions and improved customer experience.

Q.12 Define SDN.

Ans. Software-defined networking (SDN) is an architecture that aims to make networks agile and flexible. The goal of SDN is to improve network control by enabling enterprises and service providers to respond quickly to changing business requirements.

PART-B

Q.13 What are the top 5 Machine-to-Machine (M2M) applications in the world?

Ans.

- (i) Asset tracking and/or monitoring in some form or some other (stolen automobiles, fleet, construction system, wood pellets, tank level tracking, and many others.) seems to be the biggest.
- (ii) Insurance telematics is huge as it gives insurance groups the possibility to cut threat and force higher/extra appealing pricing.
- (iii) Utilities/automatic meter reading/clever grids – plenty of regulation and funding into this in the intervening time. There a plenty of countrywide solutions because the requirements and business case are driven in very numerous ways.
- (iv) Automotive is also very big and is driven by consumers demand.
- (v) mHealth is also there but not that big in size.

SDN technologies also help in distributed locations that have few IT personnel on site, such as an enterprise branch office or service provider central office.

"Naturally these places require remote and centralized delivery of connectivity, visibility and security. SDN solutions that centralize and abstract control and automate workflows across many places in the network, and their devices, improve operational reliability, speed and experience."

Q.18 How does SDN support intent-based networking?

Ans. Intent-based networking (IBN) has a variety of components, but basically is about giving network administrators the ability to define what they want the network to do, and having an automated network management platform create the desired state and enforce policies to ensure what the business wants happens.

If a key tenet of SDN is abstracted control over a fleet of infrastructure, then the provisioning paradigm and dynamic control to regulate infrastructure state is necessarily higher level. Policy is closer to declarative intent, moving away from the minutia of individual device details and imperative and reactive commands.

IDC says that intent-based networking "represents an evolution of SDN to achieve even greater degrees of operational simplicity, automated intelligence, and closed-loop functionality."

For that reason, IBN represents a notable milestone on the journey toward autonomous infrastructure that includes a self-driving network, which will function much like the self-driving car, producing desired outcomes based on what network operators and their organizations wish to accomplish.

While the self-driving car has been designed to deliver passengers safely to their destination with minimal human intervention, the self-driving network, as part of autonomous datacenter infrastructure, eventually will achieve similar outcomes in areas such as network provisioning, management, and troubleshooting – delivering applications and data, dynamically creating and altering network paths, and providing security enforcement with minimal need for operator intervention.

Q.19 How does SDN help customers with security?

Ans. SDN enables a variety of security benefits. A customer can split up a network connection between an end user and

the data center and have different security settings for the various types of network traffic: A network could have one public – facing, low security network that does not touch any sensitive information. Another segment could have much more fine-grained remote access control with software-based firewall and encryption policies on it, which allow sensitive data to traverse over it.

"For example, if a customer has an IoT group it doesn't feel is all that mature with regards to security, via the SDN controller we can segment that group off away from the critical high-value corporate traffic. "SDN users can roll out security policies across the network from the data center to the edge and if we do all of this on top of white boxes, deployments can be 30 – 60 percent cheaper than traditional gear."

The ability to look at a set of workloads and see if they match a given security policy is a key benefit of SDN, especially as data is distributed.

The ability to deploy a whitelist security model like we do with ACI (Application Centric Infrastructure) that lets only specific entities access explicit resources across your network fabric is another key security element SDN enables.

In fact, micro-segmentation has developed as a notable use case for SDN. As SDN platforms are extended to support multicloud environments, they will be used to mitigate the inherent complexity of establishing and maintaining consistent network and security policies across hybrid IT landscapes.

Q.20 What is SDN's role in cloud computing?

Ans. SDN's role in the move toward private cloud and hybrid cloud adoption seems a natural. In fact, big SDN players such as Cisco, Juniper and VMware have all made moves to tie together enterprise data center and cloud worlds. Cisco's ACI Anywhere package would, for example, let policies configured through Cisco's SDN APIC (Application Policy Infrastructure Controller) use native APIs offered by a public-cloud provider to orchestrate changes within both the private and public cloud environments.

"As organizations look to scale their hybrid cloud environments, it will be critical to leverage solutions that help improve productivity and processes". "The ability to leverage the same solution, like Cisco's ACI, in your own private-cloud environment as well as across multiple public clouds will

Q.14 Write short note on M2M.

Ans. M2M Machine – to – Machine (M2M) refers to networking of machines (or devices) for the purpose of remote monitoring and control and data exchange. Figure 1 shows the end – to – end architecture for M2M systems comprising of M2M area networks, communication network and application domain. An M2M area network comprises of machines (or M2M nodes) which have embedded hardware modules for sensing, actuation and communication. Various communication protocols can be used for M2M local area networks such as ZigBee, Bluetooth, ModBus, M – Bus,

Wireless M-Bus, Power Line Communication (PLC), 6LoWPAN, IEEE 802.15.4, etc. These communication protocols provide connectivity between M2M nodes within an M2M area network. The communication network provides connectivity to remote M2M area networks. The communication network can use either wired or wireless networks (IP-based). While the M2M area networks use either proprietary or non-IP based communication protocols, the communication network uses IP – based networks. Since non – IP based protocols are used within M2M area networks, the M2M nodes within one network cannot communicate with nodes in an external network. To enable the communication between remote M2M area networks, M2M gateways are used.

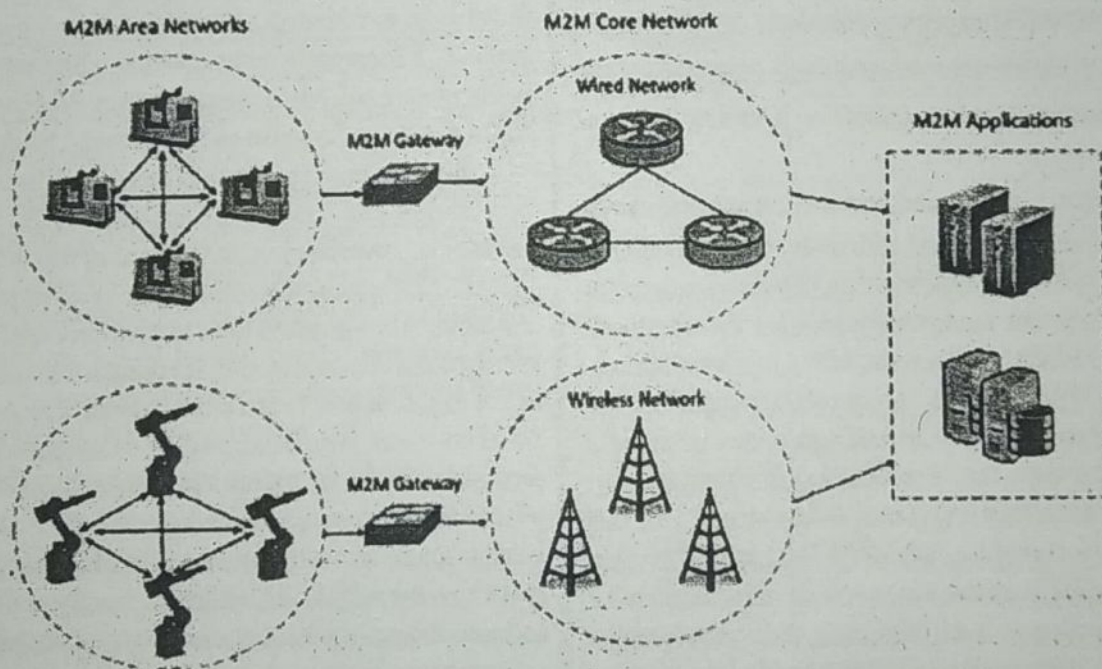


Fig. 1 : M2M system architecture

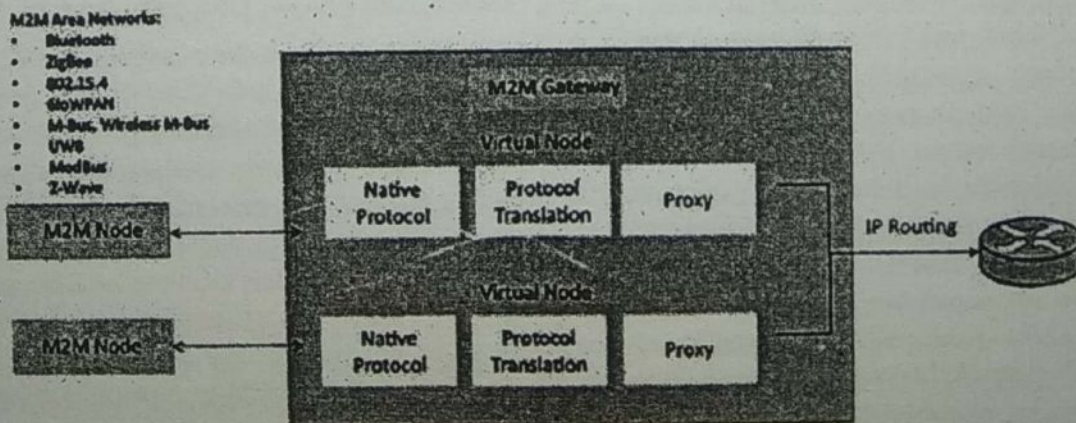


Fig. 2 : Block diagram of an M2M gateway

Fig.1 shows a block diagram of an M2M gateway. The communication between the M2M nodes and the M2M gateway is based on the communication protocols which are native to the M2M area network. M2M gateway performs protocol translations to enable IP-connectivity for M2M area networks. M2M gateway acts as a proxy performing translations from/to native protocols to/from Internet Protocol (IP). With an M2M gateway, each node in an M2M area network appears as a virtualized node for external M2M area networks.

The M2M data is gathered into point solutions such as enterprise applications, service management applications, or remote monitoring applications. M2M has various application domains such as smart metering, home automation, industrial automation, smart grids, etc. M2M solution designs (such as data collection and storage architectures and applications) are specific to the M2M application domain.

Q.15 Write short note on similarities between IoT and M2M.

Ans. M2M communications refers to direct wired or wireless communication between devices using any communications channel that does not necessarily require direct human intervention (ETSI, 2010). As such, M2M can be viewed as the forerunner of IoT. M2M communication can include industrial production facilities, enabling a sensor or meter to communicate the data that it records (e.g., temperature, throughput, and inventory level) to application software that can further process them (e.g., adjusting an industrial process based on technical parameters, such as temperature or triggering new processes, such as placing orders to replenish inventory). Such communication was aimed at monitoring remote machines from which data were received, processed at some central station, and eventually relayed back to those machines with adjusted parameters, if necessary. A core motivation for many organizations is to reduce service management costs through remote diagnostics, remote troubleshooting, remote updates, and other remote capabilities that reduce the need to deploy field service personnel. IoT accommodates the same devices/assets/machines as M2M applications, but also very small (low-power), personal, and inexpensive devices with sometimes very limited functionality that might not be able to justify a dedicated M2M hardware module. Although IoT and M2M communications have remote access to machines, or in more general terms "devices" in common, there are no other major similarities. For example,

traditional M2M solutions typically rely on point-to-point communications using embedded hardware modules and dedicated protocols. In contrast, IoT solutions depend predominantly on IP-based networks to interface device data to a cloud or middleware platform primarily using common/open protocols (in order to ensure maximum interoperability, in the sense of a remote device connected to some central hub, as well as particular interoperability among the devices themselves). Another difference is that M2M solutions offer remote access to machine data that are traditionally targeted at point solutions in service management applications. In the past, these data are rarely, if ever, integrated with enterprise applications to help improve overall business performance. Finally, IoT-based data delivery increasingly involves cloud services enabling access by any sanctioned enterprise application, whereas M2M typically employs direct point-to-point communication. The cloud-based architecture also makes IoT inherently more scalable, eliminating the need for incremental hard-wired connections or SIM card installations. M2M is often referred to as "plumbing," while IoT is viewed as a universal enabler. It could be argued that the conceptual boundaries and visions of IoT and M2M have become increasingly overlapping. Indications of this include that more recent M2M communications have evolved into a system of networks that transmits data to personal appliances. In this sense, M2M communication is taking increasingly advantage of the expansion of IP networks globally by switching from point-to-point proprietary style connections to IP-based multipoint communications. We may conclude that the focus of M2M issues tends to be more on the technical infrastructure layer. In contrast, the emerging IoT possesses much greater scope. IoT calls for the integration of device and sensor data with business intelligence, analytics, and other enterprise applications in order to achieve numerous benefits throughout manufacturing enterprises with a strong emphasis on improving products, processes, and business models.

In any case, it can be concluded that the IoT and M2M technologies are similar in that they both provide solutions for collection, storage and exchange of data between devices under minimal human supervision. What also unites them is the fact that they are increasingly closer to each other in matters of device management. With new solutions coming along, such as AVSystem's Cointel Iot Device Management platform, which are able to provide an umbrella over the IoT and M2M device management strategies, the boundary between the IoT and M2M is growing ever less marked.

While it is true that the IoT has been largely based on the foundations provided by M2M solutions, it must be added

that it has been improving on them ever since it was established as one of the main sources of innovation in the lives of individuals, businesses and whole societies.

Q.16 What do you understand by software defined networking? What are the limitations of conventional architecture?

Ans. Software Defined Networking: Software-Defined Networking (SDN) is a networking architecture that separates the control plane from the data plane and centralizes the network controller. Figure shows the conventional network architecture built with specialized hardware (switches, routers, etc.). Network devices in conventional network architectures are getting exceedingly complex with the increasing number of distributed protocols being implemented and the use of proprietary hardware and interfaces. In the conventional network architecture the control plane and data plane are coupled. Control plane is the part of the network that carries the signaling and routing message traffic while the data plane is the part of the network that carries the payload data traffic.

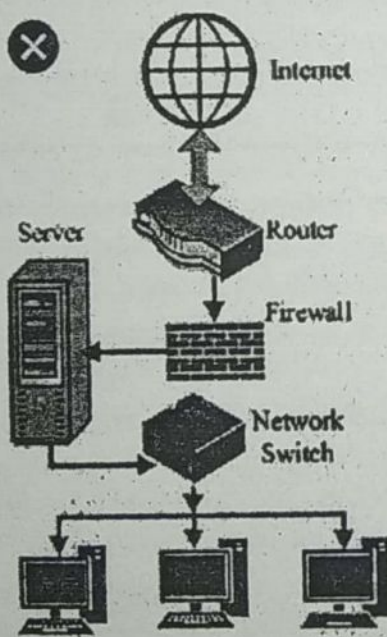


Fig.

The limitations of the conventional network architectures are as follows:

- (i) **Complex Network Devices:** Conventional networks are getting increasingly complex with more and more protocols being implemented to improve link speeds and reliability. Interoperability is limited due to the lack of standard and open interfaces. Network devices use proprietary hardware and software and have slow product life-cycles limiting innovation. The

conventional networks were well suited for static traffic patterns and had a large number of protocols designed for specific applications. For IoT applications which are deployed in cloud computing environments, the traffic patterns are more dynamic. Due to the complexity of conventional network devices, making changes in the networks to meet the dynamic traffic patterns has become increasingly difficult.

- (ii) **Management Overhead:** Conventional networks involve significant management overhead. Network managers find it increasingly difficult to manage multiple network devices and interfaces from multiple vendors. Upgradation of network requires configuration changes in multiple devices (switches, routers, firewalls, etc.)
- (iii) **Limited Scalability:** The virtualization technologies used in cloud computing environments has increased the number of virtual hosts requiring network access. IoT applications hosted in the cloud are distributed across multiple virtual machines that require exchange of traffic. The analytics components of IoT applications run distributed algorithms on a large number of virtual machines that require huge amounts of data exchange between virtual machines. Such computing environments require highly scalable and easy to manage network architectures with minimal manual configurations, which is becoming increasingly difficult with conventional networks.

Q.17 How does SDN support edge computing, IoT and remote access?

Ans. A variety of networking trends have played into the central idea of SDN. Distributing computing power to remote sites, moving data center functions to the edge, adopting cloud computing, and supporting Internet of Things environments – each of these efforts can be made easier and more cost efficient via a properly configured SDN environment.

Typically in an SDN environment, customers can see all of their devices and TCP flows, which means they can slice up the network from the data or management plane to support a variety of applications and configurations. So users can more easily segment an IoT application from the production world if they want, for example.

Some SDN controllers have the smarts to see that the network is getting congested and, in response, pump up bandwidth or processing to make sure remote and edge components don't suffer latency.

