CSE/IT

2025 Batch

5th Semester

IOT

Polytechnic

ONE SHOT

Syllabus

1. Introductionto Internet Of Things (IoT)
2. IoT Devices
3. IoT Networks
4. IoT Arduino
5. IoT and M2M

Chapter No: 1 Introduction to (IoT)

syllabus:

Introduction to IoT, Defining IoT, Things in IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, IoTProtocols, IoT communication Models, IoT communication API's, IoT enabling Technologies.

Introduction to (IoT):

- IoT का पूरा नाम Internet of Things है। इसका मतलब है — चीज़ों का ऐसा नेटवर्क जो इंटरनेट के ज़रिए एक-दूसरे से जुड़ा होता है और आपस में डेटा साझा करता है। इस नेटवर्क में सिर्फ कंप्यूटर या मोबाइल ही नहीं, बल्कि फ्रिज, एसी, लाइट, कार और अन्य स्मार्ट डिवाइस भी शामिल हो सकते हैं।
- The full form of IoT is Internet of Things. It means a network of things that are connected to each other via the Internet and share data among themselves. This network can include not only computers or mobiles, but also refrigerators, ACs, lights, cars and other smart devices.

[Physical Object] + [Controllers, Sensors, Actuators] + [Internet] = IoT

"Things" in IoT:

- Things" in IoT से तात्पर्य उन भौतिक वस्तुओं (physical objects) से है जो इंटरनेट से जुड़कर डेटा को इकट्ठा (collect), भेज (transmit), और प्राप्त (receive) कर सकती हैं। ये वस्तुएँ सेंसर, सॉफ़्टवेयर, नेटवर्क कनेक्टिविटी और अन्य तकनीकों से लैस होती हैं, जिससे ये स्मार्ट बन जाती हैं और मानव हस्तक्षेप के बिना काम कर सकती हैं।
- "Things" in IoT refers to physical objects that can collect, transmit, and receive data by connecting to the Internet. These objects are equipped with sensors, software, network connectivity, and other technologies, making them smart and able to function without human intervention.

उदाहरण (Examples of "Things" in IoT):

- **घरेलू उपकरण:** स्मार्ट लाइट, स्मार्ट एसी, स्मार्ट फ्रिज
- **स्वास्थ्य क्षेत्र:** फिटनेस बैंड, स्मार्ट वॉच, हेल्थ मॉनिटरिंग डिवाइस

Characteristics of IoT (IoT की प्रमुख विशेषताएँ):

1: कनेक्टिविटी (Connectivity):

- IoT डिवाइसेज़ आपस में और इंटरनेट से जुड़ी होती हैं। ये डेटा को भेजने और प्राप्त करने के लिए नेटवर्क (जैसे Wi-Fi, Bluetooth, 4G/5G) का उपयोग करती हैं।
- IoT devices are connected to each other and to the Internet. They use networks (such as Wi-Fi, Bluetooth, 4G/5G) to send and receive data.

Example:

- स्मार्ट बल्ब मोबाइल ऐप से जुड़कर ऑन/ऑफ हो सकता है।
- The smart bulb can be turned on/off by connecting to the mobile app.

2. सेंसरिंग (Sensing):

- IoT डिवाइसेज़ में सेंसर लगे होते हैं जो तापमान, गति, प्रकाश, नमी जैसी चीज़ों को पहचानते हैं और मापते हैं।
- IoT devices contain sensors that detect and measure things like temperature, motion, light, and humidity.

3. डेटा प्रोसेसिंग (Data Processing):

- सेंसर द्वारा इकट्ठा किया गया डेटा क्लाउड या डिवाइस में प्रोसेस (विश्लेषण) किया जाता है ताकि उचित निर्णय लिया जा सके।
- The data collected by the sensors is processed (analyzed) in the cloud or on the device to take appropriate decisions.

Example:

- फिटनेस बैंड आपके स्टेप्स और हार्ट रेट का डेटा रिकॉर्ड कर उसे ऐप में दिखाता है।
- The fitness band records your steps and heart rate data and displays it in the app.

4. स्वचालन (Automation):

- IoT डिवाइसेज़ इंसानों की मदद के बिना अपने-आप काम कर सकती हैं। इससे चीज़ें ज्यादा स्मार्ट और कुशल बनती हैं।
- IoT devices can work on their own without any human help. This makes things smarter and more efficient.

5. स्केलेबिलिटी (Scalability):

- IoT सिस्टम को बड़ी संख्या में डिवाइसेज़ के साथ आसानी से जोड़ा जा सकता है, जिससे यह छोटे से लेकर बड़े नेटवर्क तक काम कर सकता है।
- IoT systems can be easily connected to a large number of devices, allowing them to work in small to large networks.

6. सुरक्षा (Security):

- चूँकि IoT डिवाइस डेटा का आदान-प्रदान करती हैं, इसलिए इनकी सुरक्षा (encryption authentication आदि) बहुत ज़रूरी है ताकि डेटा चोरी या हैक न हो।
- Since IoT devices exchange data, their security (encryption, authentication, etc.) is very important so that the data is not stolen or hacked.

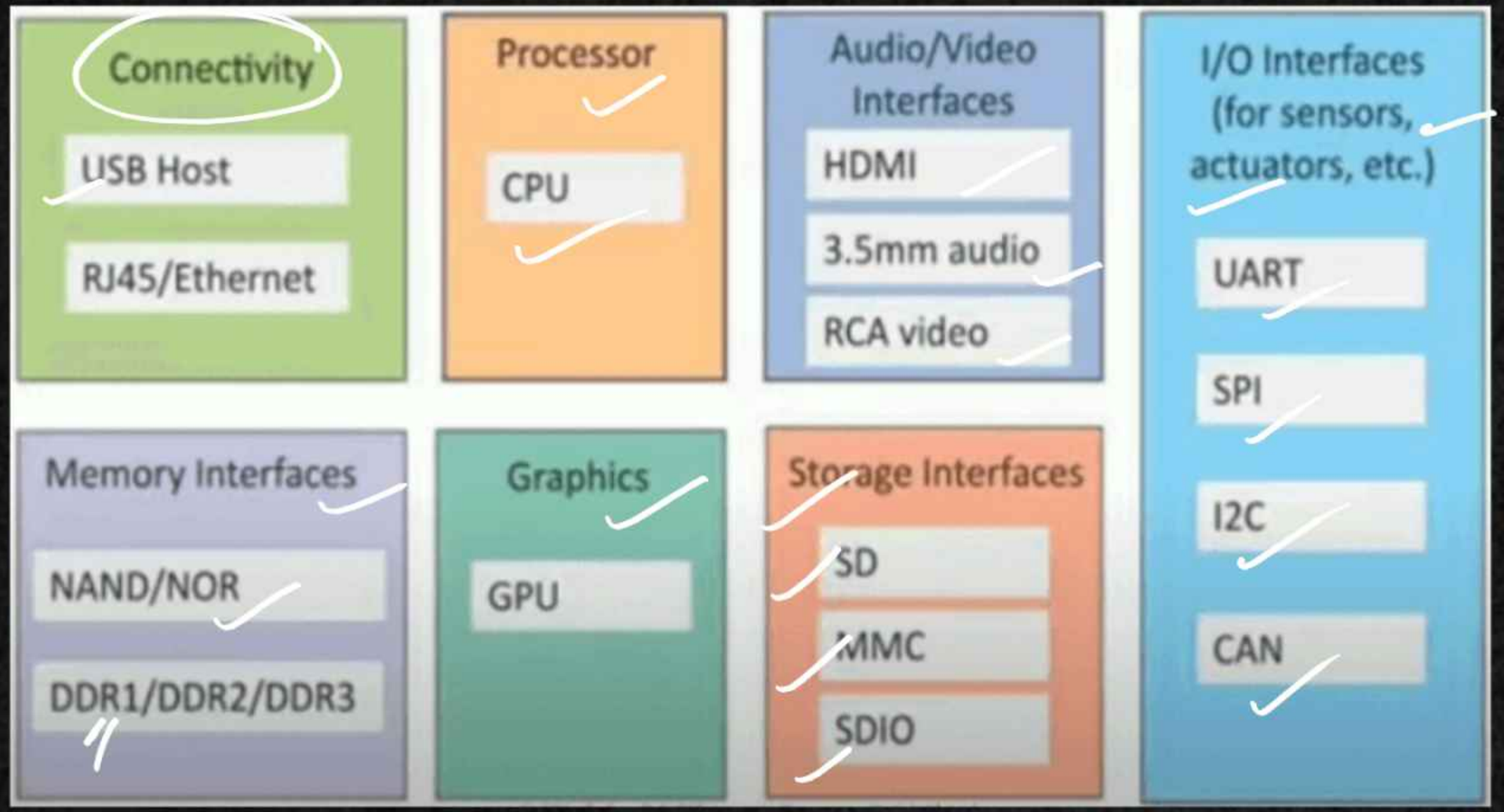
Advantages of IoT Devices (मुख्य लाभ):

- सुविधा और ऑटोमेशन
- रियल-टाइम मॉनिटरिंग
- ऊर्जा की बचत
- स्मार्ट निर्णय लेने की क्षमता
- दूर से नियंत्रण (Remote Control)
- बेहतर सुरक्षा और निगरानी
- समय और श्रम की बचत
- डेटा विश्लेषण और ट्रैकिंग आसान

- Convenience and automation
- Real-time monitoring
- Energy savings
- Smart decision making
- Remote control
- Improved security and monitoring
- Time and labor savings
- Easier data analysis and tracking

Physical Design of IOT:

- Physical Design of IoT उस सिस्टम की बनावट को दर्शाता है जिसमें IoT डिवाइस, सेंसर, एक्ट्यूएटर, नेटवर्क और अन्य हार्डवेयर मिलकर एक IoT सिस्टम को फिजिकली तैयार करते हैं।
- Physical Design of IoT refers to the system architecture in which IoT devices, sensors, actuators, networks, and other hardware combine to physically form an IoT system.


Connectivity
USB Host
RJ45/Ethernet
Processor
CPU
Audio/Video Interfaces
HDMI
3.5mm audio
RCA video
I/O Interfaces (for sensors, actuators, etc.)
UART
SPI
I2C
CAN
Memory Interfaces
NAND/NOR
DDR1/DDR2/DDR3
Graphics
GPU
Storage Interfaces
SD
MMC
SDIO

1. Connectivity (कनेक्टिविटी इंटरफेस):

- यह हिस्सा IoT डिवाइस को नेटवर्क या दूसरे डिवाइस से जोड़ने के लिए होता है।
- This part is used to connect the IoT device to the network or other devices.

USB Host:

- USB डिवाइसेज़ को कनेक्ट करने के लिए (जैसे पेन ड्राइव,).
- To connect USB devices (such as pen drives,).

RJ45 / Ethernet:

- वायर्ड इंटरनेट कनेक्शन के लिए
- For wired internet connection
- Industrial IoT में तेज और स्थिर नेटवर्क देने के लिए उपयोग होता है।
- Used in Industrial IoT to provide fast and stable networks.

2. Processor (प्रोसेसर यूनिट):

- IoT डिवाइस का मस्तिष्क (Brain) होता है।
- IoT is the brain of the device.

- CPU (Central Processing Unit):
  - सभी गणनाएँ (processing) करता है
  - Does all the calculations
  - सेंसर से आए डेटा को प्रोसेस करता है और एक्शन लेने का निर्णय करता है
  - Processes the data from the sensors and decides what action to take

3. Audio/Video Interfaces (ऑडियो/वीडियो इंटरफेस):

- यूजर को आवाज़ और वीडियो आउटपुट देने के लिए होते हैं।
- These are meant to provide sound and video output to the user.

- **HDMI:**
  - हाई-क्वालिटी वीडियो आउटपुट (High-quality video output).
  - स्क्रीन या मॉनिटर से कनेक्ट करने के लिए (For connecting to a screen or monitor)
- **3.5mm Audio:**
  - स्पीकर, माइक्रोफोन आदि के लिए (For speakers, microphones, etc.)
- **RCA Video:**
  - पुराने वीडियो डिवाइस के साथ कनेक्शन के लिए
  - For connection with older video devices

4. I/O Interfaces (Input/Output Interfaces):

- सेंसर, एक्ट्यूएटर और अन्य डिवाइसेज़ को कनेक्ट करने के लिए होते हैं।
- Used to connect sensors, actuators and other devices.

UART (Universal Asynchronous Receiver/Transmitter):

- Serial communication के लिए उपयोग होता है
- Used for serial communication
- GPS, GSM मॉड्यूल से कनेक्शन में उपयोगी
- Useful for connection to GPS, GSM module.

SPI (Serial Peripheral Interface):

- तेज़ communication के लिए (For faster communication)
- सेंसर या डिस्प्ले से जुड़ने के लिए (To connect to sensors or displays).

- **I2C (Inter-Integrated Circuit):**
  - Multi-device communication के लिए
  - For multi-device communication
  - कम पिनों से कई डिवाइस जोड़ने की सुविधा
  - Facility to connect many devices with fewer pins
- **CAN (Controller Area Network):**
  - Automotive applications जैसे गाड़ियों में उपयोग
  - Used in automotive applications such as vehicles
  - कई नोड्स के बीच तेज़ और सुरक्षित कम्युनिकेशन
  - Fast and secure communication between multiple nodes

5. Memory Interfaces (मेमोरी इंटरफेस):

- डेटा को अस्थायी या स्थायी रूप से स्टोर करने के लिए
- To store data temporarily or permanently

- **NAND/NOR:**
  - Flash memory
  - Boot code या firmware को स्टोर करने के लिए
- **DDR1/DDR2/DDR3:**
  - RAM memory (temporary memory)
  - Fast data access के लिए उपयोग होती है

6: Graphic interface:

- IoT डिवाइस में ग्राफिक्स डेटा को दिखाने और उपयोगकर्ता के साथ बातचीत को बेहतर बनाने में महत्वपूर्ण भूमिका निभाते हैं।
- In IoT devices, graphics play a crucial role in visualizing data and enhancing user interaction.