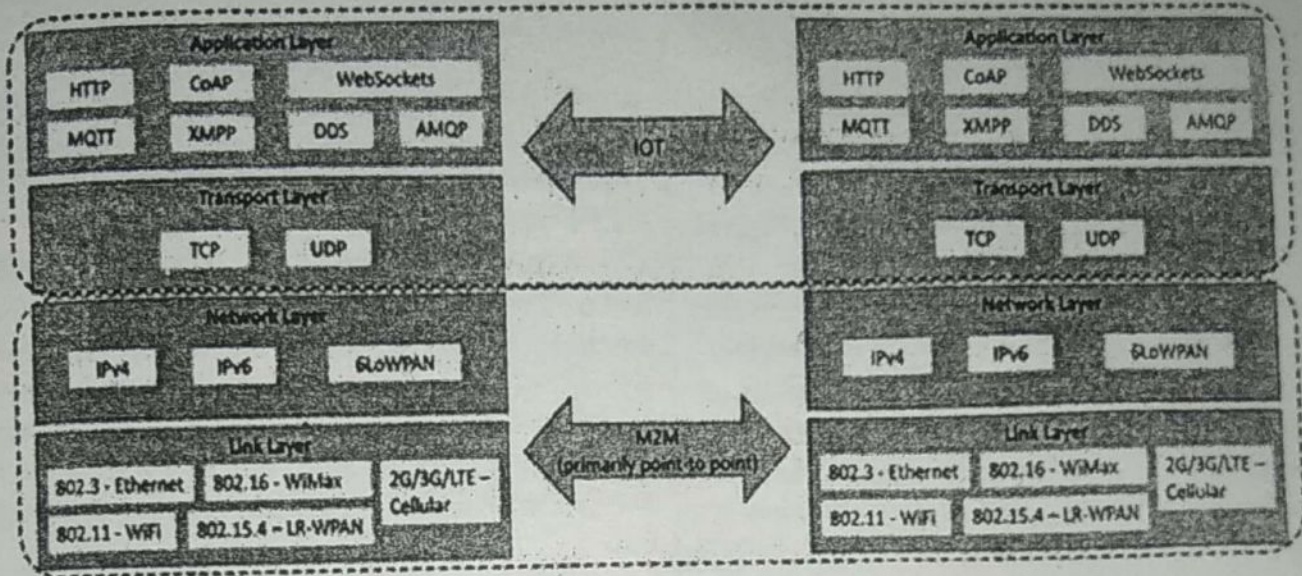


Fig. : Communication in IoT - IP-based. Whereas M2M - non-IP based networks.

PART-C

Q.26 Explain IoT in contrast to M2M in detail.

Ans. IoT: The IoT is a widely used term for a set of technologies, systems, and design principles associated with the emerging wave of Internet-connected things that are based on the physical environment. In many respects, it can initially look the same as M2M communication - connecting sensors and other devices to Information and Communication Technology (ICT) systems via wired or wireless networks.

In contrast to M2M, however, IoT also refers to the connection of such systems and sensors to the broader Internet, as well as the use of general Internet technologies. In the longer term, it is envisaged that an IoT ecosystem will emerge not dissimilar to today's Internet, allowing things and real world objects to connect, communicate, and interact with one another in the same way humans do via the web today. Increased understanding of the complexity of the systems in question, economies of scale, and methods for ensuring interoperability, in conjunction with key business drivers and governance structures across value chains, will create wide-scale adoption and deployment of IoT solutions.

No longer will the Internet be only about people, media, and content, but it will also include all real-world assets as

intelligent creatures exchanging information, interacting with people, supporting business processes of enterprises, and creating knowledge (Figure). The IoT is not a new Internet, it is an extension to the existing Internet.

IoT is about the technology, the remote monitoring, and control, and also about where these technologies are applied. IoT can have a focus on the open innovative promises of the technologies at play, and also on advanced and complex processing inside very confined and close environments such as industrial automation. When employing IoT technologies in more closed environments, an alternative interpretation of IoT could then be "Intranet of Things".

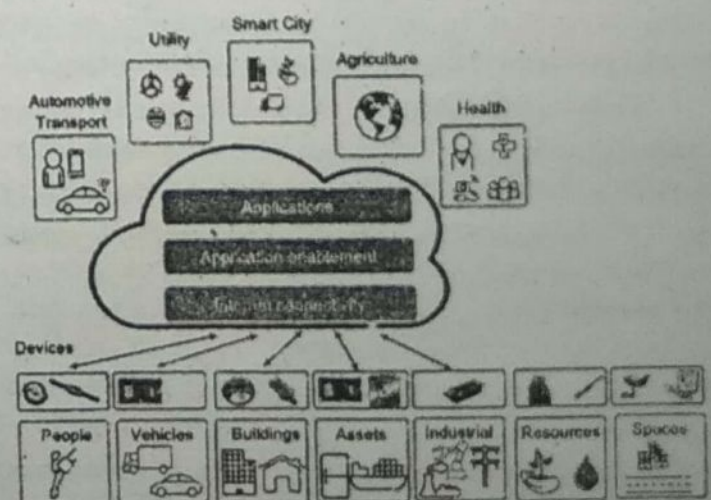


Fig.

Visions put forward (e.g. SENSEI 2013) have included notions like a global open fabric of sensor and actuator services that integrate numerous Wireless Sensor Network

enable organizations to successfully scale their cloud environments." Growth of public and private clouds and enterprises' embrace of distributed multicloud application environments will have an ongoing and significant impact on data center SDN, representing both a challenge and an opportunity for vendors.

Agility is a key attribute of digital transformation, and enterprises will adopt architectures, infrastructures, and technologies that provide for agile deployment, provisioning, and ongoing operational management. In a datacenter networking context, the imperative of digital transformation drives adoption of extensive network automation, including SDN.

Q.21 What are the challenges of NFV?

Ans. NFV challenges: NFV comes with some challenges that require considerations before implementing in existing networks. In large-scale networks, redundancy is a vital characteristic that is essential in order to minimize the downtime of the network if any network element goes down.

With NFV, the deployment should have redundancy at the physical level as well as the virtualized level. For instance, if the redundant switch is hosted on the same physical infrastructure, in the case of a power outage, both switches will fail, leading to a loss of connectivity.

Physical resources required to support a fully virtualized environment are limited. A physical host that supports multiple switch instances will require a network interface card of up to 100 Gbps for optimum functionality. The cost of such hardware to support a fully virtualized environment is on the high side.

Q.22 Write the advantages of network functions virtualization.

Ans. Advantages of Network Functions Virtualization:

- (i) **Reduced Hardware Needs:** By virtualizing your infrastructure you minimize the amount of hardware you need to purchase and maintain. You can also avoid the problem of over provisioning that is common with hardware.
- (ii) **Saving Space and Power:** One of the issues with hardware is that it takes up space and needs to be powered and cooled in order to stay operational. This isn't the same for virtual services which can be managed entirely with software.

(iii) **Lowers Time to Releasing Services:** You can deploy networking services at a faster rate than is possible with hardware. Every time the requirements of your enterprise change you can make a change and keep up quickly.

(iv) **Scalability:** Being able to upscale and downscale services on demand provide you with the long-term capacity potential that you need to be successful in the future.

Q.23 Give tabular difference between SDN and NFV for IoT.

Ans.

| Features | SDN | NFV |
|--------------------------------------|--|---|
| Focus or major role | SDN focuses on data center. | NFV focuses on service providers or operators. |
| Strategy | It splits the control and data forwarding planes. | It replaces hardware network devices with software. |
| Protocol | Uses OpenFlow | Not finalized yet, does support OpenFlow. |
| Where the applications will run? | Applications run on industry standard servers or switches. | Applications run on industry standard servers. |
| Prime initiative supporters | Vendors of enterprise networking software and hardware. | Telecom service providers or operators. |
| Business initiator | Corporate IT | Service provider |
| Customer benefit or end user benefit | Drives down complexity and cost and increases agility. | Drives down complexity and cost and increases agility. |
| Initial applications | Cloud orchestration and networking | Routers, firewalls, gateways, CDN, WAN accelerators, SLA assurance. |
| Formalization body | Open Networking Foundation (ONF) | ETSI NFV Working Group |

(NAT), application specific gateway and Firewall. The Home Gateway provides private IP addresses to each connected device in the home. The Home Gateway provides routing capabilities and translates the private IP addresses to one public address (NAT function). The gateway also provides application specific routing for applications such as VoIP and IPTV.

Figure 3 shows how NFV can be used to virtualize the Home Gateway. The NFV infrastructure in the cloud hosts a virtualized Home Gateway. The virtualized gateway provides private IP addresses to the devices in the home. The virtualized gateway also connects to network services such as VoIP and IPTV.

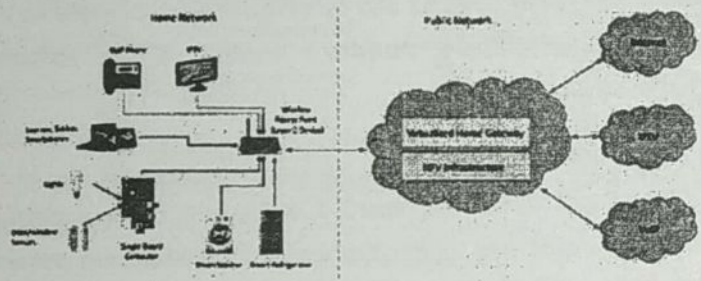


Fig. 2 : Conventional home network architecture

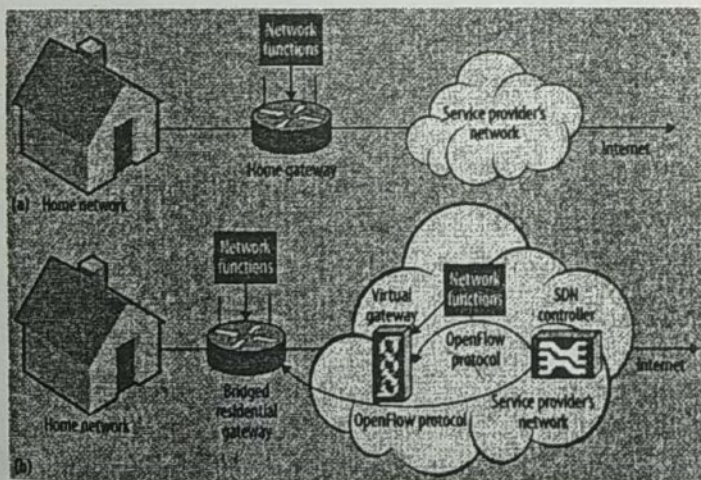


Fig. 3 : Home network with virtualized home gateway

Q.28 What are the characteristics of SDN?

Ans. Characteristics of SDN: SDN can be recognized and distinguished from other innovative networking technologies by the features discussed in the upcoming sections.

Plane Decoupling: Traditional network architecture comprises three distinct planes (control, data and management) which enable full functionalities:

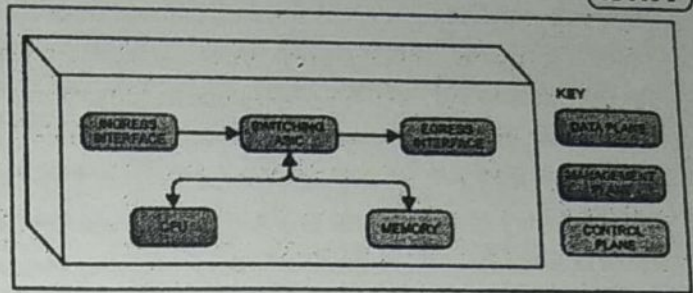


Fig. 1

The main characteristics of SDN are the segregation of the control plane (which determines the way the traffic should be handled) and the data plane (which forwards the traffic based on decisions made by the control plane) based on incoming traffic parameters, such as the MAC address, IP address, and Virtual Local Area Network (VLAN) ID.

In SDN, these policies are determined by the control plane, which is decoupled from the switch (known as the forwarding element) to a logically centralized controller which can physically be distributed and communicated to the forwarding element via a secure link (OpenFlow channel):

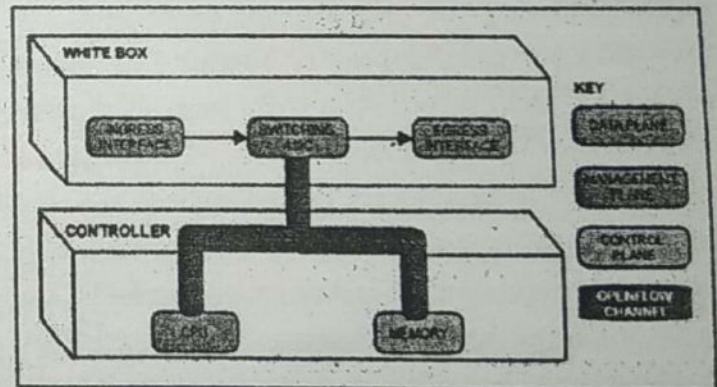


Fig. 2

In the preceding diagram, the forwarding elements that will operate in SDN environments are designed to handle the data plane. The most predominant forwarding elements are designed to support traditional network architecture and SDN network architecture. Control and management plane functionalities are moved to a high-performance server that serves as the controller.

Central Control and Simple Forwarding Elements: Control and management plane hardware and software dedicated resources, which resided on the switches in traditional network architecture, have now been migrated to the controller. This new architecture presents a forwarding

Q.24 What are NFV platform requirements and also write various NFV use cases?

Ans. NFV Platform Requirements: The NFV architecture is designed to augment or replace traditional, highly reliable network appliances. Thus, NFV must deliver the following:

- (i) High performance to 100 Gbps and up
- (ii) High reliability – up time of 99.999%
- (iii) Scalability to millions of users
- (iv) Low-latency delivery of real-time applications
- (v) Ability to integrate with legacy network architectures and link to existing operational and billing systems.

NFV Use Cases: NFV is applicable across a wide range of network functions, including fixed and mobile networks. Some leading NFV applications include:

- (i) Evolved Packet Core (EPC)
- (ii) Software-Defined Branch and SD-WAN
- (iii) IP Multi-Media Subsystem (IMS)
- (iv) Session Border Control (SBC)
- (v) Video Servers
- (vi) Virtual Customer Premises Equipment (VCPE)
- (vii) Content Delivery Networks (CDN)
- (viii) Network Monitoring
- (ix) Network Slicing
- (x) Service Delivery
- (xi) A variety of security functions - firewalls, intrusion detection and prevention systems, NAT, etc.

Q.25 Write the differences between IoT and M2M.

Ans. Difference between IoT and M2M: Though both M2M and IoT involve networking of machines or devices, they differ in the underlying technologies, systems architectures and types of applications.

The differences between M2M and IoT are described as follows:

Communication Protocols: M2M and IoT can differ in how the communication between the machines or devices happens, M2M uses either proprietary or non – IP based communication protocols for communication within the M2M

area networks. Commonly used M2M protocols include ZigBee, Bluetooth, ModBus, M-Bus, Wireless M-Bus, Power Line Communication (PLC), 6LoWPAN, IEEE 802.15.4, Z-Wave, etc. The focus of communication in M2M is usually on the protocols below the network layer. The focus of communication in IoT is usually on the protocols above the network layer such as HTTP, CoAP, WebSockets, MQTT, XMPP, DDS, AMQP, etc. as shown in figure.

Machines in M2M vs Things in IoT: The “Things” in IoT refers to physical objects that have unique identifiers and can sense and communicate with their external environment (and user applications) or their internal physical states. The unique identifiers for the things in IoT are the IP addresses (or MAC addresses). Things have software components for accessing, processing, and storing sensor information, or controlling actuators connected. IoT systems can have heterogeneous things (e.g., a home automation IoT system can include IoT devices of various types, such as fire alarms, door alarms, lighting control devices, etc.) M2M systems, in contrast to IoT, typically have homogeneous machine types within an M2M area network.

Hardware vs Software Emphasis: While the emphasis of M2M is more on hardware with embedded modules, the emphasis of IoT is more on software. IoT devices run specialized software for sensor data collection, data analysis and interfacing with the cloud through IP-based communication.

Data Collection and Analysis: M2M data is collected in point solutions and often in on – premises storage infrastructure. In contrast to M2M, the data in IoT is collected in the cloud (can be public, private or hybrid cloud). The analytics component analyzes the data and stores the results in the cloud database. The IoT data and analysis results are visualized with the cloud-based applications. The centralized controller is aware of the status of all the end nodes and sends control commands to the nodes. Observer nodes can process information and use it for various applications, however, observer nodes do not perform any control functions.

Applications: M2M data is collected in point solutions and can be accessed by on-premises applications such as diagnosis applications, service management applications, and on – premises enterprise applications. IoT data is collected in the cloud and can be accessed by cloud applications such as analytics applications, enterprise applications, remote diagnosis and management applications, etc. Since the scale of data collected in IoT is so massive, cloud-based real-time and batch data analysis frameworks are used for data analysis.

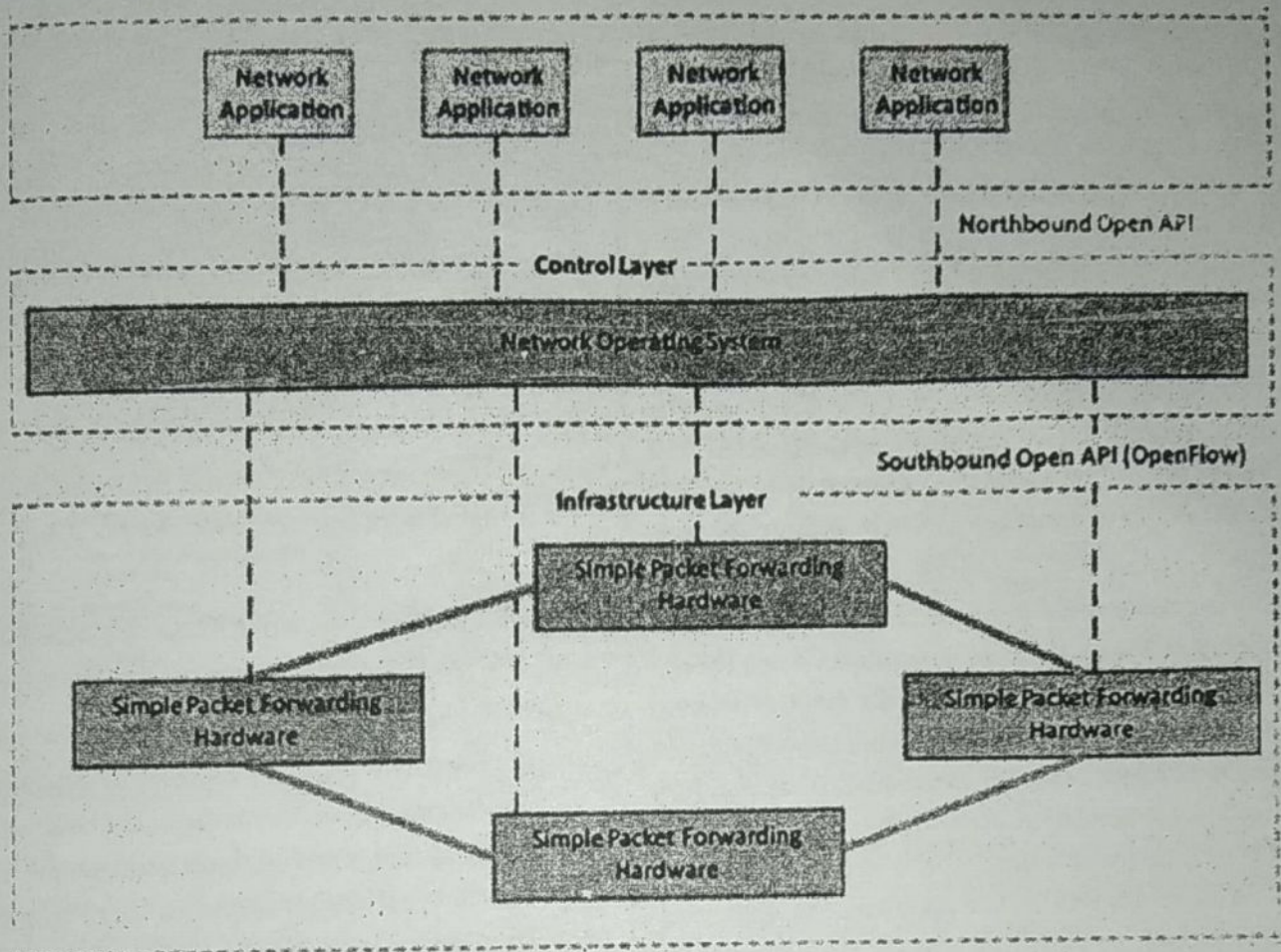


Fig. 1 : SDN architecture

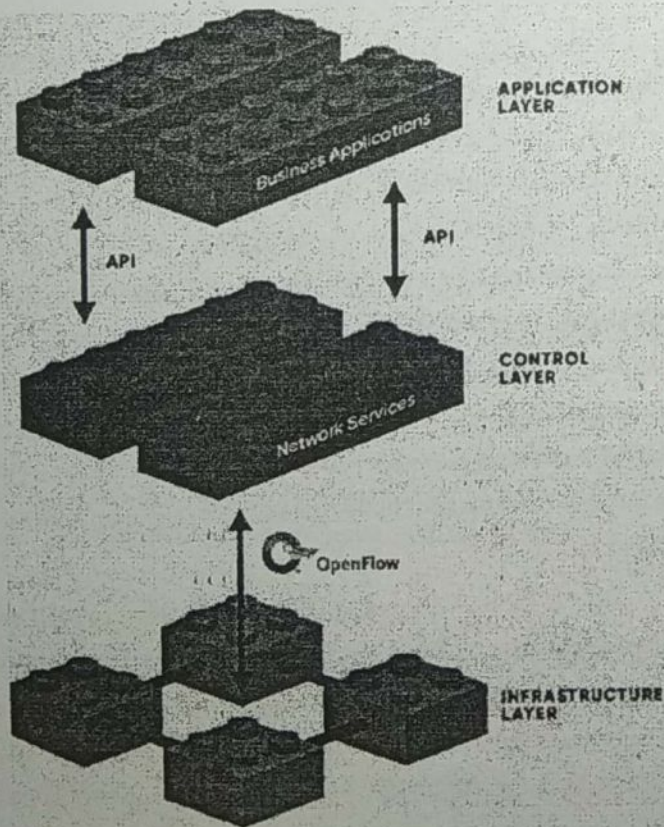


Fig. 2 : SDN layer

OpenFlow uses the concept of flows to identify network traffic based on pre-defined match rules. Flows can be programmed statically or dynamically by the SDN control software. Figure 3 shows the components of an OpenFlow switch comprising of one or more flow tables and a group table, which perform packet lookups and forwarding, and OpenFlow channel to an external controller. OpenFlow protocol is implemented on both sides of the interface between the controller and the network devices. The controller manages the switch via the OpenFlow switch protocol.

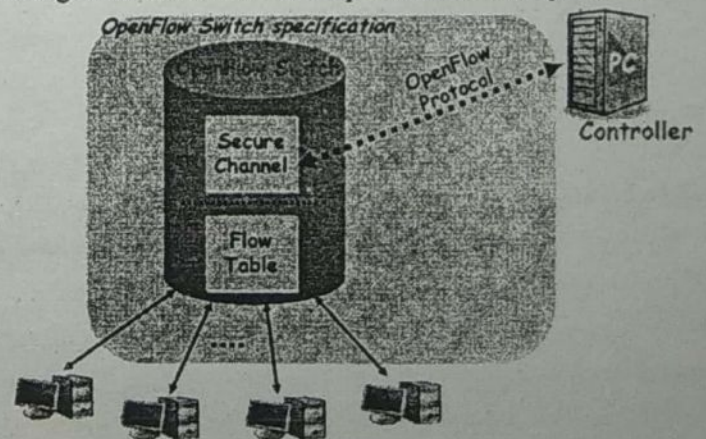


Fig. 3 : Openflow switch

(WSN) deployments and provide different levels of aggregated sensor and actuator services in an open manner for application innovation and for use in not only pure monitor and control type of applications, but also to augment or enrich other types of services with contextual information. IoT applications will not only rely on data and services from sensor and actuators alone. Equally important is the blend – in of other information sources that have relevance from the viewpoint of the physical world. These can be data from Geographic Information Systems (GIS) like road databases and weather forecasting systems, and can be of both a static nature and real-time nature. Even information extracted from social media like Twitter feeds or Facebook status updates that relate to real world observations can be fed into the same IoT system.

Looking towards the applications and services in the IoT, we see that the application opportunities are open – ended, and only imagination will set the limit of what is achievable. Starting from typical M2M applications, one can see application domains emerging that are driven from very diverse needs from across industry, society, and people, and can be of both local interest and global interest. Applications can focus on safety, convenience, or cost reduction, optimizing business processes, or fulfilling various requirements on sustainability and assisted living. Listing all possible application segments is futile, as is providing a ranking of the most important ones. We can point to examples of emerging application domains that are driven by different trends and interests.

Q.27 Explain network function virtualization in detail.

Ans. Network Function Virtualization: Network Function Virtualization (NFV) is a technology that leverages virtualization to consolidate the heterogeneous network devices onto industry standard high volume servers, switches and storage. NFV is complementary to SDN as NFV can provide the infrastructure on which SDN can run. NFV and SDN are mutually beneficial to each other but not dependent. Network functions can be virtualized without SDN, similarly, SDN can run without NFV.

Figure 1 shows the NFV architecture, as being standardized by the European Telecommunications Standards Institute (ETSI). Key elements of the NFV architecture are as follows:

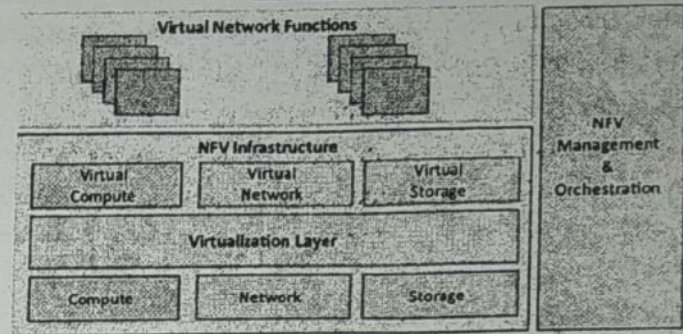


Fig. 1 : NFV architecture

- (i) **Virtualized Network Function (VNF):** VNF is a software implementation of a network function which is capable of running over the NFV Infrastructure (NFVI).
- (ii) **NFV Infrastructure (NFVI):** NFVI includes compute, network and storage resources that are virtualized.
- (iii) **NFV Management and Orchestration:** NFV management and orchestration focuses on all virtualization-specific management tasks and covers the orchestration and life-cycle management of physical and/or software resources that support the infrastructure virtualization, and the life-cycle management of VNFs.

NFV comprises of network functions implemented in software that run on virtualized resources in the cloud. NFV enables separation of network functions which are implemented in software from the underlying hardware. Thus network functions can be easily tested and upgraded by installing new software while the hardware remains the same. Virtualizing network functions reduces the equipment costs and also reduces power consumption. The multi-tenanted nature of the cloud allows virtualized network functions to be shared for multiple network services. NFV is applicable only to data plane and control plane functions in fixed and mobile networks.

Let us look at an example of how NFV can be used for virtualization of the home networks. Figure 2 shows a home network with a Home Gateway that provides Wide Area Network (WAN) connectivity to enable services such as Internet, IPTV, VoIP, etc. The Home Gateway performs various functions including – Dynamic Host Configuration Protocol (DHCP) server, Network Address Translation

The controller can add, update, and delete flow entries in flow tables, figure 4 shows an example of an OpenFlow flow table. Each flow table contains a set of flow entries. Each flow entry consists of match fields, counters, and a set of instructions that apply to matching packets. Matching starts at the first flow table and may continue to additional flow tables of the pipe line.

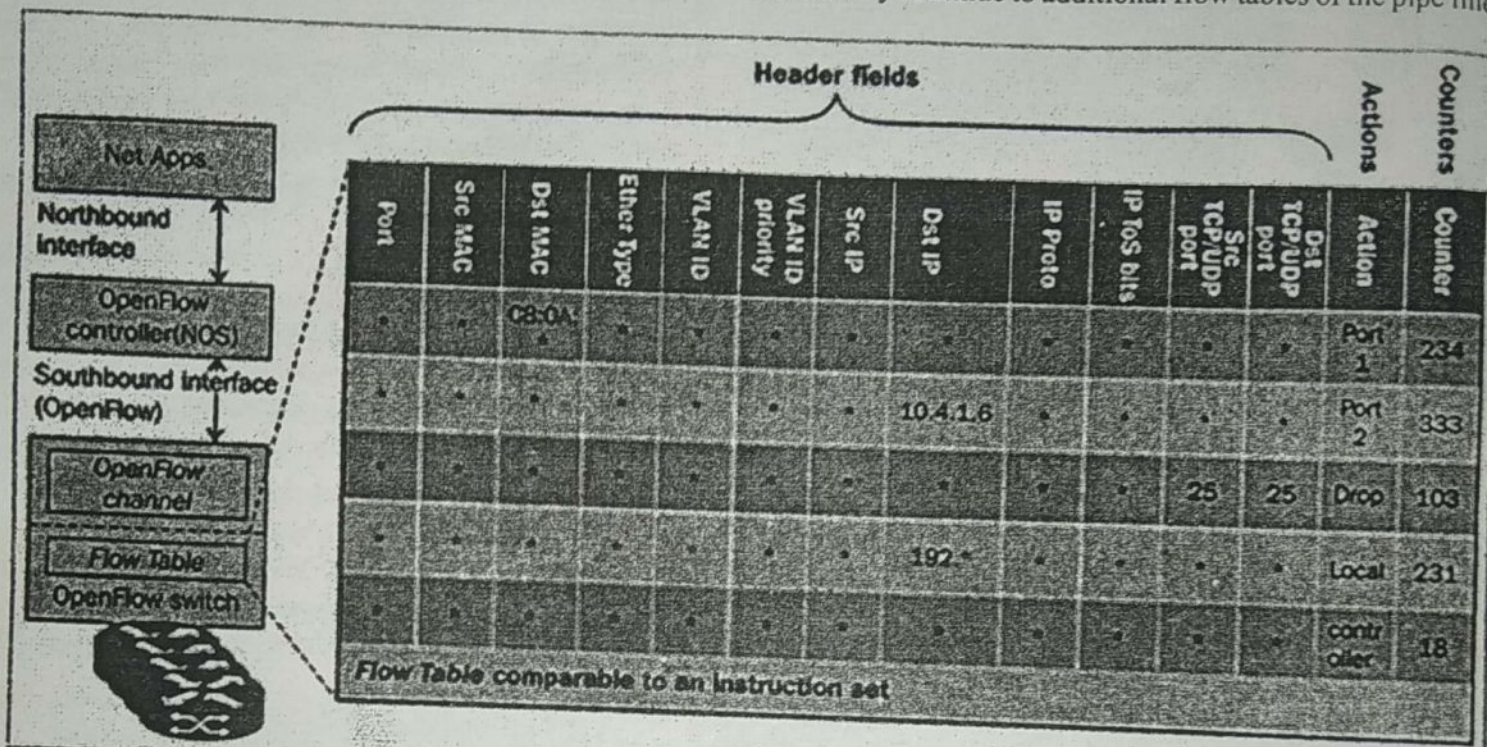


Fig. 4

Q.30 Explain M2M towards IoT- the global context in detail.

Ans. M2M towards IoT – the Global Context: M2M solutions have been around for decades and are quite common in many different scenarios. While the need to remotely monitor and control assets personal, enterprise or other - is not new, a number of concurrent things are now converging to create drivers for change not just within the technology industry, but within the wider global economy and society. Our planet is facing massive challenges environmental, social, and economic. The changes that humanity needs to deal with in the coming decades are unprecedented, not because similar things have not happened before during our common history on this planet, but because many of them are happening at the same time. From constraints on natural resources to a reconfiguration of the world's economy, many people are looking to technology to assist with these issues.

Essentially, therefore, a set of megatrends are combining to create needs and capabilities, which in turn

produce a set of IoT Technology and Business Drivers. This is illustrated in fig.

A megatrend is a pattern or trend that will have a fundamental and global impact on society at a macro level over several generations. It is something that will have a significant impact on the world in the foreseeable future. We here imply both game changers as challenges, as well as technology and science to meet these challenges. For the sake of simplicity, we also provide table as a summary of the main game changers, technology and science trends, capabilities, and implications for IoT.

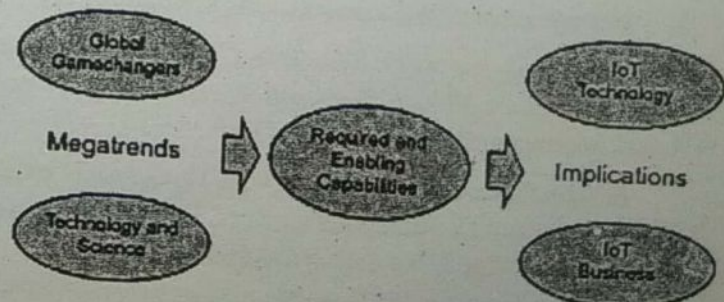
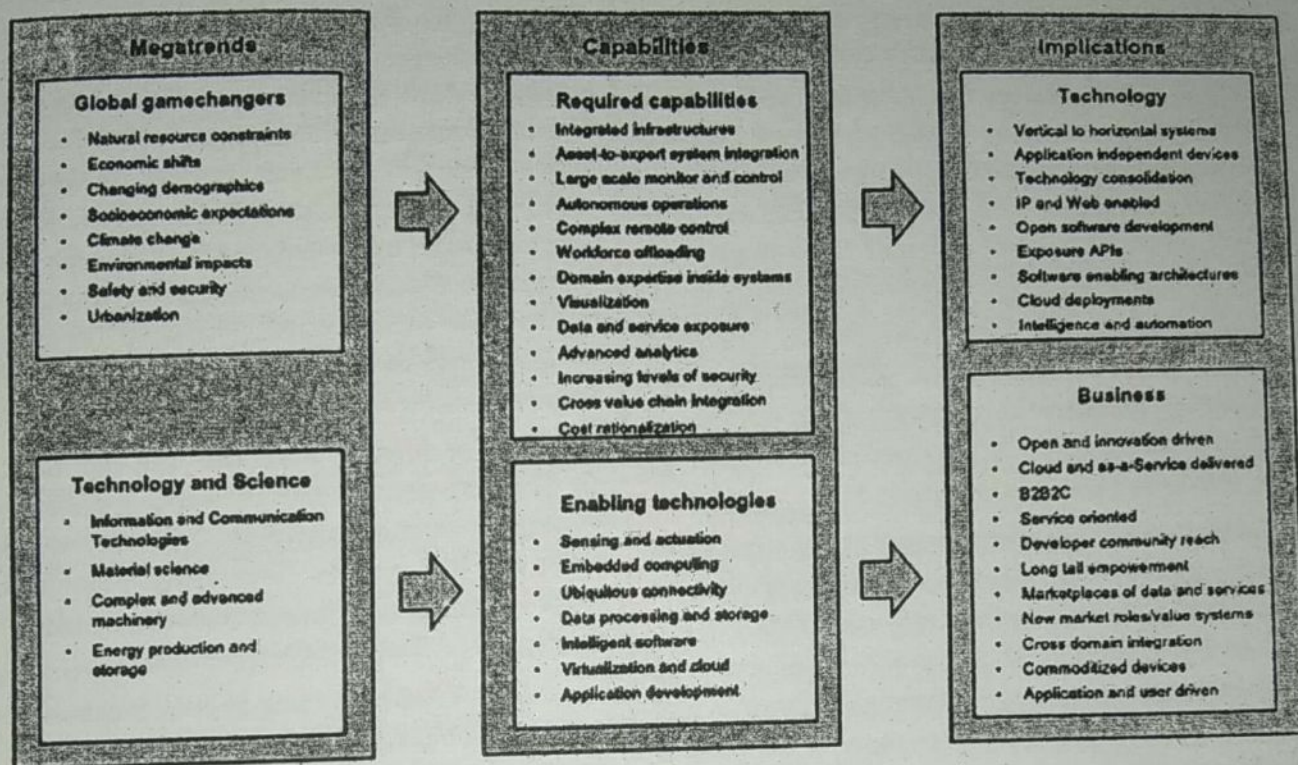


Fig.