7: Storage interfac:

- IoT डिवाइस में मेमोरी इंटरफेस वह तरीका है जो डेटा को स्टोर करने और एक्सेस करने के लिए उपयोग किया जाता है। यह इंटरफेस डिवाइस के अंदर की मेमोरी को प्रोसेसर्स और अन्य कंपोनेंट्स के साथ जोड़ता है।
- In an IoT device, the memory interface is the method used to store and access data. This interface connects the device's internal memory with processors and other components.

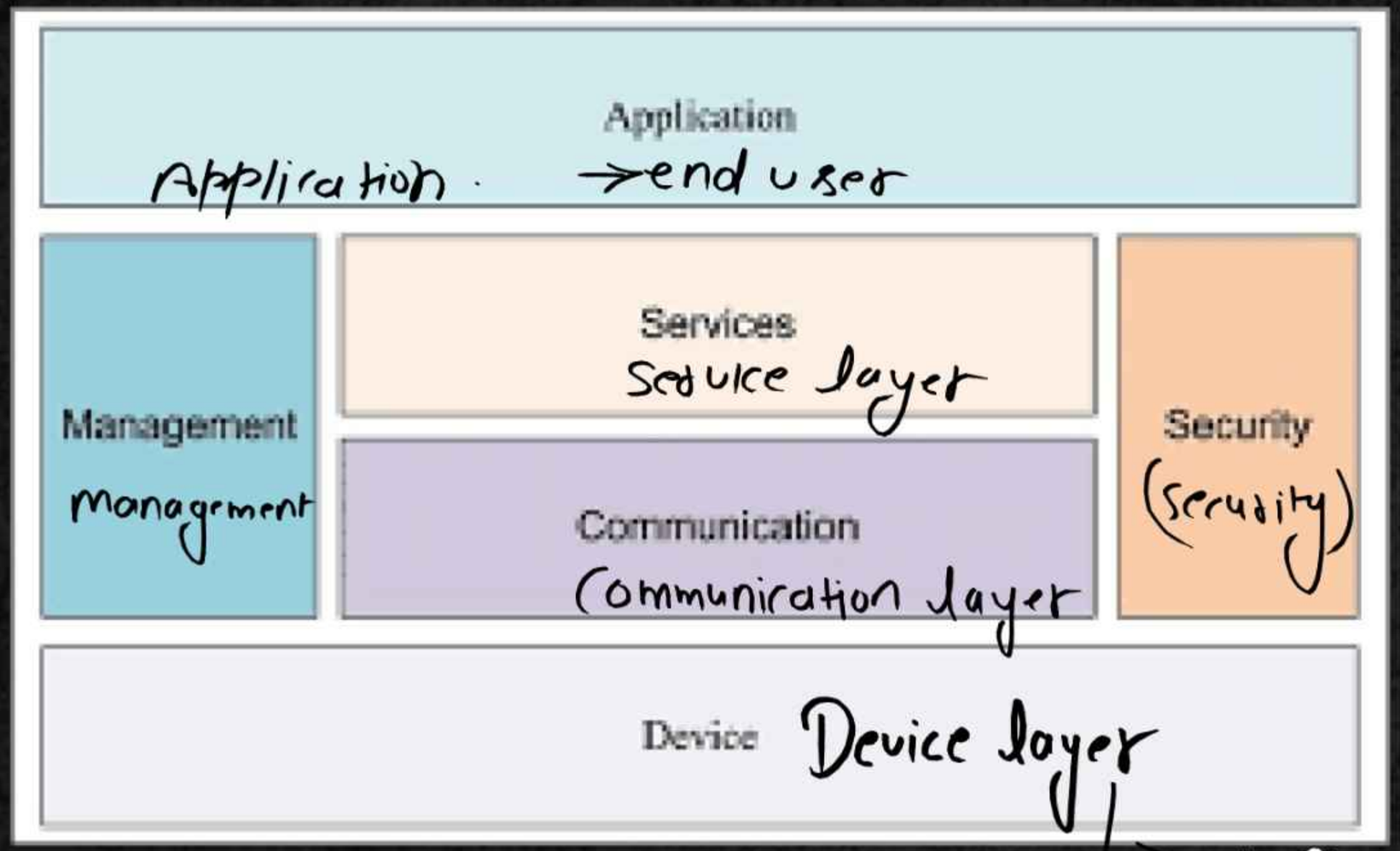
Logical Design Of IoT (IoT का तार्किक(logical) डिज़ाइन) :

- IoT (Internet of Things) का लॉजिकल डिज़ाइन एक ऐसा तरीका है जिसमें यह बताया जाता है कि IoT सिस्टम के अलग-अलग हिस्से (जैसे सेंसर, नेटवर्क, और डेटा प्रोसेसिंग) आपस में कैसे काम करते हैं। इसमें इस बात पर ध्यान दिया जाता है कि डेटा कैसे इकट्ठा होता है, कैसे भेजा जाता है, और कैसे प्रोसेस किया जाता है, बिना किसी हार्डवेयर या उपकरण के बारे में बात किए।
- The logical design of IoT (Internet of Things) refers to the abstract, conceptual framework that describes how the components of an IoT system interact and function. This design focuses on data flow, communication, and information processing without being concerned with the specific hardware or physical devices.

IoT Functional Block:

- IoT का Functional Block Diagram हमें यह समझने में मदद करता है कि IoT सिस्टम कैसे काम करता है, और हर कंपोनेंट (Block) की क्या भूमिका होती है
- The Functional Block Diagram of IoT helps us understand how the IoT system works, and what is the role of each component (block)

IoT Functional Block:



→ all Physical component.

1. Device Block (डिवाइस ब्लॉक)

- यह IoT सिस्टम की पहली और सबसे महत्वपूर्ण परत होती है। इसमें वे डिवाइस आते हैं जो हमारे आसपास के वातावरण से डेटा को इकट्ठा (collect) करते हैं या किसी कार्य को निष्पादित (actuate) करते हैं।
- This is the first and most important layer of the IoT system. It includes devices that collect data from the environment around us or actuate a task.

प्रमुख घटक:

- Sensors: तापमान, नमी, गति, प्रकाश आदि मापने के लिए।
- Actuators: कोई काम करने वाले डिवाइस (जैसे मोटर चलाना, बल्ब जलाना)।
- RFID, Barcode Readers, कैमरे आदि।

2. Communication Block (कम्युनिकेशन ब्लॉक:

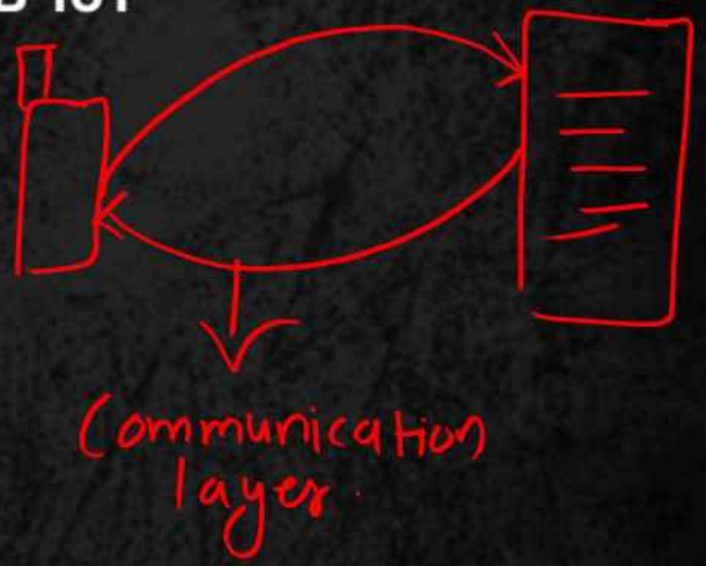
- यह ब्लॉक IoT डिवाइस और नेटवर्क/सर्वर के बीच डेटा को भेजने और प्राप्त करने का कार्य करता है। यह नेटवर्किंग की रीढ़ (backbone) होती है।
- This block works to send and receive data between the IoT device and the network/server. It is the backbone of networking.

प्रमुख तकनीकें:

- **Wired**: Ethernet, USB
- **Wireless:** WiFi, Bluetooth, ZigBee, LoRaWAN, 4G/5G, NB-IoT

उपयोग:

- Sensors का डेटा इंटरनेट तक पहुँचाना
- Transmitting sensor data to the Internet
- Cloud या Application को कम्युनिकेशन देना
- Providing communication to the cloud or application



3. Services Block (सर्विस ब्लॉक):

- यह ब्लॉक IoT सिस्टम को बुद्धिमत्ता (intelligence) देता है। यह डेटा को प्रोसेस, एनालाइज और स्टोर करता है, जिससे निर्णय लेना संभव हो सके।
- This block gives intelligence to the IoT system. It processes, analyzes and stores data, making decisions possible.

सेवाएँ:

- **Data Processing:** डेटा को समझदारी में बदलना
- **Data Storage:** क्लाउड पर डेटा स्टोर करना
- **Analytics:** विश्लेषण करना
- **AI/ML:** स्मार्ट निर्णय लेना

4. Management Block (मैनेजमेंट ब्लॉक):

- यह ब्लॉक पूरे IoT सिस्टम के संचालन (operation), निगरानी (monitoring), और नियंत्रण (control) का काम करता है।
- This block performs the operation, monitoring, and control of the entire IoT system.

कार्य:

- Device Registration और Configuration
- डिवाइस की health चेक करना
- सॉफ्टवेयर/फर्मवेयर अपडेट भेजना

5. Security Block (सिक्योरिटी ब्लॉक):

- यह ब्लॉक IoT सिस्टम को हैकिंग, अनाधिकृत एक्सेस और डेटा चोरी से बचाने का काम करता है।
- This block works to protect IoT systems from hacking, unauthorized access, and data theft.

मुख तकनीकें:

- Authentication (पासवर्ड, बायोमेट्रिक)
- Encryption (डेटा को कोड में बदलना)
- Access Control (किसे क्या एक्सेस मिलना चाहिए)

Vikas $\xrightarrow{\text{Enc}}$ [#@$]

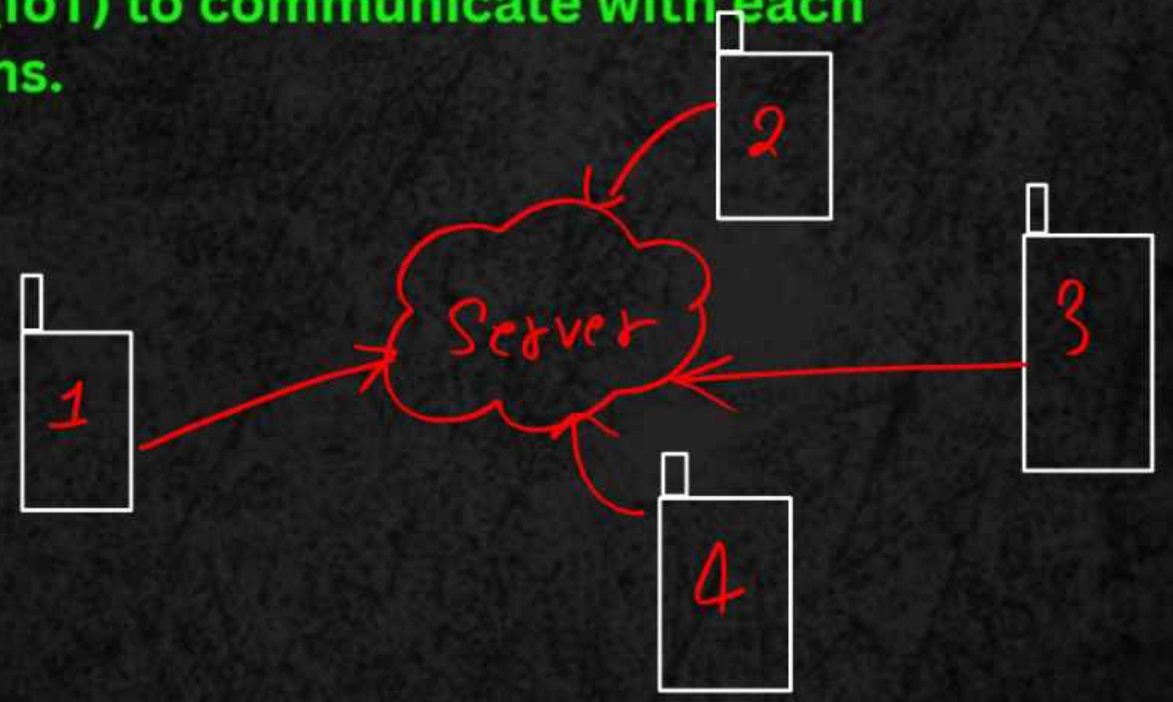
6. Application Block (एप्लिकेशन ब्लॉक):

- यह लेयर IoT सिस्टम का वह भाग है जिससे अंततः यूजर इंटरैक्ट करता है। यानी यह यूजर इंटरफेस प्रदान करती है।
- This layer is the part of the IoT system that the user ultimately interacts with. That is, it provides the user interface.