Table : A summary of Megatrends, Capabilities and IoT Implications



1. Game Changers: The game changers come from a set of social, economic and environmental shifts that create pressure for solutions to address issues and problems, but also opportunities to reformulate the manner in which our world faces them. There is an extremely strong emerging demand for monitoring, controlling and understanding the physical world, and the game changers are working in conjunction with technological and scientific advances. The transition from M2M towards IoT is one of the key facets of the technology evolution required to face these challenges. We outline some of these more globally significant game changers below, and their relationship to IoT:

- (i) **Natural Resource Constraints:** The world needs to increasingly do more with less, from raw materials to energy, water or food, the growing global population and associated economic growth demands put increasing constraints on the use of resources. The use of IoT to increase yields, improve productivity, and decrease loss across global supply chains is therefore escalating.
- (ii) **Economic Shifts:** The overall economy is in a state of flux as it moves from the post-industrial era to a digital economy. One example of this is found in the move from product-oriented to service-oriented economies. This implies a lifetime responsibility of the product used in the service offering, and will in many

cases require the products to be connected and contain embedded technologies for gathering data and information. At the same time, there are fluctuations in global economic leadership. Economies across the globe must handle the evolving nature of these forces. As technology becomes increasingly embedded and more tasks automated, countries need to manage this shift and ensure that M2M and IoT also create new jobs and industries.

- (iii) **Changing Demographics:** With increased prosperity, there will be a shift in the demographic structures around the world. Many countries will need to deal with an aging population without increasing economic expenditure. As a result, IoT will need to be used, for example, to help provide assisted living and reduce costs in healthcare and emerging "wellcare" system.
- (iv) **Socioeconomic Expectations:** The global emerging middle class results in increasing expectations on well-being and corporate social responsibility. Lifestyle and convenience will be increasingly enabled by technology as the same disruption and efficiency practices evident in industries will be applied within people's lives and homes as well.
- (v) **Climate Change and Environmental Impacts:** The impact of human activities on the environment and climate has been long debated, but is now in essence

element, which maximizes the overall resource management in the topology as the hardware processes less complex codes for forwarding the traffic. These complex algorithms now exist in the controller, and traffic forwarding decisions are made from them, which communicates the best forwarding path for every packet to the forwarding element through a secure channel from the controller to the forwarding elements. These characteristics allow a simpler ASIC to be incorporated into the forwarding elements existing in an SDN infrastructure. This also allows the provisioning of ample resources with respect to the growth in the network size.

Network Automation and Virtualization: Network automation can be described as a process by which tools are deployed, which allows the automation of configuration, management, and operations of the network by the network administrator. As a result of this, the network administrator has the ability to tailor the network to fulfill the business requirement in real time. The SDN architecture better supports network automation in comparison to traditional network architecture.

Ansible and Puppet are common examples of automation and orchestration tools that assist network administrators with tasks ranging from the management of configuration to deployment of applications seamlessly. Automation makes the network flexible, resilient, easy to manage, and responsive to business needs in real time, which results in reduced operating expenses.

Network virtualization is the abstraction of the physical network to support the running of multiple network logical instances on a common shared physical element. This supports rapid innovation, as services can be at software speed across the entire network.

SDN controllers provide both automation and virtualization to the network by utilizing the northbound and southbound API to communicate with the applications and forwarding elements.

Q.29 What are the key elements of SDN? Explain in details.

Ans. SDN attempts to create network architectures that are

simpler, inexpensive, scalable, agile and easy to manage. Figure 1 and 2 show the SDN architecture and the SDN layers in which the control and data planes are decoupled and the network controller is centralized. Software-based SDN controllers maintain a unified view of the network and make configuration, management and provisioning simpler. The underlying infrastructure in SDN uses simple packet forwarding hardware as opposed to specialized hardware in conventional networks. The underlying network infrastructure is abstracted from the applications. Network devices become simple with SDN as they do not require implementations of a large number of protocols. Network devices receive instructions from the SDN controller on how to forward the packets. These devices can be simpler and cost less as they can be built from standard hardware and software components.

Key elements of SDN are as follows:

- (i) **Centralized Network Controller:** With decoupled control and data planes and centralized network controller, the network administrators can rapidly configure the network. SDN applications can be deployed through programmable open APIs. This speeds up innovation as the network administrators no longer need to wait for the device vendors to embed new features in their proprietary hardware.
- (ii) **Programmable Open APIs:** SDN architecture supports programmable open APIs for interface between the SDN application and control layers (Northbound interface). With these open APIs various network services can be implemented, such as routing, quality of service (QoS), access control, etc.
- (iii) **Standard Communication Interface (OpenFlow):** SDN architecture uses a standard communication interface between the control and infrastructure layers (Southbound interface). OpenFlow, which is defined by the Open Networking Foundation (ONF) is the broadly accepted SDN protocol for the Southbound interface. With OpenFlow, the forwarding plane of the network devices can be directly accessed and manipulated.

scientifically proven. Technology, including IoT, will need to be applied to aggressively reduce the impact of human activity on the earth's systems.

- (vi) **Safety and Security:** Public safety and national security becomes more urgent as society becomes more advanced, but also more vulnerable. This has to do both with reducing fatalities and health as well as crime prevention, and different technologies can address a number of the issues at hand.

2. General Technology and Scientific Trends:

Technological and scientific advances and breakthroughs are occurring across a number of disciplines at an increasing pace. Below is a brief description of the science and technology advances that have a direct relevance to IoT.

Material science has a large impact across a vast range of industries, from pharmaceutical and cosmetics to electronics. Micro Electro Mechanical Systems (MEMS) can be used to build advanced micro-sized sensors like accelerometers and gyroscopes. Emerging flexible and printable electronics will enable a new range of innovations for embedding technology in the real world. New materials provide different methods to develop and manufacture a large range of different sensors and actuators, as well being used in applications for environmental control, water purification, etc. Additionally, we will see other innovative uses such as smart textiles that will provide the capability to produce the next generation of wearable technologies. From an IoT perspective, these advances in material science will see an increasing range of applications and also a broader definition of what is meant by a sensor.

Complex and advanced machinery refers to tools that are autonomous or semi-autonomous. Today they are used in a number of different industries; for example, robots and very advanced machinery is used in different harsh environments, such as deep-sea exploration, or in the mining industry in solutions such as Rio Tinto's Mine of the Future (Rio Tinto 2012). Advanced machines have many modalities, and operate with a combination of local autonomous capabilities as well as remote control. Sensing and actuation are key technologies, and local monitor – control loops for routine tasks are required in addition to reliable communications for remote operations. Often such solutions require real – time characteristics. These systems will continue to evolve and automate tasks today performed by humans – even self – driving cars have started to make headlines thanks to Google.

3. Trends in Information and Communications Technologies:

While significant advances in the fields of material science, advanced and complex machinery, and energy production and storage will have an impact on IoT, first and foremost, ICT advances will drive the manner in which these solutions are provided as they are the core enabling factors behind M2M and IoT. Ever since the development of integrated circuits during the late 1950s and early 1960s, these technologies have had an increasing impact on enterprises and society. The increasing rates of change have led to a situation where it is now cheap enough to “sense the planet”.

Today, sensors, actuators, and tags function as the digital interfaces to the physical world. Small-scale and cheap sensors and actuators provide the bridge between the physical realm and ICT systems. Tags using technologies such as RFID provide the means to put electronic identities on any object, and can be cheaply produced.

Embedded processing is evolving, not only towards higher capabilities and processing speeds, but also extending towards the smallest of applications. There is a growing market for small-scale embedded processing such as 8, 16, and 32-bit microcontrollers with on-chip RAM and flash memory, I/O capabilities, and networking interfaces such as IEEE 802.15.4 that are integrated on tiny System – on – a – Chip (SoC) solutions. These enable very constrained devices with a small footprint of a few mm² and very low power consumption (in the milli- to micro- Watt range), yet are still capable of hosting an entire TCP/IP stack including a small web server.

Q.31 Describe the architecture of NFV.

Ans. Architecture of NFV : The Network Functions Virtualization (NFV) is a network architecture or concept that utilizes the IT technology fundamentals to virtualize entire network node functions onto industry standard high volume servers, switches and storage, which could be located in data centers or centralized locations. Network nodes are in the end user premises to create communication services and illustrated in fig.1.

It involves implementing network functions in a software that can run on a range of industry standard server hardware, and that can be moved to, or instantiated in various locations in the network as in when required, without the need to install new hardware equipment.

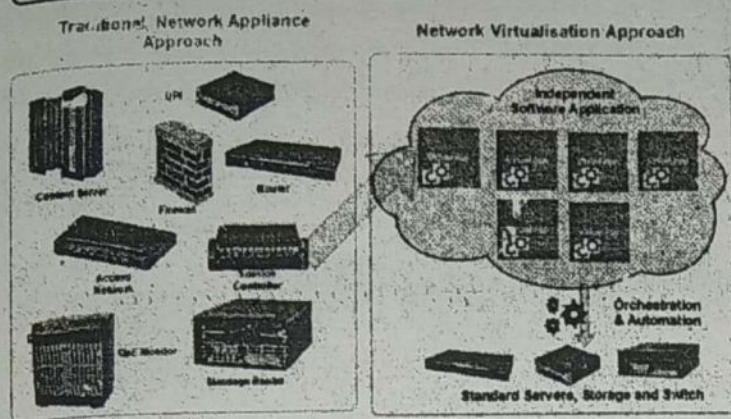


Fig. 1

ETSI NFVI Architecture: ETSI has created different standards, the one provided below is one of the most important, which illustrates how the NFVI help us to decouple the hardware and software.

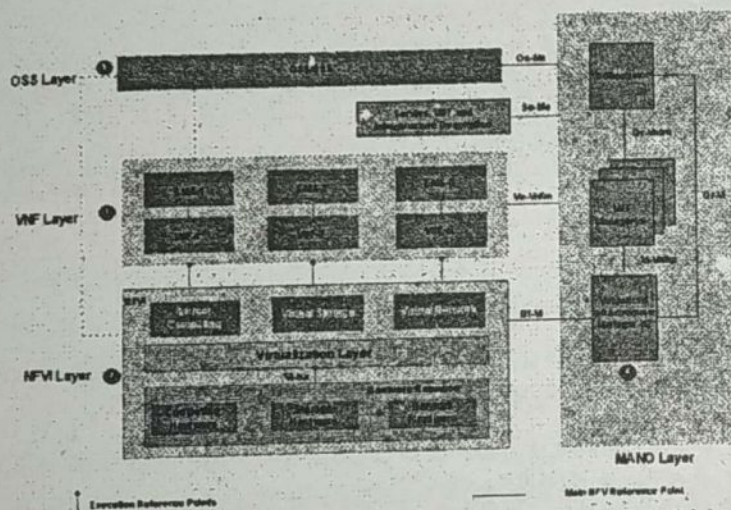


Fig. 2

NFV blocks are shown in fig.2. It can be divided into four layers :

1. Virtualization Network Function (VNF) Layer
2. NFV Infrastructure (NFVI) Layer
3. Operation Support Subsystem (OSS) Layer
4. Management and Orchestration (MANO) Layer

1. Virtualization Network Function (VNF) Layer: It has two subsections : Virtual Network Function (VNF) and Element Management System (EMS).

A Virtual Network Function (VNF) is the basic block in NFV architecture. It virtualized network function. e.g. when a router is virtualized, we call it Router VNF and when a base station is virtual we call it as base station VNF, similarly, it can be DHCP server VNF and Firewall VNF. Even when one sub-function of a network element is virtualized, it is called

VNF. For example in Evolved Packet Corer case, various sub-functions like MME, Gateways, and HSS can be separate VNFs which together function as virtual EPC.

A VNFs are deployed on Virtual Machines (VMs). A VNF can be deployed on multiple VMs where each VM hosts a single function of VNF. However, the whole VNF can also be deployed be on a single VM as well.

Element Management System (EMS) is responsible for the functional management of VNF. The manager functions include fault, configuration, accounting, performance and security management. An EMS may manage the VNFs through proprietary interfaces. There may be one EMS per VNF or one EMS that can manage multiple VNFs. EMS itself can be deployed as Virtual Network Function (VNF).

2. NFV Infrastructure (NFVI) Layer: NFV Infrastructure is the totality of hardware and software components which build up the environment in which VNFs are deployed, managed and executed. NFV infrastructure physically can span across several locations, the network provides connectivity between these locations to be part of NFV infrastructure.

NFV Infrastructure includes following :

- (i) Hardware Resources
- (ii) Virtualization Layer
- (iii) Virtual Resources

From VNF point of view, the virtualization layer and hardware resources shall be a single entity providing it the desired resource.

Hardware resource includes computing, storage and network the provides processing, storage and connectivity to VNFs through virtualization (hypervisor) layer: Computing and storage resources are commonly used in a pool. The network resource comprises of switching functions e.g. router, wired or wireless network

Virtualization layer also known as a hypervisor, it abstracts the hardware resources and decouples the VNF software from the underlying hardware to ensure a hardware independent life cycle for VNFs. It is mainly responsible for following:

- (i) Abstracting and logically partitioning physical resources, commonly as hardware abstraction layer.
- (ii) Enabling the software to implement the VNF to use the underlying Virtualization Infrastructure.
- (iii) Providing the virtualised resources to VNF, so that latter can be executed.

The virtualization layer in middle ensures VNFs are decoupled from hardware resource and therefore software can be deployed on different physical resources.

Virtual Resources: Virtualization layer abstracts the computing, storage, and network from hardware layer make available as virtual resources.

3. Operation Support Subsystem (OSS)/Business Support System (BSS) Layer: OSS/BSS refers to OSS/BSS of an operator. OSS deals with network management, fault management, configuration management and service management. BSS deals with customer management, product management and order management etc.

In the NFV architecture, the decoupled BSS/OSS of an operator may be integrated with the NFV Management and Orchestration using standard interfaces.

4. Management and Orchestration (MANO) Layer: Management and Orchestration layer is also abbreviated as MANO and it includes three components:

- (i) Virtualized Infrastructure Manager(s)
- (ii) VNF Manager(s)
- (iii) Orchestrator

MANO interacts with both NFVI and VNF layer. MANO layer manages all the resources in the infrastructure layer, it also creates and deletes resources and manages their allocation of the VNFs.

Virtualised Infrastructure Manager (VIM) comprises the functionalities that are used to control and manage the interaction of a VNF with computing, storage and network

resources under its authority, as well as their virtualisation. Virtualised infrastructure manager performs the following:

- (i) Inventory of software, computing, storage and network resources dedicated to NFV infrastructure.
- (ii) Management of infrastructure resource and allocation e.g. increasing the VMs, increasing energy efficiency etc.
- (iii) Allocation of VMs on hypervisors, compute resources, storage, and relevant network connectivity.
- (iv) Root cause analysis of performance issues from the NFV infrastructure perspective.
- (v) Collection of infrastructure fault information.
- (vi) Collection of information for capacity planning, monitoring and optimization.

VNF Manager is responsible for VNF life cycle management which includes installation, updates, query, scale up/down and termination. A VNF manager may be deployed for each VNF or a single VNF manager may be deployed to serve multiple VNFs.

Orchestrator is in charge of the orchestration and management of NFV infrastructure and software resources and realizing network services.

There is one more independent block known as service, VNF and Infrastructure apart from above building blocks. This includes data-sets that provide information regarding VNF deployment template, VNF forwarding graphs, service related information and NFV infrastructure information models.

□□□

Ans. Effects to Executing Web of Things on Agriculture Segment : They are distinctive components that leave a positive effect on actualizing internet of things on agriculture area.

With regards to the farming division, it is tied in with being exact and clear about what should be done at the perfect time. This can be automatized utilizing the internet of things where profitability, exactness, increment in effectiveness and lessening the general expenses related.

For cultivating, it is about explanatory information. For instance: To yield a superior outcome it is vital for the rancher to comprehend the dirt quality, air quality, water accessibility, climate related data, bug control and so forth. The majority of this data can be assembled by utilizing the Internet of things applications.

Along these lines understanding distinctive parts of the cultivating, it is certainly advantageous for the agriculturist to look for this kind of assistance.

Q.12 What effects will the internet of things (IoT) have on energy sector?

Ans. Effects of the Internet of Things on Energy Sector : Within the vitality part, the IoT may affect both generation and conveyance, for instance through encouraging the observing of oil wellheads and pipelines. At the point when IoT segments are implanted into parts of the electrical lattice, the subsequent framework is generally alluded to as the "brilliant network". This utilization of IoT empowers more noteworthy control by utilities over the stream of power and can improve the proficiency of lattice activities. It can likewise facilitate the mix of microgenerators into the framework.

Savvy lattice innovation can likewise furnish buyers with more noteworthy learning and control of their vitality use using shrewd meters in the home or office. The association of brilliant meters to a building's HVAC, lighting, and different frameworks can result in "keen structures" that coordinate the activity of those frameworks. Keen structures use sensors and other information to naturally modify room temperatures, lighting, and in general vitality use, bringing about more noteworthy proficiency and lower vitality cost. Data from nearby structures might be additionally coordinated to give extra efficiencies in an area or bigger division in a city.

Q.13 What is implied by a smart city, with regards to Internet of things (IoT)?

Ans. As with IoT and other prevalent innovation terms, there is no settled agreement definition or set of criteria for portraying what a shrewd city is. Explicit portrayals fluctuate broadly, however when all is said in done, they include the utilization of IoT and related innovations to enhance vitality, transportation, administration, and other metropolitan administrations for determined objectives, for example, supportability or enhanced personal satisfaction.

The related innovations incorporate are as follows:

- (i) Online networking, (for example, Facebook and Twitter),
- (ii) Portable registering, (for example, cell phones and wearable gadgets).
- (iii) Information Analytics (enormous information like the handling and utilization of substantial informational indexes, and open information and databases that are openly available).
- (iv) Distributed computing (the conveyance of registering administrations from a remote area, comparable to the way utilities, for example, power are given).

Q.14 Define smart lighting with neat figure.

Ans. Smart Lighting : Smart lighting is the umbrella that encompasses different solid state technologies such as LEDs and OLEDs to illuminate indoor and outdoor environments. Smart lighting systems mainly involve digital sensors, actuators drivers and communications interfaces. These lighting systems are programmed using advanced control algorithms and can be organized into lighting networks to operate remotely. Some of the most popular solutions are designed to change the light spectrum or color. They can also control the level of illumination in a room when an external event occurs, for example, when a user has been detected by an occupancy sensor or when an event occurs such as the detection of vehicles or people on a road.

The smart lighting system eliminates the need to operate the overall system in manual mode. The lighting network is programmed with an initial setup; however, each independent

CASE STUDY OF IoT APPLICATIONS

5

IMPORTANT QUESTIONS

PART-A

Q.1 Write the impacts of the Internet of Things (IoT) on energy sector.

Ans. Impacts of the Internet of Things (IoT) on Energy Sector : IoT might impact each production and delivery, as an example through facilitating observance of oil wellheads and pipelines. When IoT parts are embedded into components of the electrical grid, the ensuing infrastructure is usually mentioned because the "smart grid". This use of IoT allows bigger management by utilities over the flow of electricity and may enhance the potency of grid operations.

Q.2 Write impacts of Internet of Things (IoT) on agriculture sector.

Ans. Impacts of Internet of Things (IoT) on Agriculture Sector : The IoT may be leveraged by the agriculture trade through exactness agriculture, with the goal of optimizing production and potency whereas reducing prices and environmental impacts. For farming operations, it involves analysis of elaborate, usually time period knowledge on weather, soil and air quality, installation, pesterer populations, crop maturity, and alternative factors like the cost and availability of equipment and labor. Field sensors check soil wetness and beam balance, which might be in addition to location technologies to modify precise irrigation and fertilization.

Q.3 What effects will the Internet of Things (IoT) have on manufacturing sector?

Ans. Effects of Internet of Things (IoT) on Manufacturing Sector : Integration of IoT advancements into assembling and inventory network coordinations is anticipated to transformatively affect the division. The greatest effect might be acknowledged in streamlining of activities, making producing forms increasingly proficient. Efficiencies can be accomplished by associating parts of industrial facilities to upgrade creation, yet additionally by interfacing segments of stock and dispatching for production network improvement.

Another application is prescient support, which utilizes sensors to screen apparatus and production line foundation for harm. Coming about information can empower support groups to supplant parts before conceivably hazardous as well as exorbitant breakdowns happen.

Q.4 What do you mean by smart cities and the role of IoT in it?

Ans. Smart city is a initiation taken by various governments across the world to equip their city/town with smart solutions such as smart cameras, access control, medical IOT devices and etc so that the services to the general public can be monitored and enhanced in a futuristic way.

Q.5 What is applications of IoT in environmental monitoring?

Home Intrusion Detection System: Door sensors, and raise alerts are used in home IDS if necessary to record the data. Read the motion of sensor data at each door, detect door sensor and motion of opening, each sensor at regular intervals detected the motion, and events stored in the and alert sent to the database. Based on the analysis of stored data with new data find the intrusion and report accordingly virtual entities, room with attribute states, using REST services. If an intrusion is detected, means send the information to the owner of the house, or Police, etc. Figure shows home IDS and table shows the alarm status.

Table : IDS dashboard alarms status

| Room No. | Doors | Alarm status | Doors |
|----------|-------|--------------|-------|
| Room2 | Door1 | ON/OFF | Door2 |
| Room1 | Door2 | ON/OFF | Door3 |

(b) Smoke/Gas Detectors: Smoke detectors are installed in homes and buildings to detect smoke that is typically an early sign of fire. Smoke detectors use optical detection, ionization or air sampling techniques to detect smoke. Alerts raised by smoke detectors can be in the form of signals to a fire alarm system. Gas detectors can detect the presence of harmful gases such as carbon monoxide (CO), liquid petroleum gas (LPG), etc. A smart smoke/gas detector can raise alerts in human voice describing where the problem is, send or an SMS or email to the user or the local fire safety department and provide visual feedback on its status (healthy, battery-low, etc.). In the design of a system that detects gas leakage and smoke and gives visual level indication.

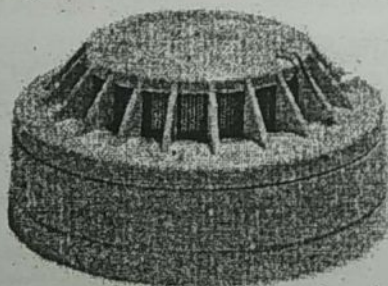


Fig. : Smoke detector

Q.17 What do you understand by smart grids? Write its domain.

Ans. Smart Grid: The smart grid is proposed to solve the issues of electricity grid (e.g., low reliability, high outages, high greenhouse gas and carbon emission, economics, safety,

and energy security). One of the definitions for the smart grid is that the smart grid is a communication network on top of the electricity grid to gather and analyze data from different components of a power grid to predict power supply and demand which can be used for power management.

In a proposed model for the smart grid by the national institute of standards and technology, the smart grid has 7 domains and roles of these domains are defined so that required information can be exchanged and necessary decisions can be made. Some of the required functionalities to deploy the smart grid are as follows :

1. **Communication Networks:** Public, private, wired, and wireless communication networks that can be used as the communication infrastructure for smart grid.
2. **Cybersecurity:** Determining measures to guarantee availability, integrity, and confidentiality of the communication and control systems which are required to manage, operate, and protect smart grid infrastructures.
3. **Distributed energy resources:** Using different kinds of generation (e.g., renewable energies) and/or storage systems (batteries, plug-in electric cars with bi-directional chargers) that are connected to distributed systems.
4. **Distribution Grid Management:** Trying to maximize the performance of components in distribution systems such as feeders and transformers and integrate them with transmission systems, increase reliability, increase the distribution system efficiency, and improve management of distributed renewable energy sources.
5. **Electric Transportation:** Integrating plug-in electric vehicles in a large – scale.
6. **Energy Efficiency:** Providing mechanisms for different kinds of customers to modify their energy usage during peak hours and optimizing the balance between power supply and demand.
7. **Energy Storage:** Using direct or indirect energy storage technologies such as pumped hydroelectric storage technology.
8. **Wide – area Monitoring:** Monitoring of power system components over a large geographic area to optimize their performance and preventing problems before they happen.

Ans. Applications of IoT in Environmental Monitoring : The application of IoT in environmental monitoring are broad environmental protection, extreme weather monitoring, water safety endangered species protection, commercial farming and more. In these applications, sensors notice and live each variety of environmental amendment.

Q.6 What is application of IoT in transportation?

Ans. Applications of IoT in Transportation : At each layer of transportation, IoT provides improved communication, control, and knowledge distribution. These applications embrace personal vehicles, industrial vehicles, trains, UAVs, and alternative instrumentation. It extends throughout the complete system of all transportation parts like control, parking, fuel consumption, and more.

Q.7 What is application of IoT in Government?

Ans. Applications of IoT in Government : IoT supports the event of sensible nations and sensible cities. This includes sweetening of infrastructure antecedently mentioned (e.g., healthcare, energy, transportation, etc.), defense, and conjointly the engineering and maintenance of communities.

Q.8 What is application of IoT in Law enforcement?

Ans. Applications of IoT in Law Enforcement : IoT enhances enforcement organizations and observe, and improves the justice system. The technology boosts transparency, distributes vital information, and removes human intervention wherever it proves excess.

IoT aids in making higher solutions to issues by exploitation technology within the place of force; for instance, light-weight in-person investigations of suspicious activities may be replaced with remote observation, logged footage of violations, and electronic ticketing. It conjointly reduces corruption by removing human management and opinion for a few violations.

Q.9 What is AMI in smart grid?

Ans. Advanced Metering Infrastructure (AMI): AMI as one of the key components of SG creates a bidirectional

communication network between smart meters (SMS) and utility system to collect, send, and analyze consumer energy consumption data.

PART-B

Q.10 What is the effect of the internet of things have on healthcare segment?

Ans. Effect of the Internet of Things on Healthcare Segment : To be completely forthright, with you there will be huge development when the internet of things administrations are executed in the medicinal services segment. As a matter of first importance and the most serious issue is giving quality human services administration to each individual has been a noteworthy test for every one of the nations. With the assistance of the internet of things, a great deal of brilliant gadgets and hardware have been intended to take into account this issue. The methodology of telemedicine and telehealth is one of the greatest accomplishment in this cutting edge period. This is happened on account of the solid system channel coordinated with a wide assortment of brilliant apparatuses which are used in everyday human services assessment of a person. Starting at now, we have seen organizations developing in this space where they are endeavoring to give specialist interview for all intents and purposes when they require it during the evening or in a crisis level.

It has a great deal of positive effect as far as the administration given by the organizations and in next couple of years the specialist and patient communication will totally rethink and it will understand a ton of current day challenges. Some of them can be as followings :

- Booking arrangements in a standard timetable.
- Traveling to the healing facility with the patient cooperating with ten distinct individuals about the sickness and after that at last go see a specialist and so forth.

Q.11 What are the effects that can be seen in executing web of things on agriculture segment?

Q.18 What are the applications and services associated with smart grids?

Ans. IoT Applications and Services in SG: IoT can support technologies in SG. Comprehensive sensing and processing abilities of IoT can improve SG abilities such as processing, warning, self-healing, disaster recovery, and reliability. Combining IoT and SG can greatly promote the development of smart terminals, meters and sensors, information equipment, and communication devices. IoT can be used to accomplish reliable data transmission in wire and wireless communication infrastructures in different parts of SG (electricity generation, transmission lines, distribution, and consumption/utilization) as follows:

1. In electricity generation, IoT can be used to monitor electricity generation of different kinds of power plants (such as coal, wind, solar, biomass), gas emissions, energy storage, energy consumption, and predict necessary power to supply consumers.
2. IoT can be used to acquire electricity consumption, dispatch, monitor and protect transmission lines, substations, and towers, manage and control equipment.
3. IoT can be used in customer side in smart meters to measure different types of parameters, intelligent power consumption, interoperability between different networks, charging and discharging of electric vehicles, manage energy efficiency and power demand.

The main IoT application scenarios are as follows.

1. **AMI with High Reliability:** AMI is a key component in SG. IoT can be used in AMI to collect data, measure abnormality in SG, exchange information between smart meters, monitor electricity quality and distributed energy, analyze user consumption pattern.
2. **Smart Home:** A smart home can be used to interact with users and SG, enhance SG services, meet marketing demand, improve QoS, control smart appliances, read power consumption information which is gathered by smart meters, and monitor renewable energy.
3. **Transmission Line Monitoring:** By using wireless broadband communication technologies, the transmission lines can be monitored to discover fault issues and eliminate them.

4. **Electric Vehicle (EV) Assistant Management System:** EV assistant management systems comprise of charging station, EV, and monitoring center. With GPS, users can inspect nearby charging stations and their parking information. The GPS will automatically guide drivers to the most suitable charging station. The monitoring center manages car batteries, charging equipment, charging stations and optimize resources.

Q.19 Explain in the following terms related with energy

- (a) Renewable Energy System
- (b) Prognostics

Ans.(a) Renewable Energy Systems: Due to the variability in the output from renewable energy sources (such as solar and wind), integrating them into the grid can cause grid stability and reliability problems. Variable output produces local voltage swings that can impact power quality. Existing grids were designed to handle power flows from centralized generation sources to the loads through transmission and distribution lines. When distributed renewable energy sources are integrated into the grid, they create power bi-directional power flows for which the grids were not originally designed. IoT based systems integrated with the transformers at the point of interconnection measure the electrical variables and how much power is fed into the grid. To ensure the grid stability, one solution is to simply cut off the overproduction. For wind energy systems, closed-loop controls can be used to regulate the voltage at point of interconnection which coordinate wind turbine outputs and provides reactive power support.

(b) Prognostics: Energy systems (smart grids, power plants, wind turbine farms, for instance) have a large number of critical components that must function correctly so that the systems can perform their operations correctly. For example, a wind turbine has a number of critical components, e.g., bearings, turning gears, for instance, that must be monitored carefully as wear and tear in such critical components or sudden change in operating conditions of the machines can result in failures. In systems such as power grids, real-time information is collected using specialized electrical sensors called Phasor Measurement Units (PMU) at the substations. The information received from PMUs must be monitored in real-time for estimating the state of the system and for

predicting failures. Energy systems have thousands of sensors that gather real-time maintenance data continuously for condition monitoring and failure prediction purposes. IoT based prognostic real-time health management systems can predict performance of machines or energy systems by analyzing the extent of deviation of a system from its normal operating profiles. Analyzing massive amounts of maintenance data collected from sensors in energy systems and equipment can provide predictions for the impending failures (potentially in real-time) so that their reliability and availability can be improved. Prognostic health management systems have been developed for different energy systems, Open PDC is a set of applications for processing of streaming time-series data collected from Phasor Measurement Units (PMU) in real-time. A generic framework for storage, processing and analysis of massive machine maintenance data, collected from a large number of sensors embedded in industrial machines.

Q.20 What are the requirements for using IoT in smart grids?

Ans. To use IoT in SG, we should have some technologies and satisfy some requirements which are listed as follows:

1. Communication Technologies: Communication technologies can be used to receive and transmit acquired information about the state of SG's devices. We have short-range and long-range communication technology standards. ZigBee, Bluetooth, and ultra-wideband technologies are examples of short-range communication technologies. For long-range communications, power line communications, optical fiber, wireless cellular networks such as 3G and 4G, and satellite communications can be used.

2. Data Fusion Techniques: Since the resources of IoT terminals (such as batteries, memory and bandwidth) are limited, it is not possible to send all information to the destination. Thus, to increase the efficiency of information gathering, data fusion techniques can be utilized to collect and combine data.

3. Energy Harvesting Process: Since most of the IoT devices use battery as one of their primary power sources, energy harvesting process is very important for IoT applications, e.g., using different sensors and cameras to monitor different parts of a smart grid.

4. Operating in Harsh Environments: IoT devices which are installed in high-voltage transmission lines and substations must work in harsh environments. Thus, to extend the lifetime of their sensors in these conditions, we should have sensors should be high or low temperature resistant, anti-electromagnetic, or waterproof.

5. Reliability: IoT applications in different environments need to satisfy different requirements such as reliability, self-organization, or self-healing. Thus, based on the actual environment, suitable IoT device must be selected to overwhelm environmental issues. For example, when some devices cannot send data due to lack of energy, a new route for the data must be found so that the network reliability remains at the required level.

6. Security: Security methods must be implemented in all IoT layers to transmit, store, and manage data, avoid information leakage and losses, and protect data.

7. Sensors: Sensors measure quantities such as current, voltage, frequency, temperature, power, light, and other signals and deliver the raw information for processing, transmitting, and analyzing. Recently, nanotechnology is used to provide high-performance material which covers different sensor applications and enhances the growth of sensor industry.

Q.21 How does IoT contribute to the retail industry?

Ans. Contribution of IoT to the Retail Industry: IoT has a ton of applications to offer in terms of improving customer experience and just about as many in retail management. Here are the main opportunities of the internet of things in retail. It contribute in following areas :

1. Customer Experience Personalization: Using the internet of things is a good way for a brand to foster a personal connection between the brand and its customers. For instance, you can attract passersby to visit your store by sending an IoT-enabled notification to their smartphones.

Retailers can use the technology to find out more about a customer in order to lay the groundwork for microtargeting. This way, marketing managers will be able to make more conscious choices and use advertising budgets more efficiently.

2. Supply Chain Optimization: GPS and RFID technology will allow brands to track each individual item through the

light can be reprogrammed to respond to the desires of people and situations throughout the day. Diverse digital communication interfaces intended for smart lighting are the digital addressable lighting interface (DALI), Ethernet, Wi-Fi, ZigBee light link or bluetooth for the programming of predefined areas and spaces. In these systems, generally the areas are segmented depending on the people or events that may occur. This allows the systems to calculate the level of light needed, so that it can accurately calculate the levels of illuminance suitable for different tasks of the users with the advantage to calculate the power consumption in real-time.

Smart lighting systems organized as lighting networks often allow different types of lights to interact with each other, so that they can be synchronized. It is also possible to individually control a light fixture through the network by means of a remote controller, for example, with an application from a graphical interface of a mobile phone or a web browser.