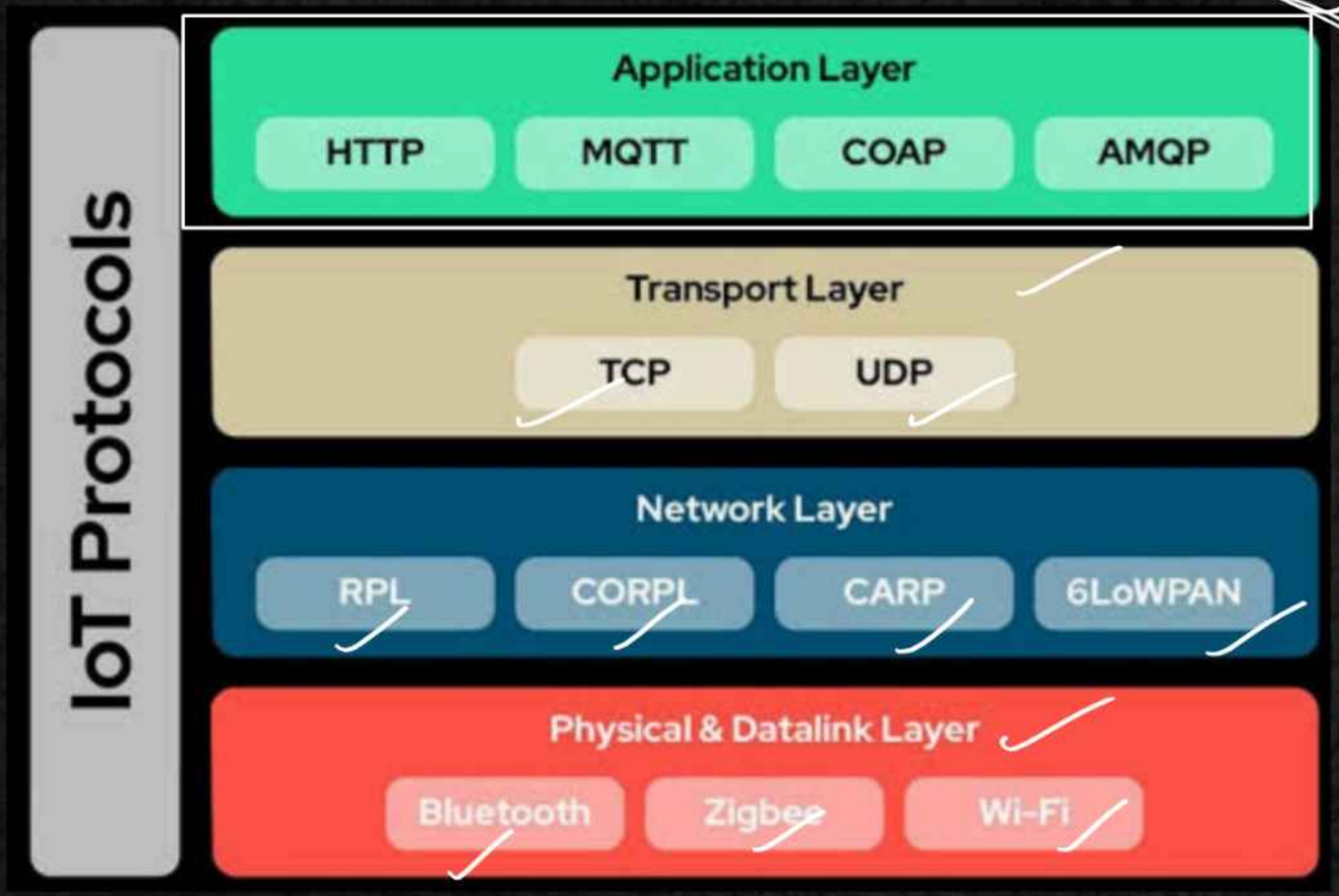
IoT Protocols for Internet Communication:

- IoT प्रोटोकॉल वे संचार मानक(communications standards)और नियम हैं जो इंटरनेट ऑफ थिंग्स (IoT) में उपकरणों (devices) को एक-दूसरे और केंद्रीय सिस्टम(centralized systems) के साथ बातचीत करने में मदद करते हैं।

- IoT protocols are communication standards and rules that enable devices in the Internet of Things (IoT) to communicate with each other and with centralized systems.



IoT Protocols for Internet Communication:



1: HTTP (Hypertext Transfer Protocol):

- डेटा संचार(communication) का आधार, जो फ़ाइलों और डेटा के आदान-प्रदान(transfer) की अनुमति देता है।
- The foundation of data communication on the web, enabling the transfer of files and data.

Use Case:

- IoT अनुप्रयोगों में सामान्यतः उपयोग किया जाता है जो वेब सर्वर या RESTful APIs के साथ बातचीत की आवश्यकता रखते हैं, जैसे क्लाउड-आधारित IoT सेवाएँ।
- Commonly used in IoT applications that require interaction with web servers or RESTful APIs, such as cloud-based IoT services.

2: CoAP (Constrained Application Protocol):

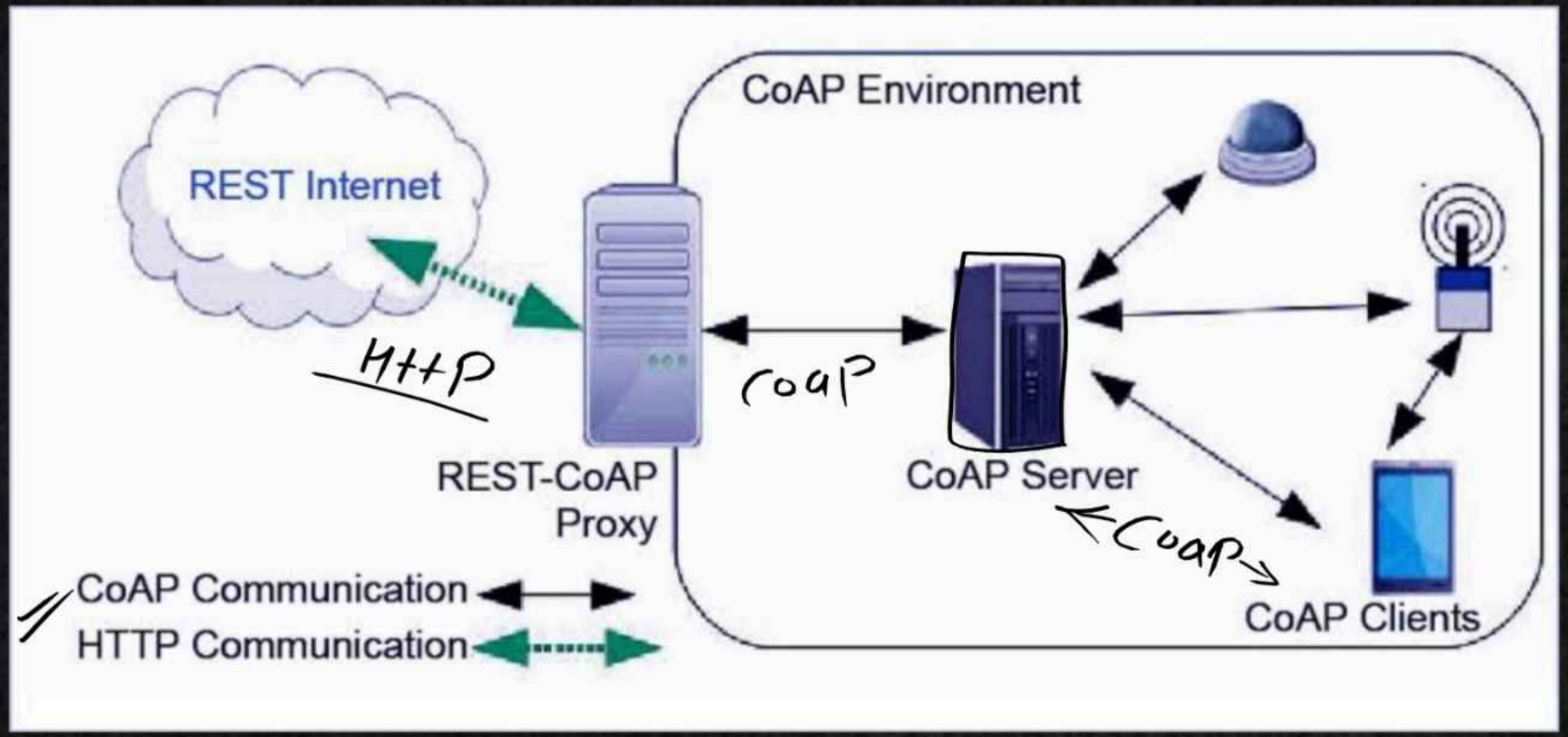
- यह प्रोटोकॉल छोटे और सीमित संसाधनों वाले डिवाइसेज़ के लिए डिज़ाइन किया गया है। इसे इंटरनेट पर काम करने के लिए बनाया गया है।
- This protocol is designed for devices with limited resources. It is made to work over the internet.

- एक स्मार्ट बल्ब जो इंटरनेट से जुड़ा है, CoAP प्रोटोकॉल का उपयोग कर सकता है। जब आप अपने फोन से बल्ब को ऑन या ऑफ करते हैं, तो यह कम बैंडविड्थ में कमांड को भेज सकता है।
- A smart bulb connected to the internet can use the CoAP protocol. When you turn the bulb on or off using your phone, it can send the command with minimal bandwidth usage.

HTTP
→ TCP (Protocol)
→ secure
→ slow

CoAP
⇒ UDP
⇒ unsecure
⇒ fast

2: CoAP (Constrained Application Protocol):



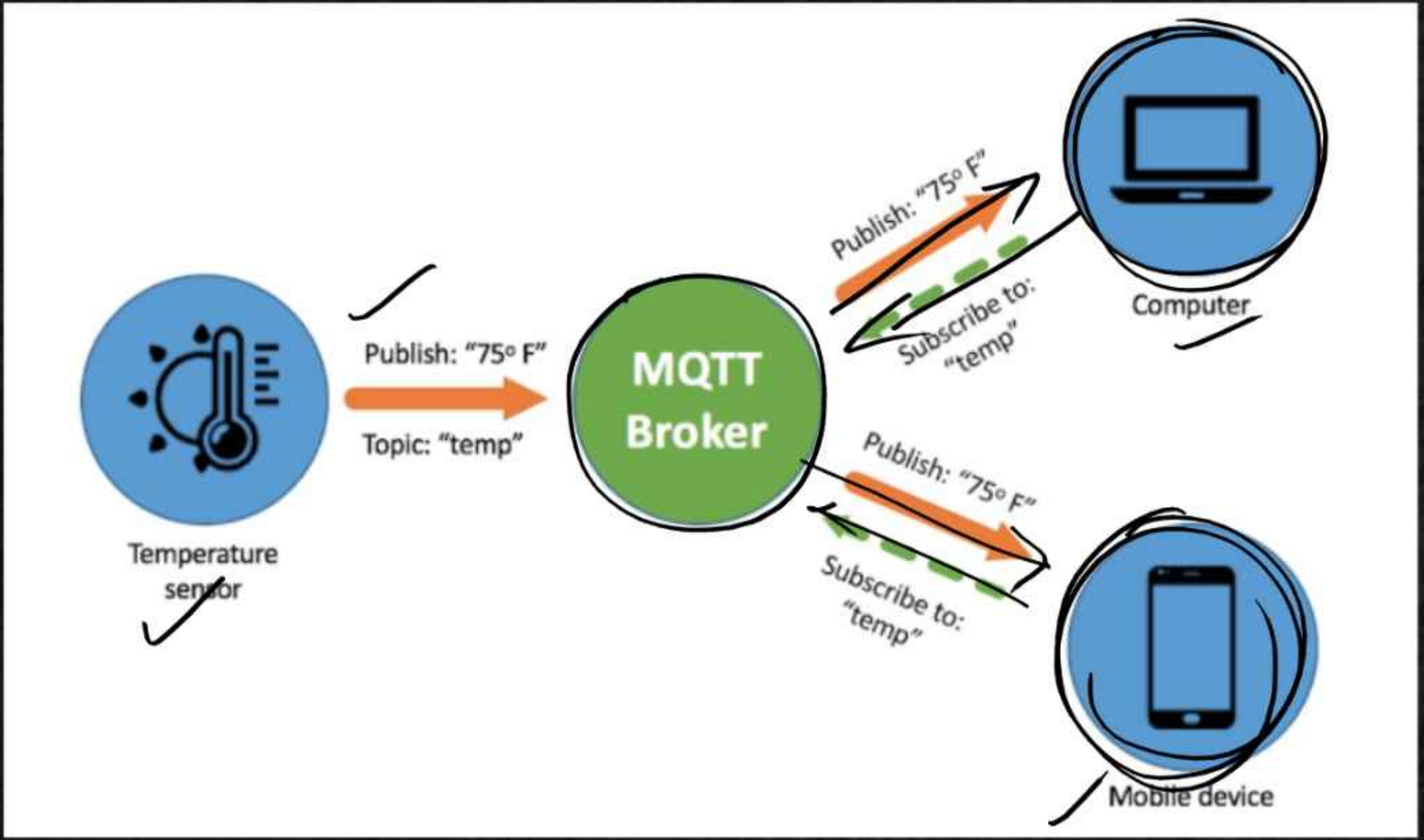
3: MQTT (Message Queuing Telemetry Transport):

- यह एक हल्का प्रोटोकॉल है जिसे उन डिवाइसों के लिए डिज़ाइन किया गया है जो कम मात्रा में डेटा भेजते हैं, जैसे सेंसर या स्मार्ट होम डिवाइस। यह कम बैंडविड्थ वाले नेटवर्क में अच्छी तरह से काम करता है।
- This is a lightweight protocol designed for devices that send small amounts of data, like sensors or smart home devices. It works well in low-bandwidth networks.

Example:

- कल्पना करें कि आपके पास एक स्मार्ट थर्मोस्टेट है जो तापमान की जानकारी आपके फोन पर भेजता है। MQTT का उपयोग इस जानकारी को जल्दी से भेजने के लिए किया जा सकता है क्योंकि इसे छोटे डेटा पैकेट्स के लिए डिज़ाइन किया गया है।
- Imagine you have a smart thermostat that sends temperature data to your phone. MQTT can be used to send this information quickly because it's designed for small data packets.