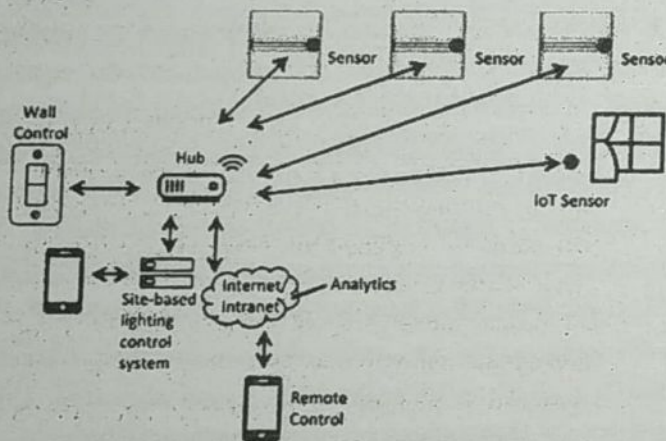


Fig. : Pictorial view of smart/ IoT -based lighting system

Q.15 Write a short note on smart appliances.

Ans. Smart Appliances: Modern homes have a number of appliances such as TVs, refrigerators, music systems, washer/dryers, etc. Managing and controlling these appliances can be cumbersome, with each appliance having its own controls or remote controls. Smart appliances make the management easier and also provide status information to the users remotely. For example, smart washer/dryers that can be controlled remotely and notify when the washing/drying cycle is complete. Smart thermostats allow controlling the temperature remotely and can learn the user preferences. Smart refrigerators can keep track of the items stored (using RFID tags) and send updates to the users when an item is

low on stock. Smart TVs allows users to search and stream videos and movies from the internet on a local storage drive. Search TV channel schedules and fetch news, weather updates and other content from the internet. Open remote is an open source automation platform for homes and buildings. Open remote is platform agnostic and works with standard hardware. With Open remote, users can control various appliances using mobile or web applications. Open remote comprises of three components – a controller that manages scheduling and runtime integration between devices, a designer that allows you to create both configurations for the controller and create user interface designs and control panels that allow you to interact with devices and control them. An IoT – based appliance control system for smart homes uses a smart central controller to set up a wireless sensor and actuator network and control modules for appliances.

Q.16 Write short on following:

- Intrusion detection
- Smoke detector

Ans.(a) Intrusion Detection : Home intrusion detection systems use security cameras and sensors (such as PIR sensors and door sensors) to detect intrusions and raise alerts. Alerts can be in the form of an SMS or an email sent to the user. Advanced systems can even send detailed alerts such as an image grab or a short video clip sent as an email attachment. A cloud controlled intrusion detection system uses location-aware services, where the geo-location of each node of a home automation system is independently detected and stored in the cloud. In the event of intrusions, the cloud services alert the accurate neighbors (who are using the home automation system) or local police. In an intrusion detection system based on UPnP technology. The system uses image processing to recognize the intrusion and extract the intrusion subject and generate Universal-Plug-and-Play (UPnP-based) instant messaging for alerts.

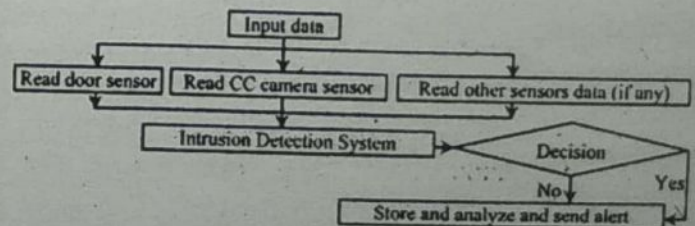


Fig. : IoT based home intrusion detection system

entire delivery process. You will be able to have a tight grip on your vendors as you will be able to monitor the delivery conditions and the location as well as predict a precise delivery time.

The range of applications of IoT in supply chain management is impressive. For instance, you can test different vendors, vehicles, and delivery routes collect the data on the process and find the cheapest framework that also transports the product with no damage.

The same stays true for product distribution and delivery to the end user.

3. Innovating In – store Experiences: Implementing the internet of things can help retailers to redesign their stores completely. You'll be able to provide a new experience for fitting rooms, create a system of intelligent suggestions, and go as far as to replace human workers with connected technology. Amazon Go is, without a doubt, the most famous and successful example of large-scale IoT implementation for revolutionizing in-store experience.

On a smaller scale, a store manager can replace a cash deck with a connected POS payment terminal.

By bringing IoT to physical stores, you'll be able to cut maintenance costs, increase the speed of service, and eliminate human error.

4. Increased Store Management Efficiency: The internet of things empowers a range of technologies that improve the efficiency of business operations in retail. These include:

- (a) Automated packaging services
- (b) SKU accounting
- (c) IoT drones for inventory monitoring

Q.22 What are the benefits of IoT in the retail industry?

Ans. After having conducted a thorough review of IoT opportunities in retail, let's go over the benefits of implementing the technology as a part of your own retail business. Reducing shrinkage and fraud as the internet of things adds an additional layer of traceability and visibility of the inventory and delivery process.

- (i) **Optimizing Product Placement:** IoT allows store managers to identify premium store areas, test the placement of different items in those spots, and find

the most efficient layout thanks to detailed reports based on the data gathered by sensors.

- (ii) **Efficient use of In – store Staff:** IoT can use cameras, sensors, and facial recognition algorithms in order to identify an impatient or confused shopper. Staff will be able to make proactive decisions and successfully engineer the atmosphere within the store.
- (iii) **Improved Retail Management and Tracking:** IoT helps store managers be aware of the number of products on the shelves and in the inventory, replenish stocks on time and more. The technology can also send automated reports that will later improve financial management and taxing.
- (iv) **Connecting Online and In-store Experiences:** The internet of things in the retail industry allows users to benefit from brand-related digital solutions while using physical stores. This way, retail companies can achieve synergy between online and in-store experiences.

Q.23 What do you mean by smart irrigation and green house control with reference of agriculture?

Ans. Smart Irrigation: Smart irrigation systems can improve crop yields while saving water. Smart irrigation systems use IoT devices with soil moisture sensors to determine the amount of moisture in the soil and release the flow of water through the irrigation pipes only when the moisture levels go below a predefined threshold. Smart irrigation systems also collect moisture level measurements on a server or in the cloud where the collected data can be analyzed to plan watering schedules. Cultivar's RainCloud is a device for smart irrigation that uses water valves, soil sensors and a WiFi enabled programmable computer.

Green House Control: Green houses are structures with glass or plastic roofs that provide a conducive environment for growth of plants. The climatological conditions inside a green house can be monitored and controlled to provide the best conditions for growth of plants. The temperature, humidity, soil moisture, light and carbon dioxide levels are monitored using sensors and the climatological conditions are controlled automatically using actuation devices (such as valves for releasing water and switches for controlling fans). IoT

systems play an important role in green house control and help in improving productivity. The data collected from various sensors is stored on centralized servers or in the cloud where analysis is performed to optimize the control strategies and also correlate the productivity with different control strategies. In, the design of a wireless sensing and control system for precision green house management.

The system uses wireless sensor network to monitor and control the agricultural parameters like temperature and humidity in real time for better management and maintenance of agricultural production.

Q.24 Explain the following terms:

- (a) Machine diagnosis and prognosis
- (b) Indoor air quality monitoring

Ans.(a) Machine Diagnosis and Prognosis: Machine prognosis refers to predicting the performance of a machine by analyzing the data on the current operating conditions and how much deviations exist from the normal operating conditions. Machine diagnosis refers to determining the cause of a machine fault. IoT plays a major role in both prognosis and diagnosis of industrial machines. Industrial machines have a large number of components that must function correctly for the machine to perform its operations. Sensors in machines can monitor the operating conditions such as (temperature and vibration levels). The sensor data measurements are done on timescales of few milliseconds to few seconds, which leads to generation of massive amount of data. IoT based systems integrated with cloud-based storage and analytics back-ends can help in storage, collection and analysis of such massive scale machine sensor data. A number of methods have been proposed for reliability analysis and fault prediction in machines. Case-based reasoning (CBR) is a commonly used method that finds solutions to new problems based on past experience. This past experience is organized and represented as cases in a case-base. CBR is an effective technique for problem solving in the fields in which it is hard to establish a quantitative mathematical model, such as machine diagnosis and prognosis. Since for each machine, data from a very large number of sensors is collected, using such high dimensional data for creation of case library reduces the case retrieval efficiency. Therefore, data reduction and feature extraction methods are used to find the representative set of

features which have the same classification ability as the complete of features.

(b) Indoor Air Quality Monitoring: Monitoring indoor air quality in factories is important for health and safety of the workers. Harmful and toxic gases such as carbon monoxide (CO), nitrogen monoxide (NO), Nitrogen Dioxide (NO₂), etc., can cause serious health problems. IoT based gas monitoring systems can help in monitoring the indoor air quality using various gas sensors. The indoor air quality can vary for different locations. Wireless sensor networks based IoT devices can identify the hazardous zones, so that corrective measures can be taken to ensure proper ventilation. In a hybrid sensor system for indoor air quality monitoring is presented, which contains both stationary sensors (for accurate readings and calibration) and mobile sensors (for coverage). In a wireless solution for indoor air quality monitoring is described that measures the environmental parameters like temperature, humidity, gaseous pollutants, aerosol and particulate matter to determine the indoor air quality.

Q.25 Explain the following term related to healthcare and lifestyle.

- (a) Health and fitness monitoring
- (b) Wearable electronics

Ans.(a) Health and Fitness Monitoring: Wearable IoT devices that allow non-invasive and continuous monitoring of physiological parameters can help in continuous health and fitness monitoring. These wearable devices may can be in various forms such as belts and wrist-bands. The wearable devices form a type of wireless sensor networks called body area networks in which the measurements from a number of wearable devices are continuous sent to a master node (such as a smart-phone) which then sends the data to a server or a cloud-based back-end for analysis and archiving. Health-care providers can analyze the collected health-care data to determine any health conditions or anomalies. Commonly uses body sensors include: body temperature, heart rate, pulse oximeter oxygen saturation (SpO₂), blood pressure, electrocardiogram (ECG), movement (with accelerometers), and electroencephalogram (EEG). An ubiquitous mobility approach for body sensor networks in health-care. In a wearable ubiquitous health-care monitoring system is presented that uses integrated electrocardiogram (ECG),

accelerometer and oxygen saturation (SpO2) sensors. Fitbit wristband is a wearable device that tracks steps, distance, and calories burned during the day and sleep quality at night.

(b) Wearable Electronics: Wearable electronics such as wearable gadgets (smart watches, smart glasses, wristbands, etc.) and fashion electronics (with electronics integrated in clothing and accessories, (e.g., Google Glass or Moto 360 smart watch) provide various functions and features to assist us in our daily activities and making us lead healthy lifestyles. Smart watches that run mobile operating systems (such as Android) provide enhanced functionality beyond just timekeeping. With smart watches, the users can search the Internet, play audio/video files, make calls (with or without paired mobile phones), play games and use various kinds of mobile applications. Smart glasses allows users to take photos and record videos, get map directions, check flight status, and search the internet by using voice commands. Smart shoes monitor the walking or running speeds and jumps with the help of embedded sensors and be paired with smart-phones to visualize the data. Smart wristbands can track the daily exercise and calories burnt.

PART-C

Q.26 Explain domain specific "cities" in detail.

Ans. Cities: Cities include following Parameters :

1. Smart Parking: Finding a parking space during rush hours in crowded cities can be time consuming and frustrating. Furthermore, drivers blindly searching for parking spaces create additional traffic congestion. Smart parking make the search for parking space easier and convenient for drivers. Smart parking are powered by IoT systems that detect the number of empty parking slots and send the information over the internet to smart parking application back-ends. These applications can be accessed by the drivers from smart-phones, tablets and in-car navigation systems. In smart parking, sensors are used for each parking slot, to detect whether the slot is empty or occupied. This information is aggregated by a local controller and then sent over the internet to the database. Polycarpou et. al. describe latest trends in parking availability monitoring, parking reservation and dynamic pricing schemes. Design and implementation of a

prototype smart parking system based on wireless sensor network technology with features like remote parking monitoring, automated guidance and parking reservation mechanism.

2. Smart Lighting: Smart lighting systems for roads, parks and buildings can help in saving energy. According to an IEA report lighting is responsible for 19% of global electricity use and around 6% of global greenhouse gas emissions. Smart lighting allows lighting to be dynamically controlled and also adaptive to the ambient conditions. Smart lights connected to the internet can be controlled remotely to configure lighting schedules and lighting intensity. Custom lighting configuration can be set for different situations such as a foggy day, festival, etc. Smart lights equipped with sensors can communicate with other lights and exchange information on the sensed ambient conditions to adapt the lighting. Castro et. al. describe the need for smart lighting system in smart cities, smart lighting features and how to develop interoperable smart lighting solutions.

3. Smart Roads: Smart roads equipped with sensors can provide information on driving conditions, travel time estimates and alerts in case of poor driving conditions, traffic congestions and accidents. Such information can help in making the roads safer and help in reducing traffic jams. Information sensed from the roads can be communicated via internet to cloud-based applications and social media and disseminated to the drivers who subscribe to such applications. In, a distributed and autonomous system of sensor network nodes for improving driving safety on public roads is proposed. The system can provide the drivers and passengers with a consistent view of the road situation a few hundred meters ahead of them or a few dozen miles away, so that they can react to potential dangers early enough.

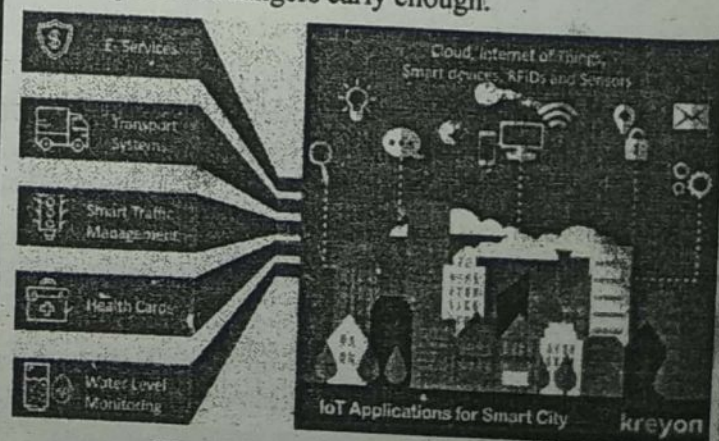


Fig. : IoT Applications for smart cities

4. Structural Health Monitoring: Structural health monitoring systems use a network of sensors to monitor the vibration levels in the structures such as bridges and buildings. The data collected from these sensors is analyzed to assess the health of the structures. By analyzing the data it is possible to detect cracks and mechanical breakdowns, locate the damages to a structure and also calculate the remaining life of the structure. Using such systems, advance warnings can be given in the case of imminent failure of the structure.

Since structural health monitoring systems use large number of wireless sensor nodes which are powered by traditional batteries, researchers are exploring energy harvesting technologies to harvesting ambient energy, such as mechanical vibrations, sunlight, and wind.

5. Surveillance: Surveillance of infrastructure, public transport and events in cities is required to ensure safety and security. City wide surveillance infrastructure comprising of large number of distributed and internet connected video surveillance cameras can be created. The video feeds from surveillance cameras can be aggregated in cloud-based scalable storage solutions.

Cloud-based video analytics applications can be developed to search for patterns or specific events from the video feeds. In a smart city surveillance system is described that leverages benefits of cloud data stores.

6. Emergency Response: IoT systems can be used for monitoring the critical infrastructure in cities such as buildings, gas and water pipelines, public transport and power substations, IoT systems for fire detection, gas and water leakage detection can help in generating alerts and minimizing their effects on the critical infrastructure. IoT systems for critical infrastructure monitoring enable aggregation and sharing of information collected from large number of sensors. Using cloud-based architectures, multi-modal information such as sensor data, audio, video feeds can be analyzed in near real-time to detect adverse events. Response to alerts generated by such systems can be in the form of alerts sent to the public, re-routing of traffic, evacuations of the affected areas, etc. Attwood et. al. describe critical infrastructure response framework for smart cities. A traffic management system for emergency services adapts by dynamically adjusting traffic lights, changing related driving policies, recommending behavior change to drivers, and applying essential security controls. Such systems can reduce the

latency of emergency services for vehicles such as ambulances and police cars while minimizing disruption of regular traffic.

Q.27 Explain domain specific "environment" in detail.

Ans. Environment: Environment is define as the surrounding of biotic and abiotic component and include the following parameters.

1. Weather Monitoring: IoT-based weather monitoring systems can collect data from a number of sensor attached (such as temperature, humidity, pressure, etc.) and send the data to cloud-based applications and storage back-ends. The data collected in the cloud can then be analyzed and visualized by cloud-based applications. Weather alerts can be sent to the subscribed users from such applications. Airpi is a weather and air quality monitoring kit capable of recording and uploading information about temperature, humidity, air pressure, light levels, UV levels, carbon monoxide, nitrogen dioxide and smoke level to the internet, a pervasive weather monitoring system is integrated with buses to measure weather variables like humidity, temperature and air quality during the bus path.

2. Air Pollution Monitoring: IoT based air pollution monitoring systems can monitor emission of harmful gases (CO_2 , CO, NO, NO_2 , etc.) by factories and automobiles using gaseous and meteorological sensors. The collected data can be analyzed to make informed decisions on pollutions control approaches. A real-time air quality monitoring system is presented that comprises of several distributed monitoring stations that communicate via wireless with a back-end server using machine-to-machine communication, an air pollution system integrates a single-chip microcontroller, several air pollution sensors, GPRS – Modem, and a GFS module.