3: MQTT (Message Queuing Telemetry Transport):



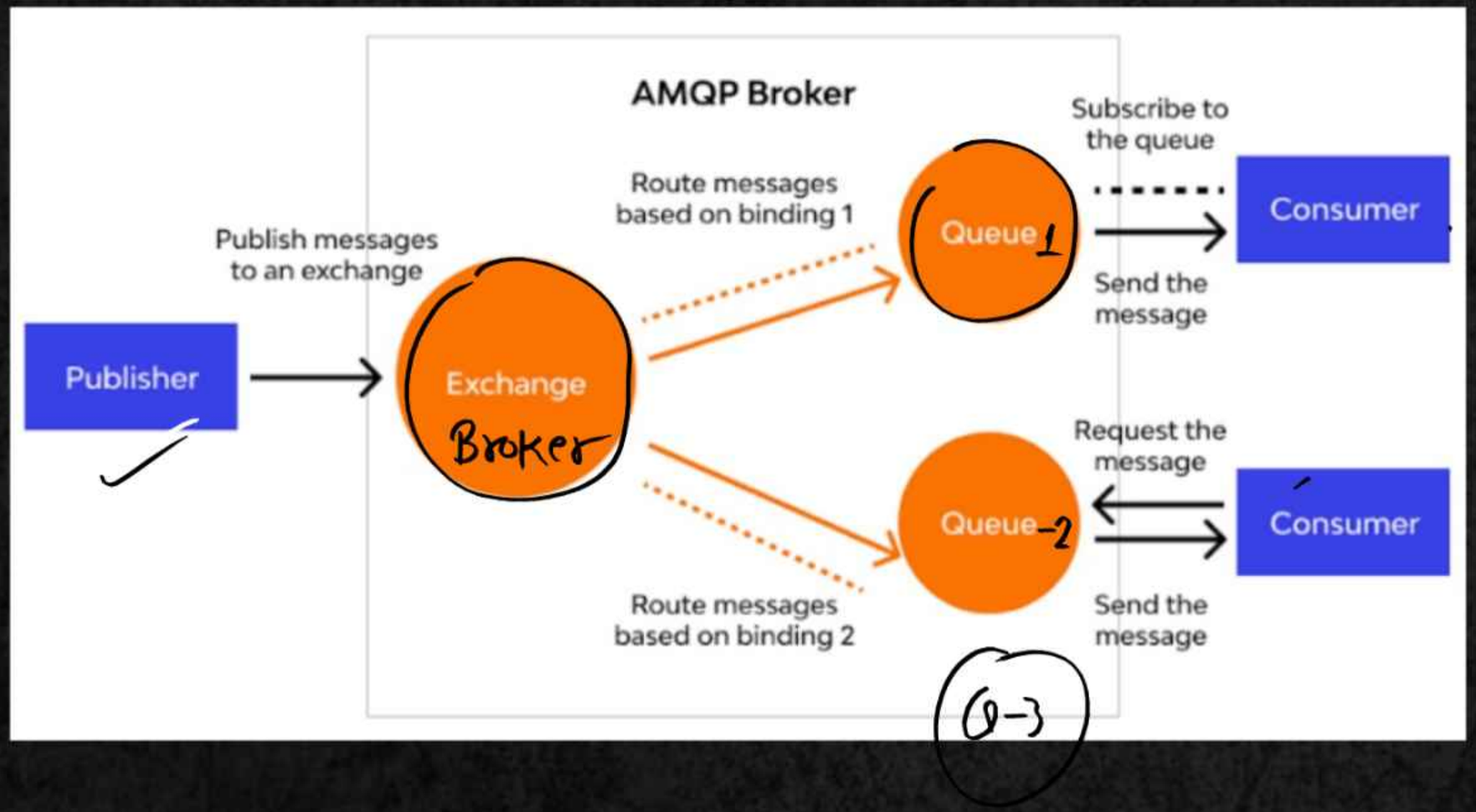
4: Advance Message Queuing Protocol:

- एक मैसेजिंग प्रोटोकॉल है जो विभिन्न एप्लिकेशन या सिस्टम्स को एक-दूसरे के साथ मैसेज भेजने और प्राप्त करने के माध्यम से संचार करने में मदद करता है।
- Advanced Message Queuing Protocol (AMQP) is a messaging protocol that helps different applications or systems communicate with each other by sending and receiving messages.

Component of (AQMP):

- Publisher.
- Consumer.
- Broker.
- Queue.

4: Advance Message Queuing Protocol:



IoT Communication:

- IoT (इंटरनेट ऑफ थिंग्स) में संचार का मतलब है कि कैसे अलग-अलग डिवाइस एक-दूसरे से बातें करते हैं और जानकारी शेयर करते हैं।
- Communication in IoT (Internet of Things) refers to how different devices talk to each other and share information.

IoT Communication Models

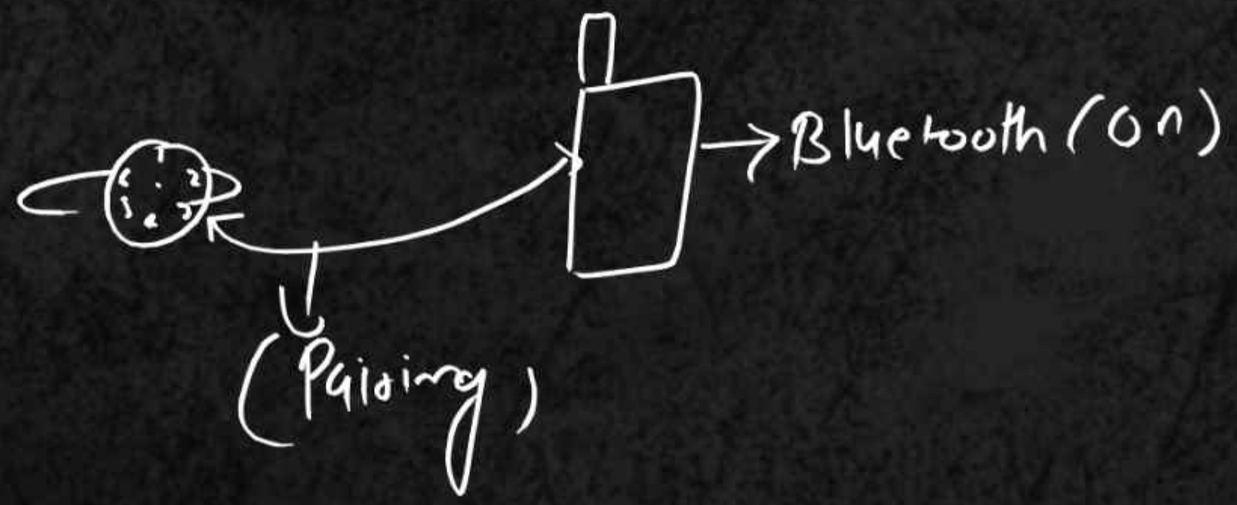
- Device to Device Communication Model.
- Device to Cloud Computing Model.
- Device toGateway Communication Model.
- Back end Data Sharing Communication Model.

1: Device to Device Communication Model.

- इस मॉडल में दो या उससे अधिक IoT डिवाइस सीधे एक-दूसरे से बात करते हैं।
- In this model, two or more IoT devices communicate directly with each other.

उदाहरण:

- आपका स्मार्टफोन और स्मार्टवॉच एक-दूसरे से Bluetooth के ज़रिए डेटा शेयर करते हैं।
- Your smartphone and smartwatch sharing data via Bluetooth.



2: Device to Cloud Computing Model.

- इस मॉडल में IoT डिवाइस इंटरनेट के माध्यम से सीधे क्लाउड सर्वर से कनेक्ट होते हैं। ये डिवाइस क्लाउड में डेटा भेजते हैं, जहाँ उसे स्टोर किया जाता है या प्रोसेस किया जाता है।
- In this model, IoT devices connect to cloud servers via the internet. Devices send data to the cloud where it's stored or processed.

उदाहरण:

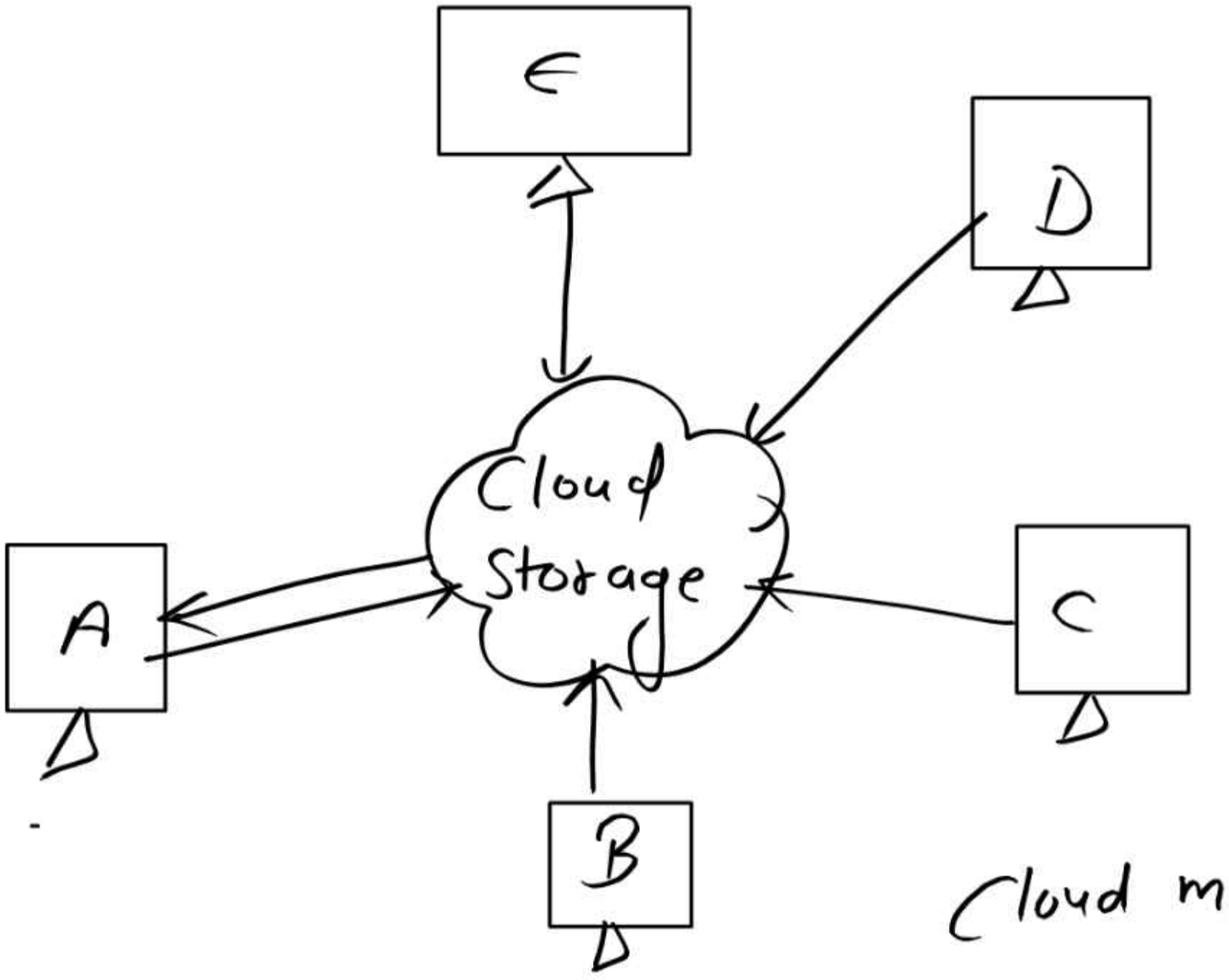
- आपके घर के सिक्योरिटी कैमरे का वीडियो क्लाउड सर्वर पर सेव होता है, जिसे आप कहीं से भी देख सकते हैं।
- A home security camera uploads video footage to a cloud server, which you can access from anywhere.

↳ 24X7 hour.
↳ Remote location access.
↳ low cost

↳ fast access.

↳ Ali-baba
↳ IBM
↳ Google cloud

} Cloud Provider.



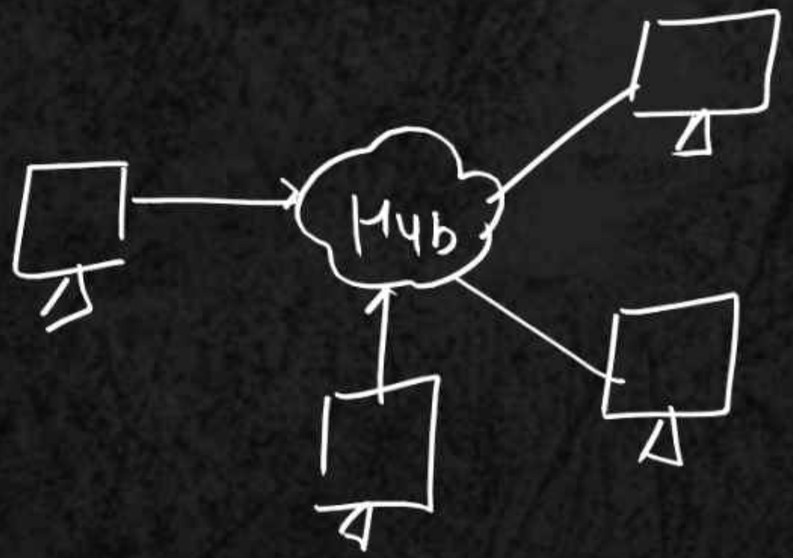
Cloud model

3: Device toGateway Communication Model:

- इसमें IoT डिवाइस एक गेटवे (जैसे राउटर या हब) से कनेक्ट होते हैं, और यह गेटवे जानकारी को प्रोसेस करके क्लाउड या अन्य डिवाइस तक भेजता है।
- Here, IoT devices connect to a gateway (such as a router or hub), which processes the data and sends it to the cloud or other devices

.उदाहरण:

- स्मार्ट होम हब, जो आपके लाइट, कैमरे, और थर्मोस्टेट को कंट्रोल करता है और इंटरनेट से कनेक्ट करता है।
- A smart home hub controls your lights, cameras, and thermostat, and connects them to the internet.



4: Back end Data Sharing Communication Model:

- Back-end Data Sharing Communication Model एक ऐसा IoT Communication Model है जिसमें डेटा को क्लाउड या सर्वर पर इकट्ठा करने के बाद, उसे अन्य एप्लिकेशन, डिवाइसेज़ या सिस्टम के साथ साझा (share) किया जाता है।
- Back-end Data Sharing Communication Model is an IoT Communication Model in which after collecting the data on the cloud or server, it is shared with other applications, devices, or systems.

IoT Communication API:

- API (Application Programming Interface) एक प्रोग्रामिंग इंटरफेस होता है, जो IoT डिवाइसों को आपस में या किसी सर्वर से कनेक्ट करने में मदद करता है।
- API (Application Programming Interface) is a programming interface that helps IoT devices connect to each other or to a server.
- यह API प्रोटोकॉल्स (जैसे HTTP, MQTT, CoAP) का उपयोग करके काम करती है ताकि डिवाइस इंटरनेट के जरिए बात कर सकें।
- It works by using API protocols (e.g. HTTP, MQTT, CoAP) so that devices can talk over the internet.

Types of IoT Communication API:

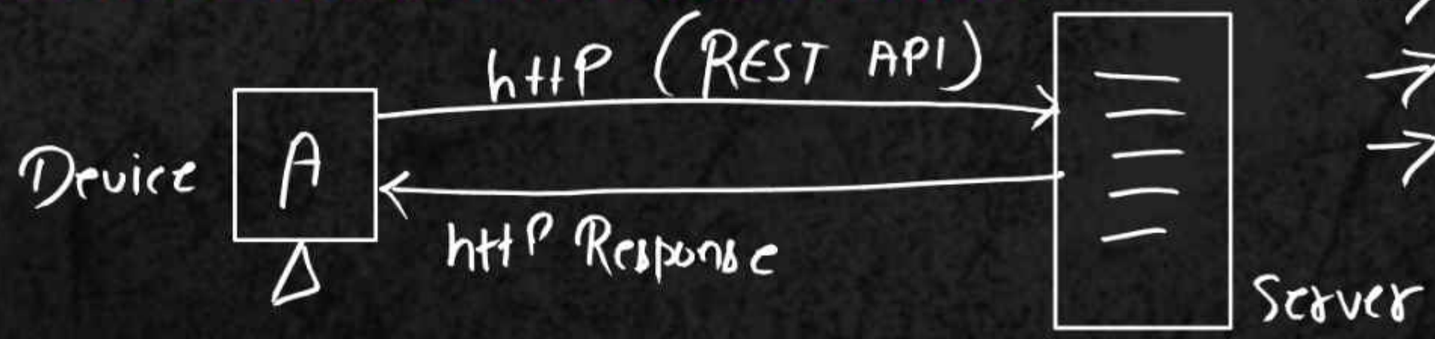
- IoT में API के विभिन्न प्रकार होते हैं, जिनका उपयोग डिवाइसों को कनेक्ट करने, डेटा साझा करने और कंट्रोल करने के लिए किया जाता है।
- IoT has different types of APIs that are used to connect devices, share data, and control them.

  - Control API.
  - Security API.
  - Cloud API.
  - Data API.
  - REST based API.
  - web Socket based API.

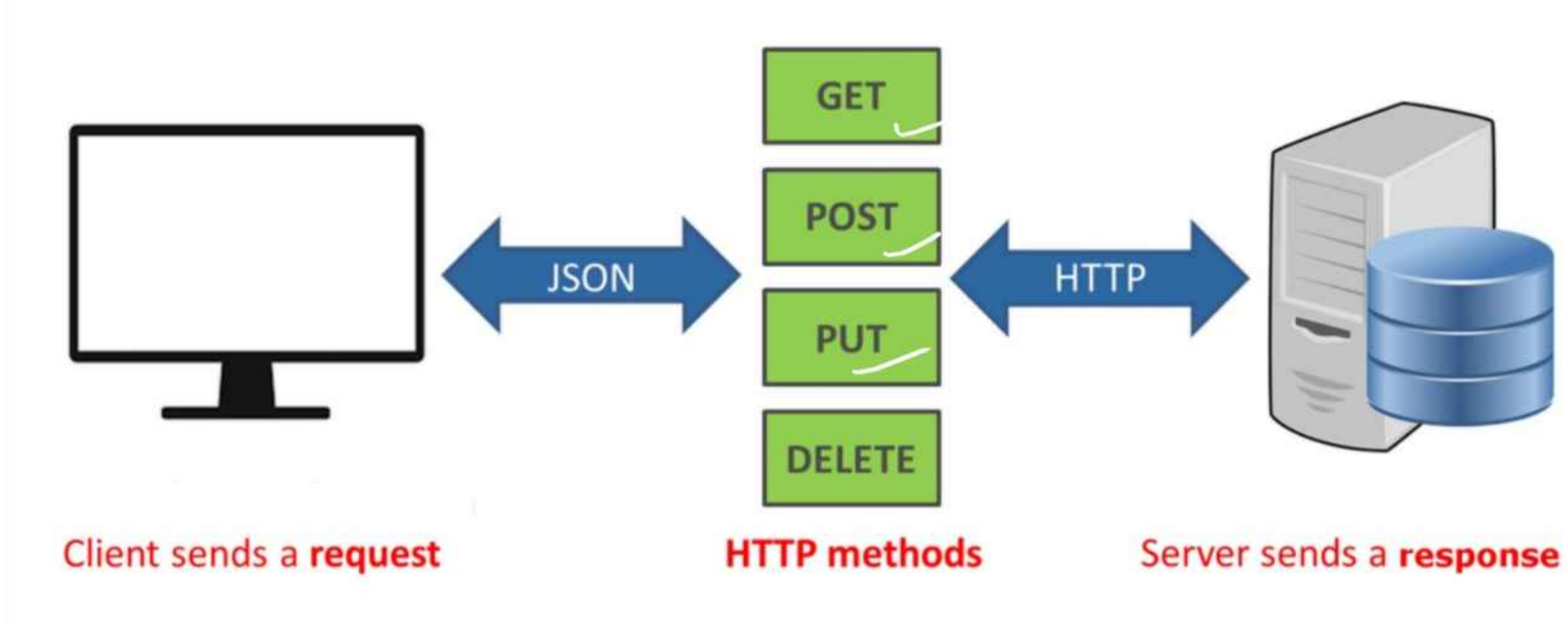
REST Based Communication API:

- REST आधारित API (Representational State Transfer) एक वेब API है, जो HTTP प्रोटोकॉल का उपयोग करके विभिन्न क्लाइंट और सर्वर के बीच डेटा का आदान-प्रदान करती है। REST API का उपयोग आजकल वेब और मोबाइल एप्लिकेशनों में बहुत अधिक होता है, क्योंकि यह सरल, स्केलेबल और लचीला होता है।

- REST based API (Representational State Transfer) is a web API that exchanges data between different clients and servers using the HTTP protocol. REST API is widely used in web and mobile applications nowadays because it is simple, scalable, and flexible.



→ Secure
→ TCP
→ Slow.


GET
POST
PUT
DELETE
JSON
HTTP
Client sends a **request**
HTTP methods
Server sends a **response**

How does REST API work:

GET request:

- यदि आप किसी वेबसाइट पर कोई प्रोडक्ट देखना चाहते हैं, तो क्लाइंट एक GET अनुरोध सर्वर को भेजेगा। सर्वर प्रोडक्ट की जानकारी वापस भेजेगा।
- If you want to view a product on a website, the client will send a GET request to the server. The server will send back the product information.

POST request:

- यदि आप नया ऑर्डर प्लेस करना चाहते हैं, तो आपका ऐप सर्वर को एक POST अनुरोध भेजेगा। सर्वर इस नए ऑर्डर को डेटाबेस में सेव कर लेगा।
- If you want to place a new order, your app will send a POST request to the server. The server will save this new order in the database.

How does REST API work:

PUT request:

- यदि आप अपने ऑर्डर की डिटेल्स को अपडेट करना चाहते हैं, तो एक PUT अनुरोध भेजा जाएगा।
- If you want to update your order details, a PUT request will be sent.

DELETE request:

- अगर आप ऑर्डर को कैंसिल करना चाहते हैं, तो DELETE अनुरोध भेजा जाएगा।
- If you want to cancel the order, a DELETE request will be sent.

Advantage: of REST API:

- REST API HTTP methods (GET, POST, PUT, DELETE) पर आधारित होती है, जो बहुत साधारण और समझने में आसान होती हैं।
- REST APIs में XML की बजाय JSON का उपयोग होता है, जिससे यह fast और lightweight होती हैं।
- REST APIs को किसी भी programming language (Java, Python, PHP, etc.) से call किया जा सकता है।
- हर request अलग होती है — server को पिछली request याद रखने की जरूरत नहीं होती, जिससे performance बढ़ती है।

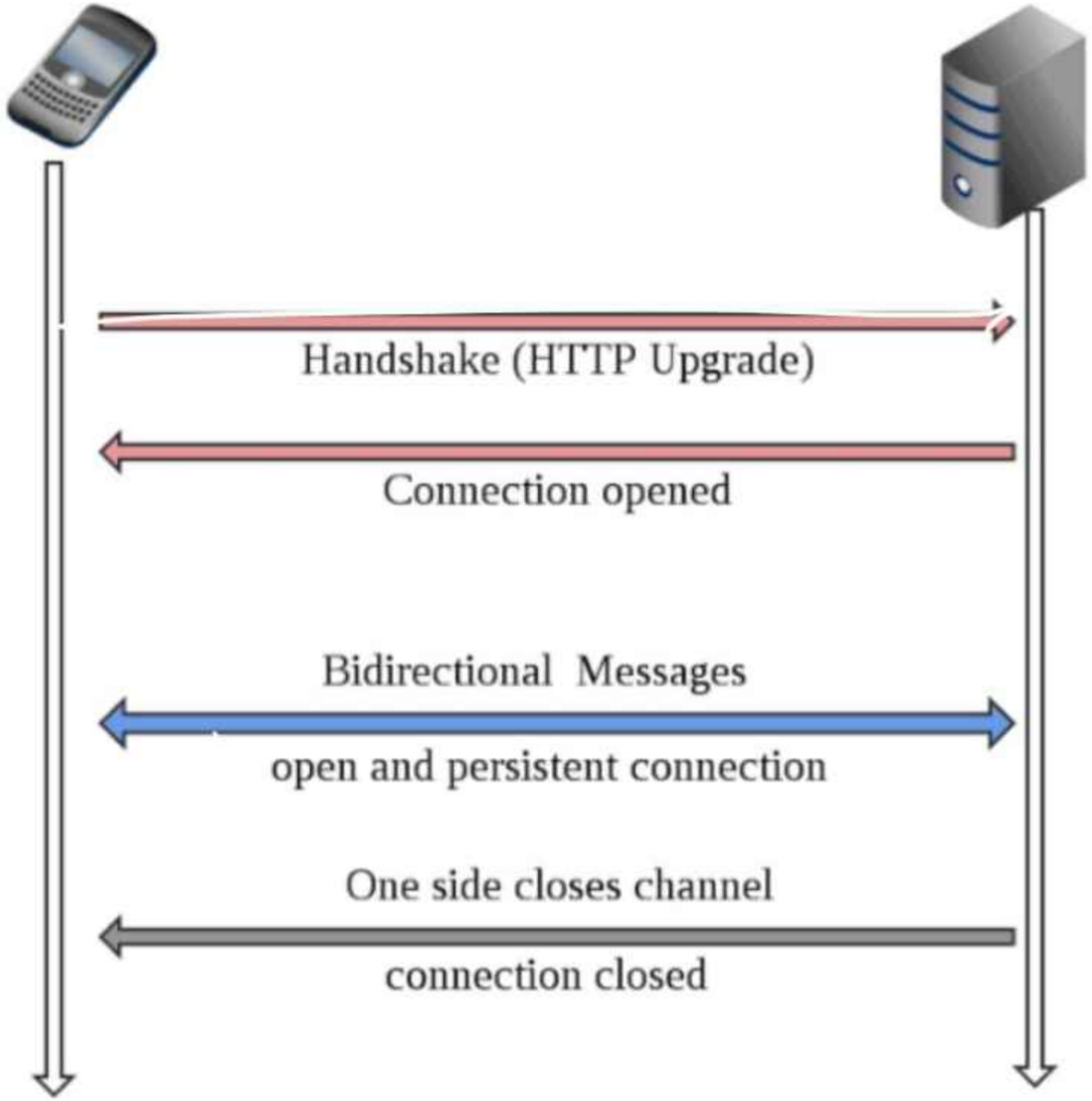
Advantage: of REST API:

- REST APIs are based on HTTP methods (GET, POST, PUT, DELETE), which are very simple and easy to understand.
- REST APIs use JSON instead of XML, which makes them fast and lightweight.
- REST APIs can be called from any programming language (Java, Python, PHP, etc.).
- Each request is unique — the server does not need to remember the previous request, which increases performance.