3. Noise Pollution Monitoring: Due to growing urban development, noise levels in cities have increased and even become alarmingly high in some cities. Noise pollution can cause health hazards for humans due to sleep disruption and stress. Noise pollution monitoring can help in generating noise maps for cities. Urban noise maps can help the policy makers in urban planning and making policies to control noise levels near residential areas, schools and parks. IoT based noise pollution monitoring systems use a number of noise monitoring stations that are deployed at different places in a city. The

data on noise levels from the stations is collected on servers or in the cloud. The collected data is then aggregated to generate noise maps. In, a noise mapping study for a city revealed that the city suffered from serious noise pollution. In the design of smart phone application users to continuously measure noise levels and send to a central server where all generated information is aggregated and mapped to a meaningful noise visualization map.

4. Forest Fire Detection: Forest fires can cause damage to natural resources, property and human life. There can be different causes of forest fires including lightening, human negligence, volcanic eruptions and sparks from rock falls. Early detection of forest fires can help in minimizing the damage. IoT based forest fire detection systems use a number of monitoring nodes deployed at different locations in a forest. Each monitoring node collects measurements on ambient conditions including temperature, humidity, light levels, etc. A system for early detection of forest fires provides early warning of a potential forest fire and estimates the scale and intensity of the fire if it materializes. The system uses multi-criteria detection which is implemented by the artificial neural network (ANN). The ANN fuses sensing data corresponding to multiple attributes of a forest fire (such as temperature, humidity, infrared and visible light) to detect forest fires.

5. River Floods Detection: River floods can cause extensive damage to the natural and human resources and human life. River floods occur due to continuous rainfall which cause the river levels to rise and flow rates to increase rapidly. Early warnings of floods can be given by monitoring the water level and flow rate. IoT based river flood monitoring system use a number of sensor nodes that monitor the water level (using ultrasonic sensors) and flow rate (using the flow velocity sensors). Data from a number of such sensor nodes is aggregated in a server or in the cloud. Monitoring applications raise alerts when rapid increase in water level and flow rate is detected a river flood monitoring system measures river and weather conditions through wireless sensor nodes equipped with different sensors. In a motes-based sensor network for river flood monitoring, the system includes a water level monitoring module, network video recorder module, and data processing module that provides flood information in the form of raw data, predicted data, and video feed.

Q.28 Explain environment monitoring applications in detail.

Ans. Environment Monitoring Applications : The environment monitoring applications are as followings :

(a) Waste Management: The problem of waste management is very crucial issue in big cities, due to two reasons; first the cost of service and second the problem of storage of accumulating garbage. In order to save and make use of inexpensive environmental advantages, a deeper penetration of information and communications technologies solutions in this field will be required. For example, intelligent waste containers help identify the level of load the trucks carry and allow for an optimization of the collector trucks route, which in turn can reduce the cost of waste collection and improve the quality of recycling. To incorporate and make effective use of such smart waste management services, the IoT will connect these intelligent waste containers, to a control center where an optimization software will process the data and determine the optimal management and route the collector truck should follow.

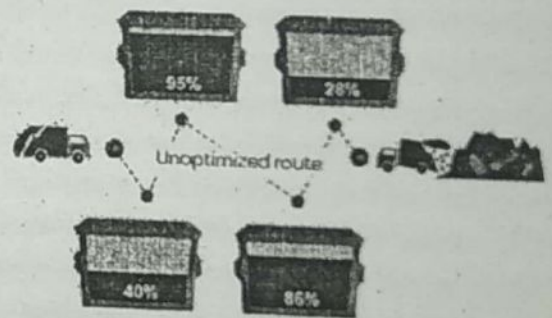


Fig. : IoT application waste management

(b) Vehicle Tracking:



Fig. : Internet of things (IoT) environmental monitoring-vehicle tracking

The vehicle tracking facility makes use of road sensors and intelligent display systems that help drivers to find the best path for parking in the city. The benefits from this service are many such as faster the car takes to locate a parking slot means lesser CO emission from the car, lesser traffic problems, and ultimately happier citizens. The IoT

combination with bluetooth, where NFC (which offers low speeds) initiates initial pairing of devices to establish a bluetooth connection while the actual data transfer takes place over bluetooth.

3. Smart Vending Machines: Smart vending machines connected to the internet allow remote monitoring of inventory levels, elastic pricing of products, promotions, and contactless payments using NFC, smartphone applications that communicate with smart vending machines allow user preferences to be remembered and learned with time. When a user moves from one vending machine to the other and pairs the smartphone with the vending machine, a user specific interface is presented. Users can save their preferences and favorite products, sensors in a smart vending machine monitor its operations and send the data to the cloud which can be used for predictive maintenance. Smart vending machines can communicate with other vending machines in their vicinity and share their inventory levels so that the customers can be routed to the nearest machine in case a product goes out of stock in a machine. For perishable items, the smart vending machines can reduce the price as the expiry date nears. New products can be recommended to the customers based on the purchase history and preferences.

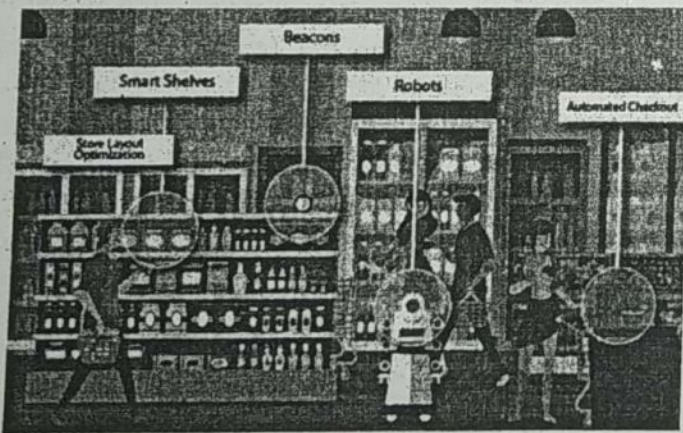


Fig. : Smart Vending Machine

Q.30 Write a detail note on applications of IoT for logistics.

Ans. Applications of IoT for Logistics:

1. Route Generation and Scheduling: Modern transportation systems are driven by data collected from multiple sources which is processed to provide new services to the stakeholders. By collecting large amount of data from

various sources and processing the data into useful information, data-driven transportation systems can provide new services such as advanced route guidance, dynamic vehicle routing, anticipating customer demands for pickup and delivery problem for instance. Route generation and scheduling systems can generate end-to-end routes using a combination of route patterns and transportation modes and feasible schedules based on the availability of vehicles. As the transportation network grows in size and complexity, the number of possible route combinations increase exponentially. IoT based systems backed by the cloud can provide fast response to the route generation queries and can be scaled up to serve a large transportation network.

2. Fleet Tracking: Vehicle fleet tracking systems use GPS technology to track the locations of the vehicles in real-time. Cloud-based fleet tracking systems can be scaled up on demand to handle large number of vehicles. Alerts can be generated in case of deviations in planned routes. The vehicle locations and routes data can be aggregated and analyzed for detecting bottlenecks in the supply chain such as traffic congestions on routes, assignments and generation of alternative routes, and supply chain optimization. In a fleet tracking system for commercial vehicles the system can analyze messages sent from the vehicles to identify unexpected incidents and discrepancies between actual and planned data, so that remedial actions can be taken.

3. Shipment Monitoring: Shipment monitoring solutions for transportation systems allow monitoring the conditions inside containers. For example, containers carrying fresh food produce can be monitored to prevent spoilage of food. IoT based shipment monitoring systems use sensors such as temperature, pressure, humidity, for instance, to monitor the conditions inside the containers and send the data to the cloud, where it can be analyzed to detect food spoilage. The analysis and interpretation of data on the environmental conditions inside the container and food truck positioning can enable more effective routing decisions in real time. Therefore, it is possible to take remedial measures such as - the food that has a limited time budget before it rotten can be re-routed to a closer destinations, alerts can be raised to the driver and the distributor about the transit conditions, such as container temperature exceeding the allowed limit, humidity levels going out of the allowed limit, for instance, and corrective actions can be taken before the food gets damaged. For fragile

infrastructure can directly integrate the vehicle parking facility. Furthermore, like we discussed earlier, by using communication technologies, such as Near Field Communication (NFC) or Radio Frequency Identifiers (RFID), we can understand the electronic confirmation system of parking and locate slots reserved for residents or disabled persons, thus offering a better service to residents that can make use of those slots and also as an efficient tool to spot any violations quickly.



Fig. : Internet of things (IoT) environmental monitoring – vehicle tracking

The monitoring technology currently in use for air and water safety mainly uses manual labor along with some advanced instruments, and lab processing techniques. Through IoT systems, the need for manual labor is reduced. As a result, frequent sampling is allowed, increasing the range of monitoring and sampling, allowing sophisticated on-site testing, and providing responses to detection systems. This prevents any further contamination of water bodies and other natural resources and related disasters.

(c) **Extreme Weather:** Powerful, advanced systems currently used for weather forecasting allow deep monitoring, but they suffer from using broad instruments, such as radar and satellites. These instruments that are used for small details lack the accurate targeting potential for smart technology.

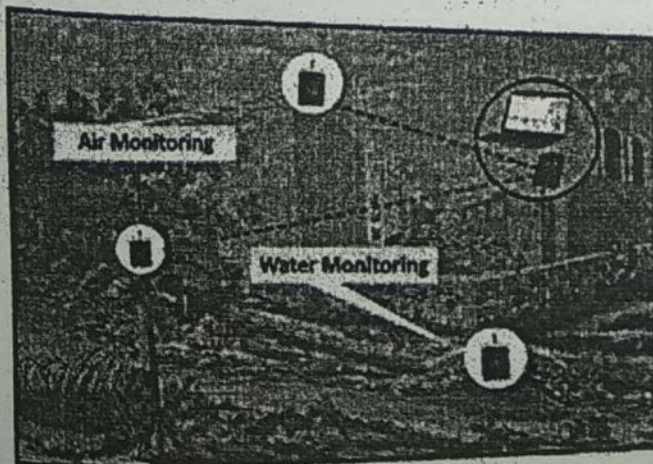


Fig. : IoT application – air and water pollution

Now, through the new IoT advances, the IoT system promises more data that fine-grain, better flexibility, and accuracy. Effective weather forecasting procedures require high detail as well as flexibility in instrument type, range, and deployment. This results in early responses to prevent loss of life and property through early detection.

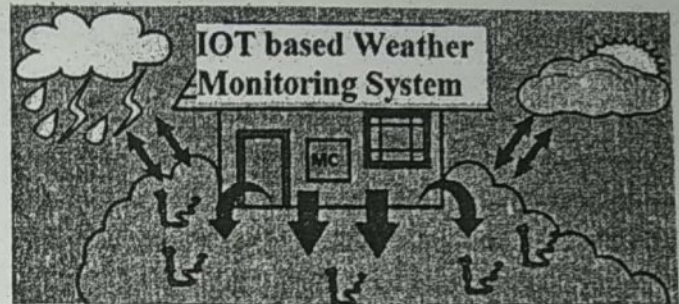


Fig. : IoT environmental monitoring benefits—extreme weather

So, this was all about IoT environmental monitoring application.

Q.29 Write the applications of IoT for retail.

Ans. Applications of IoT for Retail:

1. Inventory Management: Inventory management for retail has become increasingly important in the recent years with the growing competition. While over – stocking of products can result in additional storage expenses and risk (in case of perishables), under-stocking can lead to loss of revenue. IoT systems using Radio Frequency Identification (RFID) tags can help in inventory management and maintaining the right inventory levels. RFID tags attached to the products allow them to be tracked in real-time so that the inventory levels can be determined accurately and products which are low on stock can be replenished. Tracking can be done using RFID readers attached to the retail store shelves or in the warehouse. IoT systems enable remote monitoring of inventory using the data collected by the RFID readers.

2. Smart Payments: Smart payment solutions such as contact-less payments powered by technologies such as Near field communication (NFC) and bluetooth, Near Field Communication (NFC) is a set of standards for smartphones and other devices to communicate with each other by bringing them into proximity or by touching them. Customers can store the credit card information in their NFC – enabled smartphones and make payments by bringing the smartphones near the point of sale terminals. NFC maybe used in

systems to track progress but IoT takes a step further and provides intricacies to even the minute problems.

a. Digital/Connected Factory: The machinery that is embedded with an IoT system can transfer information related to operations to the people such as the original equipment manufacturers and to field engineers. This way process automation and optimization is made advantageous by enabling operation managers and factory heads to remotely manage the factory units. Along with this, a unit which is digitally connected helps in establishing a better line of command and also helps to identify areas with key results and areas that might have potential problems for managers.

b. Facility Management: The IoT sensors placed inside manufacturing equipment triggers alerts based on condition-based maintenance. Most of the machine tools are critical and are designed to function between a specific temperature and vibration ranges. Whenever an equipment deviates from its prescribed parameters, IoT sensors can actively monitor machines and send an alert. Manufacturers in this way can conserve energy, reduce costs, eliminate machine downtime and increase operational efficiency, by ensuring the prescribed working environment for machinery.

c. Production Flow Monitoring: IoT in manufacturing is capable of monitoring an entire production line be it from the refining process completely down to the packaging of final products. Because this complete monitoring of the process takes place in real-time. It provides us the scope to recommend any adjustments in operations for better management of the industry's operational cost. Since the monitoring is done quite closely, it lags in the actual production thereby eliminating wastes and unnecessary work.

d. Inventory Management: This is best industrial IoT application, through IoT systems monitoring of events across a supply chain is done. These systems allow one to track the inventory and trace it globally on a line – item level. This way the users are notified if there are any significant deviations from the plan of action. As a result, this provides a far-reaching and cross-channel visibility into inventories which helps managers in getting realistic estimates of the available material, the work in progress and the estimated arrival time of new materials. Ultimately this makes supply more optimal

and reduces additional and shared costs that arise in the value chain.

e. Plant Safety and Security: A workers' safety and security in the plant improve by IoT combined with big data analysis. The IoT system monitors some key performance indicators (KPIs) of health and safety, such as the number of injuries, frequent rates of illness, vehicle incidents, and property damage or any kind of loss incurred during daily operations. Thus, an effective monitoring system ensures better and effective safety. If there are some indicators that are lagging they addressed, thus ensuring better health, safety, and environment (HSE) issues. That's no one can ignore industrial IoT applications.

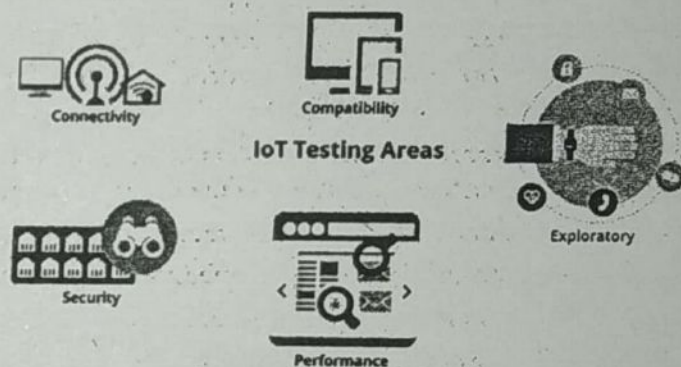


Fig. 1 : Applications—plant safety and security

f. Quality Control: A product cycle has various stages, IoT sensors collect a mixture of product data and other third-party synchronized data from the stages of a product cycle. This data contains information on the composition of raw materials used in the making of a product, the temperature and working environment, different wastes, the importance of transportation etc. On the final making of the products. Moreover, the IoT device can also provide data about the customer sentiments while he/she uses the product. All of these inputs from different sources and through IoT systems can analyze to identify and correct potential quality issues.

g. Packaging Optimization: Manufacturers can gain insights into the usage patterns and handling of product from different customers by using IoT sensors embedded in products and/or packaging. There are smart tracking mechanisms that can trace product deterioration during the product transit. Other factors such as weather impact, a condition of roads and other environment variables on the

products, vibration levels during shipments can be tracked using accelerometer and gyroscope sensors attached to IoT devices. In a system for monitoring container integrity and operating conditions, the system monitors the vibration patterns of a container and its contents to reveal information related to its operating environment and integrity during transport, handling and storage.

4. Remote Vehicle Diagnostics: Remote vehicle diagnostic systems can detect faults in the vehicles or warn of impending faults. These diagnostic systems use on-board IoT devices for collecting data on vehicle operation (such as speed, engine RPM, coolant temperature, fault code number) and status of various vehicle sub-systems. Such data can be captured by integrating on-board diagnostic systems with IoT devices using protocols such as CAN bus. Modern commercial vehicles support on-board diagnostic (OBD) standards such as OBD-II. OBD systems provide real-time data on the status of vehicle sub-systems and diagnostic trouble codes which allow rapidly identifying the faults in the vehicle. IoT based vehicle diagnostic systems can send the vehicle data to centralized servers or the cloud where it can be analyzed to generate alerts and suggest remedial actions. In a real-time online vehicle diagnostics and early fault estimation system makes use of on-board vehicle diagnostics device and expert system to achieve real-time vehicle diagnostics and fault warning.