Web Socket Based Communication API:

- WebSocket आधारित Communication API एक ऐसी तकनीक है जो क्लाइंट (जैसे आपका वेब ब्राउज़र या ऐप) और सर्वर के बीच रीयल-टाइम में दो-तरफा (bidirectional) संचार की सुविधा देती है।
- WebSocket based communication API is a technology that facilitates bidirectional communication between a client (such as your web browser or app) and a server in real-time.

- एक बार जब कनेक्शन बन जाता है, तो क्लाइंट और सर्वर बिना बार-बार कनेक्शन स्थापित किए, लगातार एक-दूसरे को संदेश भेज सकते हैं। यह तकनीक लाइव चैट, ऑनलाइन गेम्स, और स्टॉक मार्केट जैसे लाइव डेटा अपडेट के लिए बहुत उपयोगी होती है।
- Once the connection is established, the client and server can continuously send messages to each other without having to establish the connection again and again.


Handshake (HTTP Upgrade)
Connection opened
Bidirectional Messages
open and persistent connection
One side closes channel
connection closed

How does WebSocket work:

(1.)Connection Setup:

- शुरुआत में, कनेक्शन एक सामान्य HTTP कनेक्शन की तरह शुरू होता है। फिर इसे WebSocket प्रोटोकॉल में अपग्रेड कर दिया जाता है, और यह कनेक्शन तब तक खुला रहता है जब तक कि क्लाइंट या सर्वर इसे बंद न कर दें।.
- The connection starts like a regular web connection (HTTP). But then it switches to WebSocket mode, and the connection stays open until either the client or server decides to close it.

(2.)Two-way communication:

- WebSocket में, क्लाइंट और सर्वर दोनों एक ही समय पर संदेश भेज और प्राप्त कर सकते हैं, बिना किसी देरी के।
- Both the client and the server can send and receive messages at the same time. Neither side has to wait for the other to finish.

How does WebSocket work:

(3.) Real-time updates:

- जैसे ही कोई नया डेटा उपलब्ध होता है, वह तुरंत दूसरी ओर भेजा जाता है।
- WebSockets allow instant updates, meaning as soon as there's new data, it gets sent right away.

Advantageof Web Socket API:

1. Instant data transfer:

- डेटा तुरंत क्लाइंट और सर्वर के बीच ट्रांसफर होता है, बिना किसी देरी के।
- Data is transferred between the client and server instantly, without any delays.

2. Less data overhead:

- एक बार कनेक्शन बन जाने के बाद, बार-बार HTTP हेडर भेजने की जरूरत नहीं होती, जिससे डेटा तेजी से भेजा जाता है।
- Once the connection is open, there's no need to send extra information with every message, making it faster.

3. Two-way communication:

- क्लाइंट और सर्वर दोनों एक ही समय में डेटा भेज सकते हैं।
- Both sides can send messages at the same time, making interactions smoother.

IoT Enabling Technologies:

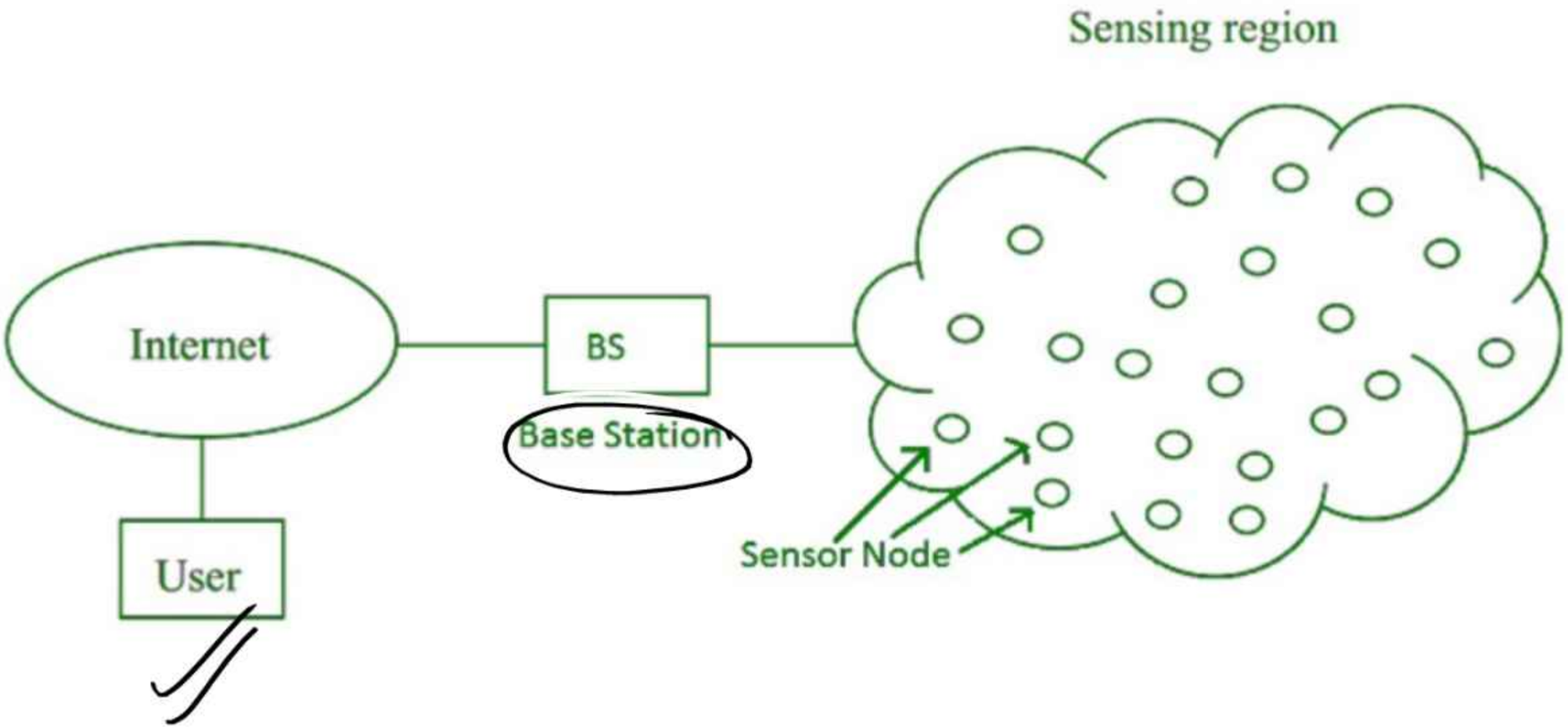
- "IoT Enabling Technologies वे जरूरी तकनीकें होती हैं जो IoT डिवाइस को काम करने लायक बनाती हैं। ये तकनीकें डिवाइस को आपस में जोड़ती हैं, डाटा इकट्ठा करती हैं, उसे समझती हैं और स्मार्ट तरीके से काम कराती हैं।"
- "IoT Enabling Technologies are the essential technologies that make IoT devices work. These technologies connect devices, collect data, understand it, and make them work smarter."

Technologies:

- Wireless Sensor Networks
- Cloud Computing
- Big Data Analytics
- Communication Protocols
- Embedded Systems

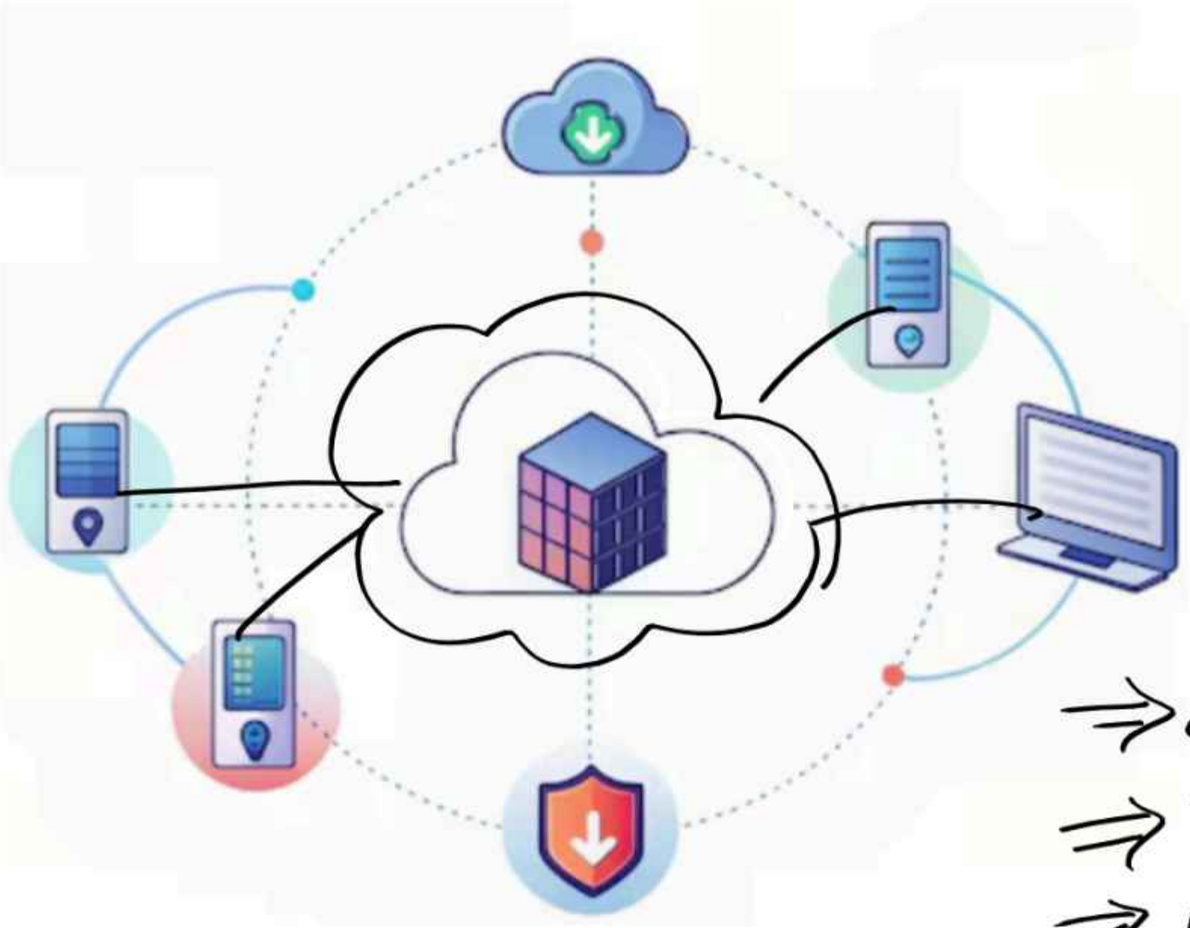
1: Wireless Sensor Networks:

- वायरलेस सेंसर नेटवर्क (WSN) एक ऐसा नेटवर्क होता है जिसमें कई छोटे-छोटे सेंसर डिवाइस होते हैं, जो बिना तार (wireless) के एक-दूसरे से जुड़े होते हैं। ये सेंसर अलग-अलग स्थानों पर लगे होते हैं और वातावरण से डेटा (जैसे तापमान, नमी, गति आदि) इकट्ठा करते हैं
- A wireless sensor network (WSN) is a network consisting of many small sensor devices that are connected to each other wirelessly. These sensors are installed at different locations and collect data (such as temperature, humidity, motion, etc.) from the environment


Sensing region
Internet
BS
Base Station
User
Sensor Node

2: Cloud Computing:

- क्लाउड कंप्यूटिंग हमें इंटरनेट पर एक वर्चुअल स्टोरेज और कंप्यूटिंग पावर देता है, जिससे हम कहीं से भी और किसी भी डिवाइस से अपनी फाइलें या एप्लिकेशन एक्सेस कर सकते हैं।
- Cloud computing gives us a virtual storage and computing power over the Internet, so we can access our files or applications from anywhere and from any device.
- उदाहरण के लिए, जब हम Google Drive या Dropbox में फाइलें सेव करते हैं, तो वे फाइलें हमारे कंप्यूटर पर नहीं, बल्कि "क्लाउड" पर होती हैं, और हम उन्हें इंटरनेट से एक्सेस कर सकते हैं
- For example, when we save files to Google Drive or Dropbox, those files are not on our computer, but in the "cloud", and we can access them from the Internet



⇒ 24X7
⇒ Remote access
⇒ Any device can access.

3: Big Data Analytics :

- जब हमारे पास बहुत सारा डेटा होता है, जैसे कि सोशल मीडिया से मिलने वाली जानकारी, ऑनलाइन शॉपिंग की डिटेल्स, या किसी कंपनी के लाखों ग्राहकों के रिकॉर्ड, तो इस डेटा को समझने और इसमें से उपयोगी जानकारी निकालने के लिए बिग डेटा एनालिटिक्स का इस्तेमाल किया जाता है।
- When we have a lot of data, such as information from social media, online shopping details, or millions of customer records of a company, big data analytics is used to understand this data and extract useful information from it.