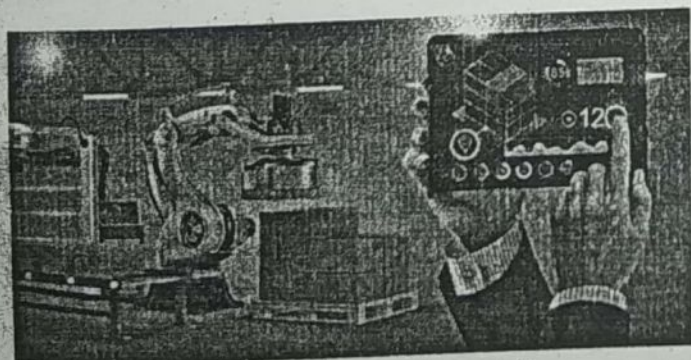


Fig. : Remote vehicle diagnostics

Q.31 What is IoT in logistics? What are the effects of IoT in the field of logistics?

Ans. Internet of Things in Logistics: Contemporary logistics is based on information and communication technologies that support realization of business processes and enable connection of users in supply chains. It is of great

significance in logistics to provide identification of objects and communication among participants. Identification technologies applied to various objects in logistics have led to the existence of smart containers, pallets, packaging, packing materials, vehicles, shelves, forklifts, infrastructure, ports, terminals and others. RFID, GPS and WSN systems have wide uses in logistics and supply chains:

- (a) **RFID System :** RFID systems enable automatic identification of objects and wireless data reading. RFID tags can be active, passive and semi-passive and contain a large quantity of data on objects on which they are placed. RFID tags are used for marking traffic, transport and reloading means, logistic units, individual articles and shelves in retail, special types of goods (money, gold, medicine, dangerous goods), post office packages, location and traffic in warehouses, identification cards, documents and others.
- (b) **GPS System :** GPS systems enable positioning of objects in real time. GPS devices receive satellite signals, determine their position in space and time, preserve data on location and transmit them to user of system. GPS is used in almost all segments of logistics, providing data on exact position and time where and when some object is found. GPS devices are placed on transport means of all types of traffic, semi-trailers, pallets, containers and individual goods, industrial and reloading mechanization, any devices that workers use in business processes.
- (c) **WSN System :** WSN enable collection and transmission of data between sensor nodes, access devices and network users. A sensor node consists of a set of active and passive sensors and can communicate, preserve and process collected data. Sensors are used for identification of objects and their physical characteristics – characteristics of goods, transport and reloading means, containers, locations in warehouses and sale facilities, equipment and traffic infrastructure, and others.

The application of these systems enabled the development of new business models and the concept of digital logistics in which a company automatically manages business processes and connects with its suppliers and buyers. In

logistic systems, there are various models of connection through the internet which relate to one or more companies or participants in supply chains. These models represent the initial IoT solutions which lead to global connection of all participants and objects. According to some research (Macaulay et al., 2015) 75% of companies used IoT solutions in 2014 in relation to 15% in 2012. The development of the IoT concept and universal communication network will enable a virtual model of business connecting whereby all participants will have data on objects available in real time. In literature, logistics is cited as the first field for application of IoT. The reason for this is that logistics depends on the quality of logistic network, connectivity of all participants in supply chain, fast and reliable information, everywhere and at all times. Logistic decisions are brought on the basis of available information and influence all other participants in supply chains. IoT connects identification technologies (RFID, GPS and WSN), built-in intelligence, advanced analysis of large quantities of data, software applications and systems of decision making at different control levels. The software systems in logistics (LIS, WMS, TMS, OMS, CRM, SCM) will realize maximum effects since high quality data and information on current state of objects on network will be used. Figure shows the application of the IoT concept in logistics. The significance of the IoT concept can be viewed at the level of logistic processes, participants in supply chains and on the global level. The greatest effects in the field of logistics are as followings:

- Monitoring transport and reloading means, logistic units, goods and people in real time.
- Measuring resource performances and planning in conformity with current state.
- Logistic controlling of activities and processes, reacting to deviation and disturbance conditions and applying corrective actions in order to realize the set goals.
- Analytics of all data and information in order to analyze the existing state and identify the possibilities for new business promotions.
- Automating of business processes by eliminating manual work along with improvement of quality and reduction of costs.
- Optimizing people, the system and means and their coordination and integration.

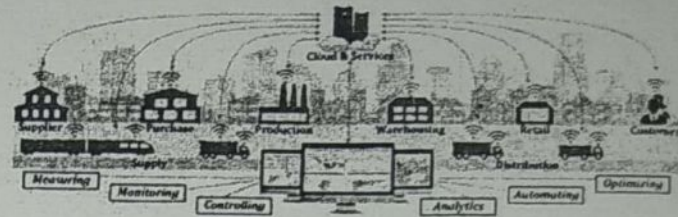


Fig. : IoT in logistics

Q.32 What are the applications of IOT for agriculture?

Ans. Applications of IoT in Agriculture:

(a) Precision Farming: Also known as precision agriculture, precision farming can be thought of as anything that makes farming practice more controlled and accurate when it comes to raising livestock and growing crops. In this approach of farm management, a key component is the use of IT and various items like sensors, control systems, robotics, autonomous vehicles, automated hardware, variable rate technology, and so on.

The adoption of access to high-speed internet, mobile devices, and reliable, low-cost satellites (for imagery and positioning) by the manufacturer are a few key technologies characterizing the precision agriculture trend.

Precision agriculture is one of the most famous applications of IoT in the agricultural sector and numerous organizations are leveraging this technique around the world. Crop Metrics is a precision agriculture organization focused on ultra-modern agronomic solutions while specializing in the management of precision irrigation.

The products and services of crop metrics include VRI optimization, soil moisture probes, virtual optimizer PRO, and so on. VRI (Variable Rate Irrigation) optimization maximizes profitability on irrigated crop fields with topography or soil variability, improve yields and increases water use efficiency.

The soil moisture probe technology provides complete in-season local agronomy support, and recommendations to optimize water use efficiency. The virtual optimizer PRO combines various technologies for water management into one central, cloud – based, and powerful location designed for consultants and growers to take advantage of the benefits in precision irrigation via a simplified interface.

(b) Agricultural Drones: Technology has changed over time and agricultural drones are a very good example of this. Today,

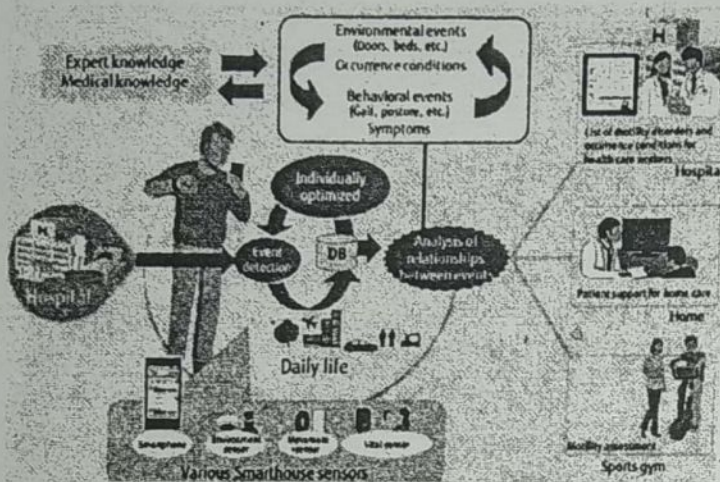


Fig. 3

(d) **Medical Information Distribution:** This is a most prominent innovation of IoT applications in healthcare, the distribution of accurate and current information to patients remains one of the most challenging concerns of medical care. IoT devices not only improve health in the daily lives of individuals but also facilities and professional practice.

IoT systems take healthcare out of facilities like hospitals and allow intrusive care into the office, home or social space. They empower and enable individuals to cater to their own health, and allow healthcare providers to deliver better care to patients. As a result, this has resulted and paved way for fewer accidents that usually result from miscommunication, improved patient satisfaction and better preventive care.

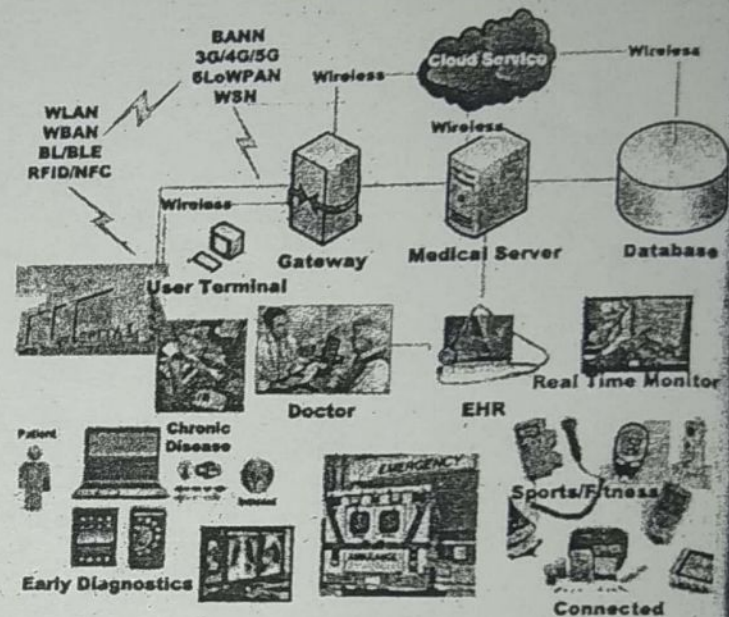


Fig. 4 : Internet of things in applications in healthcare-medical information distribution

(e) **Emergency Care :** The emergency support services have always had the problem of suffering from limited resources and getting disconnected with the base facility. The advanced automation and analytics of IoT cater to this problem in the healthcare sector. An emergency can be analyzed from a far distance or rather miles away. The providers get access to the patient profiles way before their arrival because of which they can deliver essential care to the patients on time. In this way, associated losses are reduced, and emergency health care is improved.

agriculture is one of the major industries to incorporate drones. Drones are being used in agriculture in order to enhance various agricultural practices. The ways ground – based and aerial-based drones are being used in agriculture are crop health assessment, irrigation, crop monitoring, crop spraying, planting, soil and field analysis.

The major benefits of using drones include crop health imaging, integrated GIS mapping, ease of use, saves time, and the potential to increase yields. With strategy and planning based on real-time data collection and processing, drone technology will give a high – tech makeover to the agriculture industry.

Precision Hawk is an organization that uses drones for gathering valuable data via a series of sensors that are used for imaging, mapping, and surveying of agricultural land. These drones perform in – flight monitoring and observations. The farmers enter the details of what field to survey and select an altitude or ground resolution.

From the drone data, we can draw insights regarding plant health indices, plant counting and yield prediction, plant height measurement, canopy cover mapping, field water ponding mapping, scouting reports, stockpile measuring, chlorophyll measurement, nitrogen content in wheat, drainage mapping, weed pressure mapping, and so on.

The drone collects multispectral, thermal, and visual imagery during the flight and then lands in the same location it took off.

(c) Livestock Monitoring: Large farm owners can utilize wireless IoT applications to collect data regarding the location, well – being and health of their cattle. This information helps them in identifying animals that are sick so they can be separated from the herd, thereby preventing the spread of disease. It also lowers labor costs as ranchers can locate their cattle with the help of IoT based sensors.

(d) Smart Greenhouses: Greenhouse farming is a methodology that helps in enhancing the yield of vegetables, fruits, crops, etc. Greenhouses control the environmental parameters through manual intervention or a proportional control mechanism. As manual intervention results in production loss, energy loss, and labor costs, these methods are less effective. A smart greenhouse can be designed with the help of IoT; this design intelligently monitors as well as controls the climate, eliminating the need for manual intervention.

For controlling the environment in a smart greenhouse, different sensors that measure the environmental parameters according to the plant requirement are used. We can create a cloud server for remotely accessing the system when it is connected using IoT. This eliminates the need for constant manual monitoring. Inside the greenhouse, the cloud server also enables data processing and applies a control action. This design provides cost-effective and optimal solutions for farmers with minimal manual intervention.

The IoT sensors in the greenhouse provide information on the light levels, pressure, humidity, and temperature. These sensors can control the actuators automatically to open a window, turn on lights, control a heater, turn on a mister or turn on a fan, all controlled through a WiFi signal.

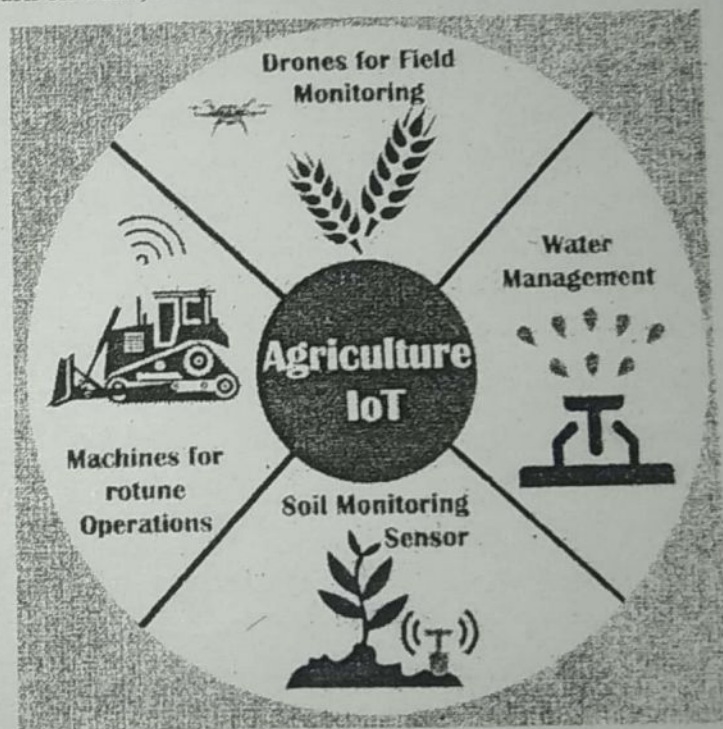


Fig. : Internet of things applications in agriculture

Q.33 Write the industrial application of IoT.

Ans. Industrial IoT Applications: Industrial internet of things (IIoT) is an ever growing and rapidly increasing sector that accounts for most of the share of IoT spending in the global market.

Industrialists and manufactures in almost every sector have a tremendous opportunity to not only monitor. But also automate many of complex process involved in manufacturing. For long time industries and plants have had sensors and

product. Through these insights, one can re-engineer products and their packaging for delivering better performance in both costs of packaging and customer experience.

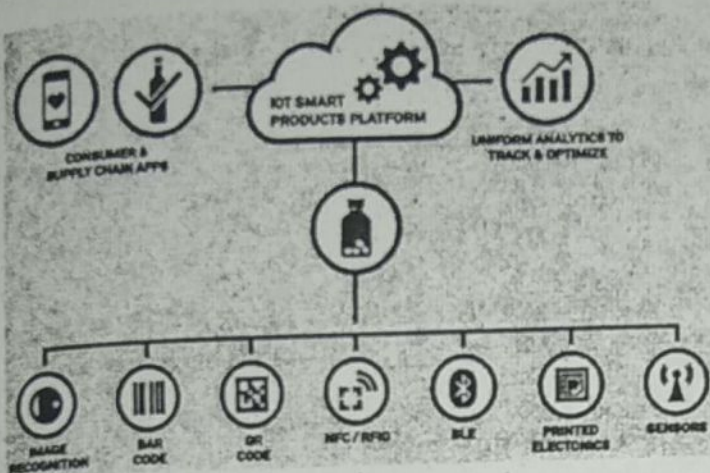


Fig. 2 : Packing optimization

Q.34 What are the various application of IoT for healthcare?

Ans. IoT Applications in Healthcare: The current technology in healthcare and a general practice of medicine gets enhanced with the IoT system. Professionals reach is expanding within a facility. The diverse data collected from large sets of real-world cases increases both the accuracy and size of medical data. The precision of medical care delivery is also improved by incorporating more sophisticated technologies in the healthcare system.

(a) Research: The resources that current medical research uses lack critical real-world information. It mostly uses leftovers, controlled environments and volunteers for medical examination. IoT opens ways to a sea of valuable data and information through analysis, real-time field data, and testing.



Fig. 1 : IoT Healthcare Applications - Research

IoT can deliver data that is far superior to standard analytics through making use of instruments that are capable of performing potential research. As a result, IoT helps in healthcare by providing more practical and reliable data, which yields better solutions and discovery of issues that were previously unknown, that's why research is one of the most important IoT applications in healthcare.

(b) Devices: Even current devices are improving in their power, precision, and availability; they still offer fewer benefits and qualities that an IoT system offers. IoT has the potential to unlock existing technology, and lead us towards better healthcare and medical device solutions.

IoT tries and fills gaps between the way we deliver healthcare and the equipment by creating a system rather than just tools. It then detects flaws and reveals patterns and missing elements in healthcare and suggests improvements.

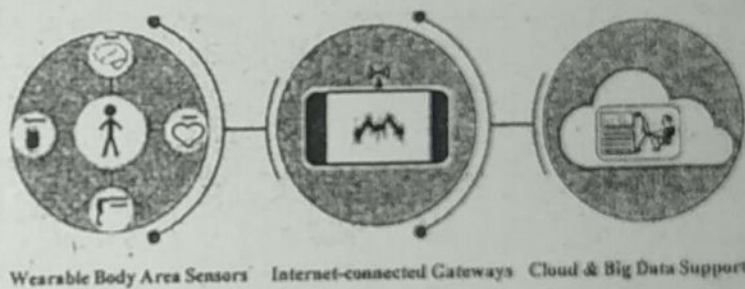


Fig. 2 : IoT in Healthcare - Devices

(c) Care: IoT empowers healthcare professionals to use their knowledge and training in a better way to solve problems. It helps them utilize better data and equipment that in turn supports more precise and swift actions. IoT allows in the professional development of healthcare professionals because they practically exercise their talent rather than spending time on administrative tasks.