4 DATA ANALYSIS

3 CLEANING OF DATA

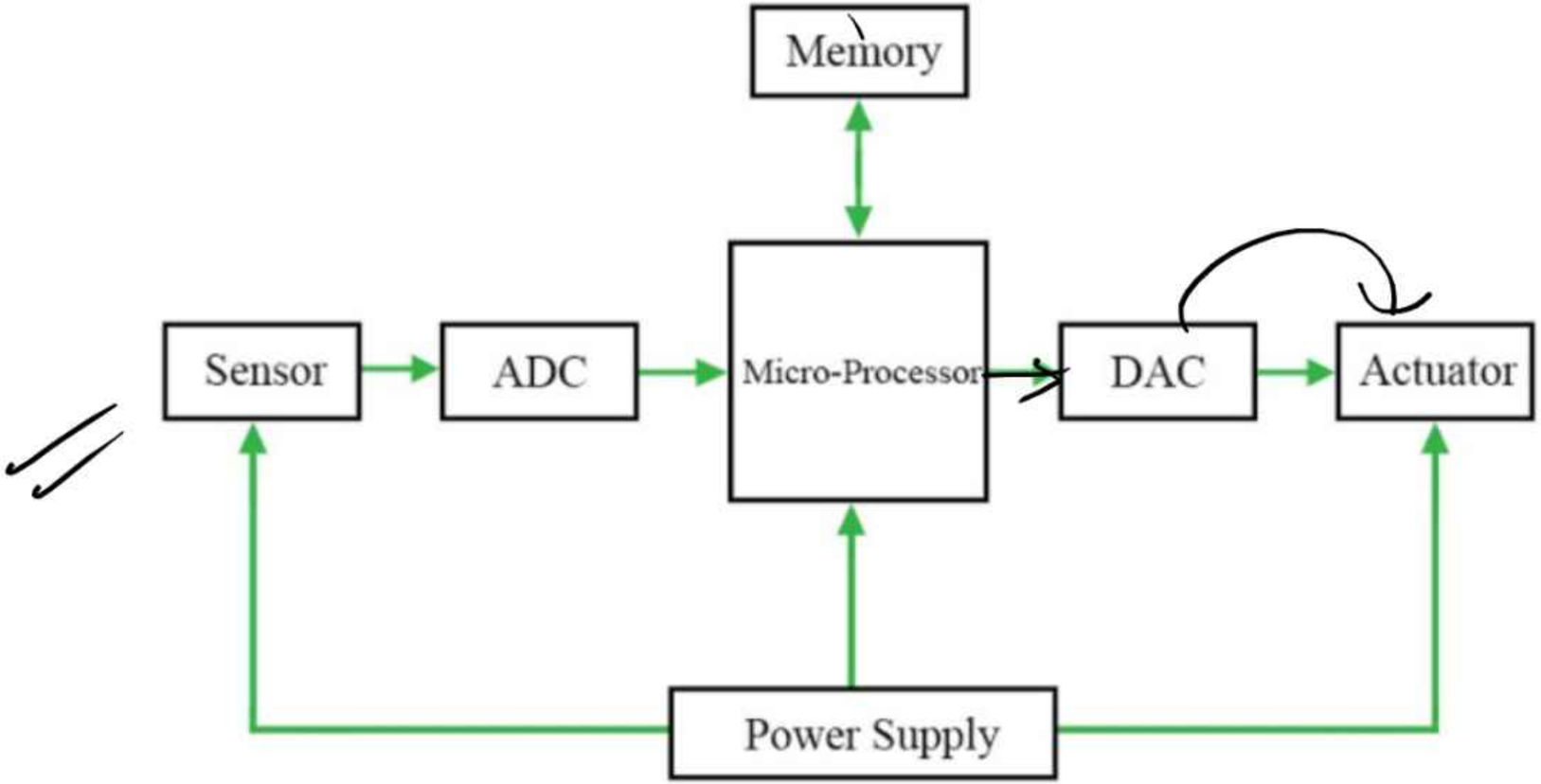
2 DATA PROCESSING

1 DATA COLLECTION

Name	Class	Sub
A	=	=
B	=	=
C	=	X

4: Embedded Systems

- Embedded System एक विशेष प्रकार का कंप्यूटर सिस्टम होता है, जो किसी बड़ी मशीन या डिवाइस के अंदर लगाया जाता है और उसे कंट्रोल करने के लिए डिजाइन किया जाता है। इसमें हार्डवेयर (जैसे माइक्रोकंट्रोलर) और सॉफ्टवेयर दोनों शामिल होते हैं,
- Embedded System is a special type of computer system that is installed inside a larger machine or device and is designed to control it. It includes both hardware (such as microcontroller) and software,


Memory
Sensor
ADC
Micro-Processor
DAC
Actuator
Power Supply

Chapter No-1 (introduction (IoT)

Q 1: What do you mean by IoT? What are the features of IoT?
Q 1:IoT से आपका क्या अभिप्राय है? IoT की विशेषताएँ क्या हैं?

Q 2: What do you mean by protocol? Explain IoT protocol.
Q 2:प्रोटोकॉल से आपका क्या तात्पर्य है? IoT प्रोटोकॉल की व्याख्या करें।

Q 3: Explain logical and physical design in IoT with their components?
Q 3:उनके घटकों के साथ IoT में तार्किक और भौतिक (logical and physical) डिजाइन की व्याख्या करें?

Q 4: What is REST in REST based communication API?
Q 4:REST आधारित संचार एपीआई में REST क्या है?

Q 5: Name the IoT enabling technology.
Q 5:IoT सक्रिय-प्रौद्योगिकी नामित कीजिये।

Chapter No: 2 Introduction to (IoT)

syllabus:

How electronic devices fit with the Internet of Things, and why they are important: Breadboard and its internal connections, LED and its connections, Tri-color LED, Resistor Introduction to the many 'end devices', sensors and actuators, differentiate between different sensor types.

electronic devices fit with the Internet of Things:

- IoT (Internet of Things) में इलेक्ट्रॉनिक डिवाइसेज़ को इस प्रकार डिजाइन और कनेक्ट किया जाता है कि वे इंटरनेट के माध्यम से डेटा को एकत्र कर सकें, भेज सकें, प्राप्त कर सकें और एक-दूसरे या क्लाउड सिस्टम के साथ मिलकर काम कर सकें। इन डिवाइसेज़ में सेंसर, कंट्रोलर, नेटवर्क कनेक्टिविटी और इंटेलिजेंट सॉफ्टवेयर लगे होते हैं, जिससे ये डिवाइस 'स्मार्ट' हो जाती हैं।"
- In IoT, electronic devices are designed and connected in such a way that they can collect data, send it, receive it, and work together through the internet. These devices include sensors, controllers, network connectivity modules, and intelligent software, which makes them smart.

Step-by-Step Process:

Step 1: Sensor और Actuator जोड़ना:

Sensors = यह डिवाइस के आसपास के वातावरण से जानकारी इकट्ठा करते हैं।

Sensors = These collect information from the environment around the device.

- (जैसे तापमान, ध्वनि, प्रकाश, गति आदि)
- (such as temperature, sound, light, motion etc.)

Actuators = यह एक्शन लेते हैं (ON/OFF करना, गेट खोलना आदि)।

Actuators = These perform actions (turning ON/OFF, opening gates, etc.).

Step 2: Connectivity जोड़ना (इंटरनेट से जोड़ना)

- डिवाइस को इंटरनेट से जोड़ने के लिए निम्न तकनीकों का उपयोग होता है:
- The following technologies are used to connect a device to the Internet:

example: Wi-Fi, Bluetooth, Zigbee, 4G/5G, LoRaWAN

Step 3: Embedded System या Microcontroller लगाना:

- यह छोटा कंप्यूटर चिप होता है जो डिवाइस को कंट्रोल करता है
- This is a small computer chip that is digitally controlled.

Example: वॉशिंग मशीन में माइक्रोकंट्रोलर होता है जो wash cycle को कंट्रोल करता है।

- The washing machine has a microcontroller which controls the wash cycle.

Step 4: क्लाउड या सर्वर से डेटा भेजना:

- सेंसर जो डेटा इकट्ठा करते हैं, वो क्लाउड पर भेजा जाता है। वहाँ उसका analysis होता है और कोई action plan बनता है।
- The data collected by the sensors is sent to the cloud. There it is analyzed and an action plan is made.

Step 5: AI/ML से स्मार्ट Decision लेना:

- Artificial Intelligence (AI) और Machine Learning (ML) के जरिए डिवाइस खुद से निर्णय ले सकती है।
- Through Artificial Intelligence (AI) and Machine Learning (ML), the device can take decisions on its own.

Step 6: Security और Privacy लगाना:

- IoT डिवाइस डेटा शेयर करती हैं, इसलिए इनकी सुरक्षा जरूरी होती है।
- IoT devices share data, so their security is important.
  - Data Encryption
  - Passwords
  - Authentication
  - Firewall

IoT डिवाइसेज़ का महत्त्व (Importance of IoT Devices):

1. स्वचालन (Automation):

- IoT डिवाइसेज़ इंसानों की बजाय खुद से काम करने में सक्षम होते हैं। उदाहरण के लिए, अगर आपने एक स्मार्ट बल्ब को इस तरह सेट किया है कि वह रात को 7 बजे अपने आप जल जाए और सुबह 6 बजे बंद हो जाए, तो वह बिना किसी मैनुअल हस्तक्षेप के यह कार्य करेगा।
- IoT devices are capable of performing tasks on their own instead of humans. For example, if you have set a smart bulb to turn on automatically at 7 pm and turn off at 6 am, it will do so without any manual intervention.

2. रीयल-टाइम निगरानी (Real-Time Monitoring)

- IoT डिवाइसेज़ सेंसर के माध्यम से लगातार डेटा को इकट्ठा करते हैं और तुरंत यूज़र या सिस्टम को भेजते हैं। इसका उपयोग चिकित्सा, सुरक्षा, औद्योगिक प्रक्रियाओं आदि में होता है।

- IoT devices continuously collect data through sensors and send it immediately to the user or system. It is used in medical, security, industrial processes, etc.

3. ऊर्जा की बचत (Energy Saving):
- IoT डिवाइसेज़ तब ही चालू होते हैं जब जरूरत होती है, जिससे अनावश्यक बिजली की खपत नहीं होती।
- IoT devices only turn on when needed, eliminating unnecessary power consumption.

4. सुरक्षा और निगरानी (Security and Surveillance):
- IoT डिवाइसेज़ जैसे स्मार्ट डोर लॉक, सिक्योरिटी कैमरा, अलार्म सिस्टम आदि हमें सुरक्षा प्रदान करते हैं। ये डिवाइस इंटरनेट से जुड़े होते हैं, जिससे हम दूर बैठकर भी घर या ऑफिस की निगरानी कर सकते हैं।
- IoT devices such as smart door locks, security cameras, alarm systems, etc. provide us security. These devices are connected to the internet, so that we can monitor the home or office even from a distance.

5. स्वास्थ्य सेवाओं में उपयोग (Use in Healthcare)

- IoT डिवाइसेज़ जैसे स्मार्ट वॉच या बैंड हमारे स्वास्थ्य से जुड़ी जानकारी जैसे हार्ट रेट, स्टेप्स, कैलोरीज़ आदि को मापते हैं और रिकॉर्ड करते हैं।
- IoT devices such as smart watches or bands measure and record our health-related information such as heart rate, steps, calories, etc.

6. कृषि में स्मार्ट तकनीक (Smart Farming)

- IoT का प्रयोग खेती में भी हो रहा है। सेंसर मिट्टी की नमी, तापमान, और मौसम की स्थिति को मापते हैं और उसी के अनुसार सिंचाई या खाद डाली जाती है।
- IoT is also being used in farming. Sensors measure soil moisture, temperature, and weather conditions and irrigation or fertilizer is applied accordingly.

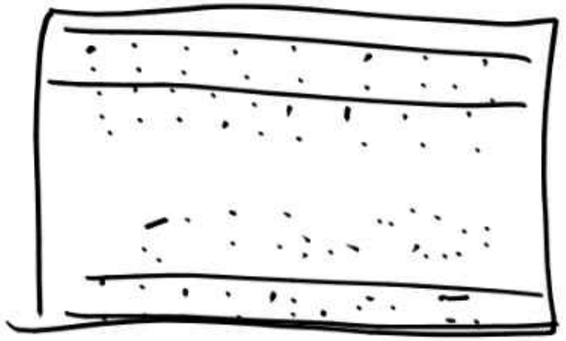
7. डेटा आधारित निर्णय (Data-Driven Decision Making)

- IoT डिवाइसेज़ से जो डेटा प्राप्त होता है, उसका विश्लेषण करके हम स्मार्ट निर्णय ले सकते हैं।
- We can make smart decisions by analyzing the data we get from IoT devices.

ब्रेडबोर्ड (Breadboard):

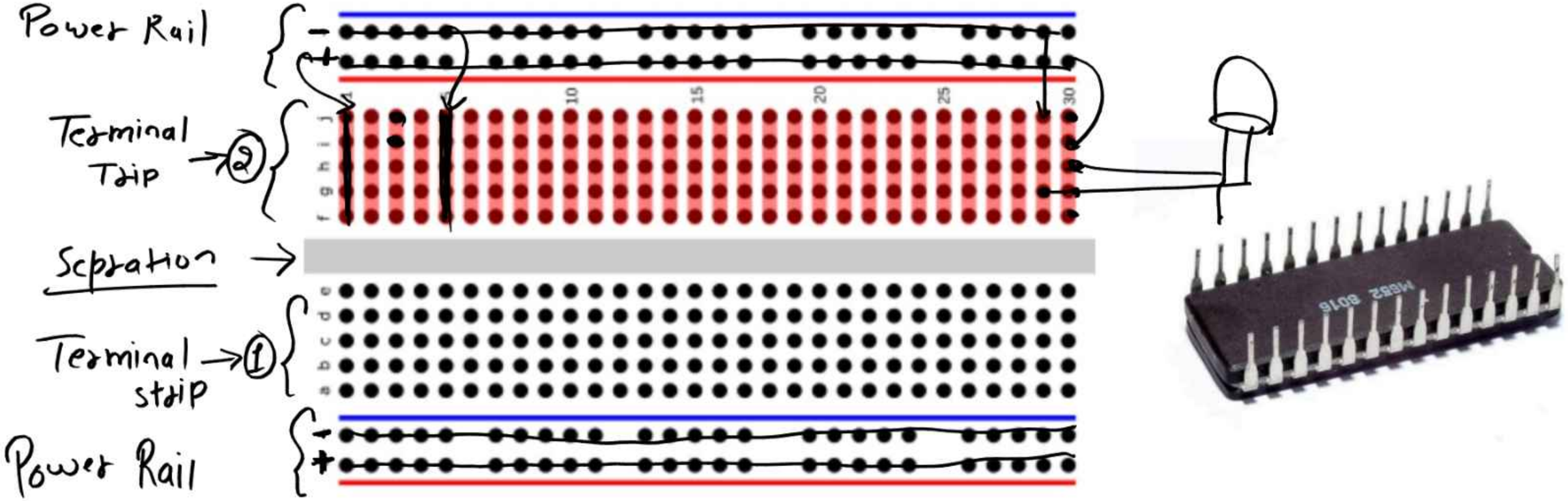
- Breadboard एक ऐसी प्लेट होती है जिसका उपयोग इलेक्ट्रॉनिक्स सर्किट को अस्थायी रूप से जोड़ने के लिए किया जाता है। इसमें छोटे-छोटे छेद होते हैं, जिनमें वायर या कंपोनेंट्स (जैसे रेजिस्टर, LED, ट्रांजिस्टर आदि) को आसानी से फिट किया जा सकता है। इसका फायदा यह है कि इसमें सर्किट को बिना सोल्डरिंग के बनाया और टेस्ट किया जा सकता है। यह खासकर शुरुआती स्तर पर प्रयोग करने के लिए उपयोगी होता है।

- Breadboard is a plate used to temporarily assemble electronic circuits. It has small holes in which wires or components (such as resistors, LEDs, transistors, etc.) can be easily fitted. Its advantage is that circuits can be built and tested in it without soldering. It is especially useful for experimenting at the initial level.



ब्रेडबोर्ड आर्किटेक्चर (Breadboard architecture):

- The architecture of a breadboard is quite simple and can be broken down into three main sections:
- Breadboard की आर्किटेक्चर काफी सरल होती है। इसे तीन मुख्य भागों में बांटा जा सकता है:



Component of Breadboard:

(1.) Power Rails:

- Breadboard के किनारे पर दो लंबी लाइनें होती हैं, जिन्हें पावर रेल्स कहा जाता है। एक लाइन पॉजिटिव (+) पावर के लिए होती है और दूसरी नेगेटिव (-) या ग्राउंड के लिए। इनसे सर्किट के अलग-अलग हिस्सों को पावर सप्लाई की जाती है।
- Along the edges of the breadboard, there are two long rows called power rails. One row is for positive (+) power, and the other is for negative (-) or ground. These are used to distribute power to different parts of the circuit.

(2.) Terminal Strips (टर्मिनल स्ट्रिप्स):

- बीच में कई कॉलम और रो होते हैं। हर कॉलम में पांच छेद होते हैं, और ये छेद आपस में जुड़े होते हैं। इसका मतलब यह है कि एक कॉलम के पांच छेदों में डाला गया कोई भी वायर या कंपोनेंट एक ही पॉइंट पर कनेक्ट होता है। इस स्ट्रिप का इस्तेमाल सर्किट के मुख्य हिस्सों को जोड़ने के लिए किया जाता है।

- In the middle, there are columns and rows of holes. Each column typically has five connected holes, meaning anything plugged into the same column is electrically connected. This section is used to connect the main components of the circuit.

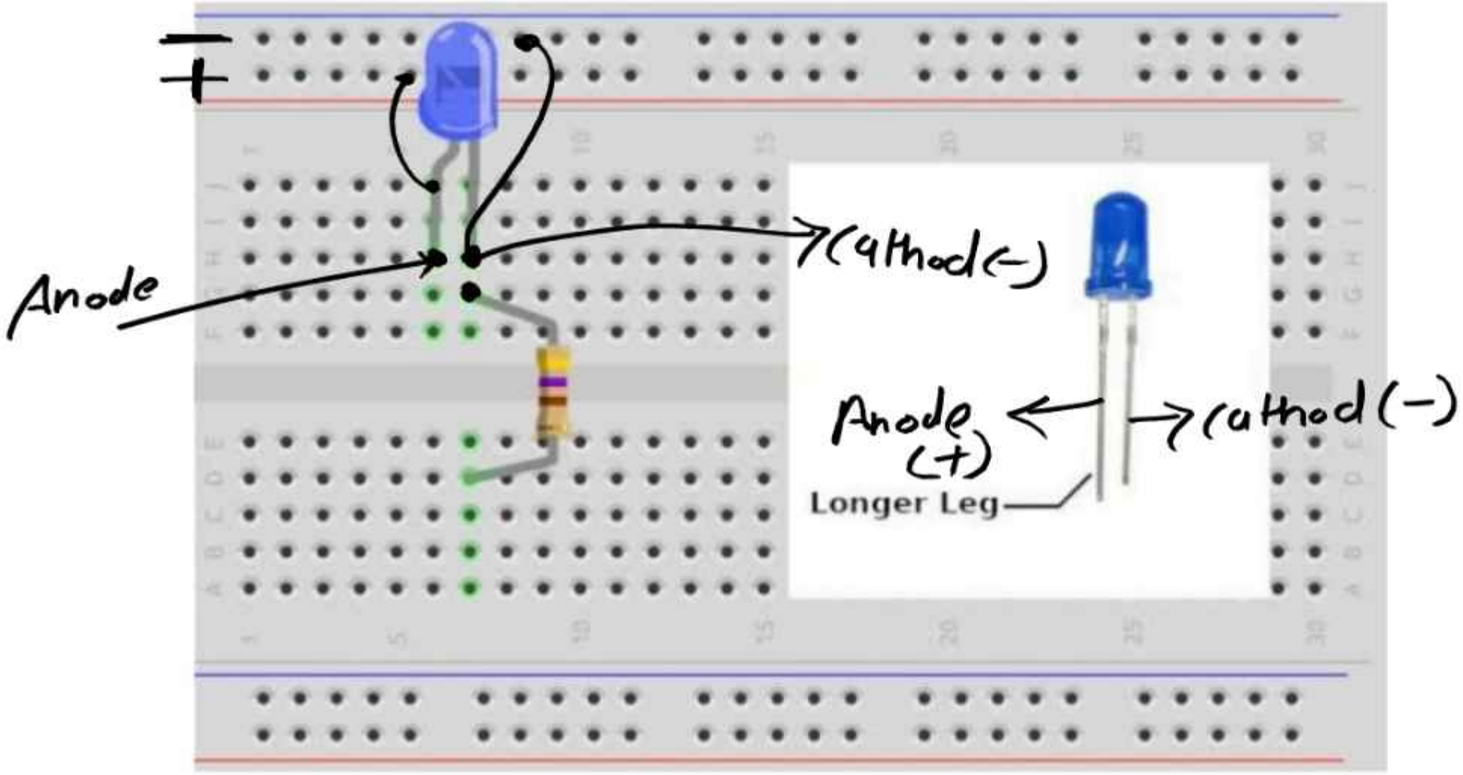
(3.) Separation Line (सेंटर गैप या सेपरेशन लाइन):

- बीच में एक खाली जगह होती है, जो दो टर्मिनल स्ट्रिप्स को अलग करती है। यह जगह मुख्य रूप से Integrated Circuits (ICs) को रखने के लिए होती है, ताकि IC के अलग-अलग पिन्स को दोनों तरफ कनेक्ट किया जा सके।

- There is a gap in the middle of the breadboard that separates two sets of terminal strips. This gap is useful for placing Integrated Circuits (ICs) because the two sides can connect to different pins of the IC.

ब्रेडबोर्ड कनेक्शन (Breadboard connection):

- Breadboard पर LED, रेजिस्टर, और बैटरी को कनेक्ट करना बहुत आसान है।
- Connecting an LED, resistor, and battery on a breadboard is simple.



कनेक्शन स्टेप्स (Connection Steps):

(1.) पावर सप्लाई कनेक्ट करें (Connect the power supply):

- सबसे पहले, Breadboard के एक साइड की पावर रेल्स में बैटरी के पॉजिटिव (+) और नेगेटिव (-) वायर कनेक्ट करें। बैटरी के पॉजिटिव वायर (लाल) को पावर रेल्स के पॉजिटिव लाइन से और नेगेटिव वायर (काला) को नेगेटिव लाइन से कनेक्ट करें।
- First, connect the positive (+) and negative (-) wires of the battery to the power rails on one side of the breadboard. Connect the positive wire (red) of the battery to the positive line of the power rails and the negative wire (black) to the negative line.

(2.)LED को Breadboard में लगाएं (Place the LED on the Breadboard):

- LED के दो पिन होते हैं: Anode (लंबा पिन) और Cathode (छोटा पिन)। LED का Anode (लंबा पिन) उस रो में लगाएं, जिसे आप पॉजिटिव (+) पावर सप्लाई से कनेक्ट करना चाहते हैं। LED का Cathode (छोटा पिन) उस रो में लगाएं, जिसे आप रेजिस्टर के साथ जोड़ेंगे।

- The LED has two pins: Anode (long pin) and Cathode (short pin). Insert the Anode (long pin) into a row where you want to connect it to the positive (+) power supply. Insert the Cathode (short pin) into a different row where you'll connect the resistor.

(3.)रेजिस्टर कनेक्ट करें(Connect the Resistor):

- 220Ω या 1kΩ का रेजिस्टर लें और इसे LED के Cathode (छोटे पिन) वाले रो में लगाएं।
- Take a 220Ω or 1kΩ resistor and connect one leg of the resistor to the same row where the LED's Cathode (short pin) is connected
- रेजिस्टर का दूसरा सिरा Breadboard की नेगेटिव (-) पावर रेल्स से कनेक्ट करें।
- Connect the other leg of the resistor to the negative (-) power rail on the breadboard.

लाइट एमिटिंग डायोड (Light Emitting Diode):

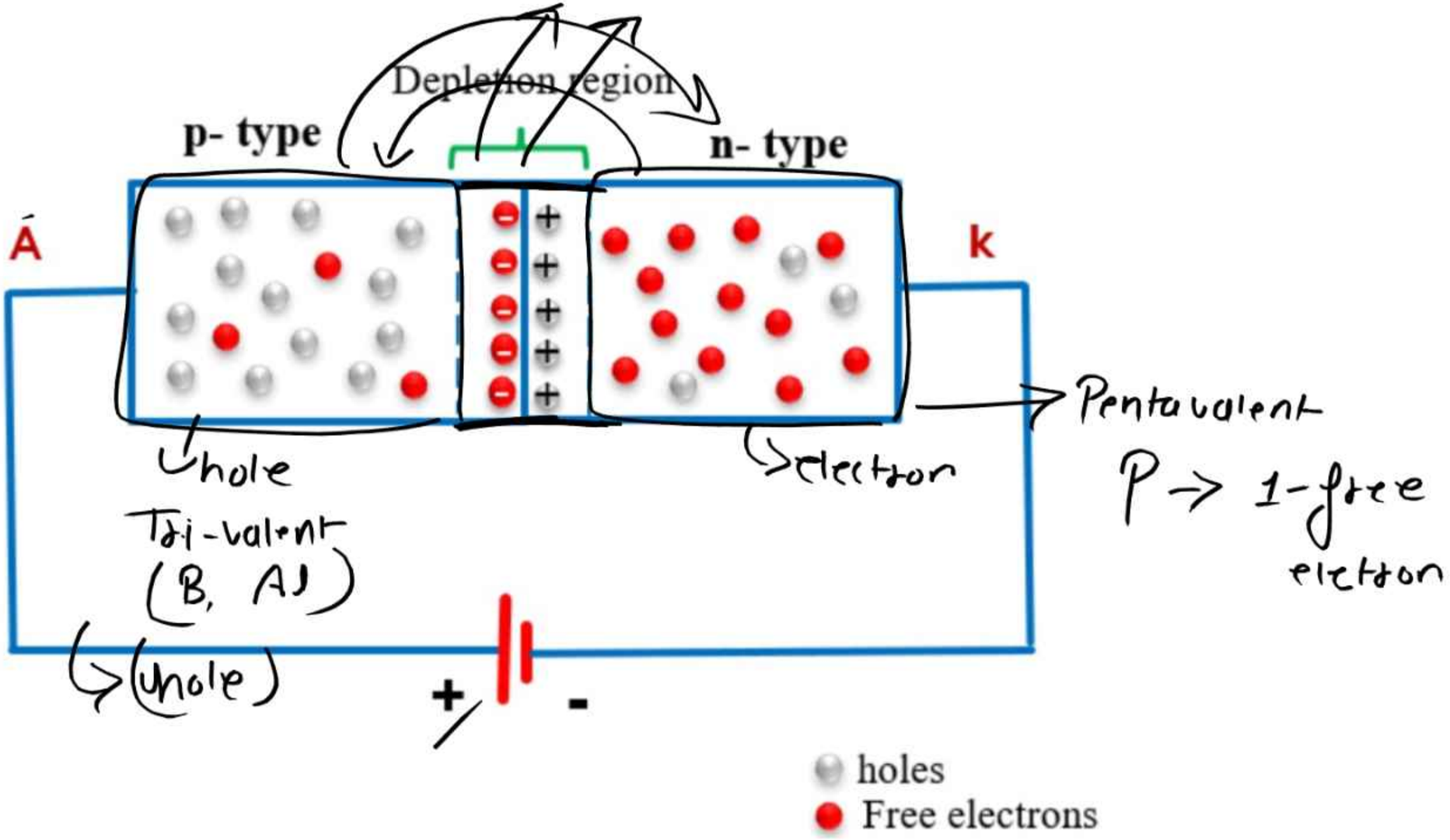
- लाइट एमिटिंग डायोड (LED) एक अर्धचालक युक्ति (Semiconductor Device) है जो विद्युत धारा प्रवाहित होने पर प्रकाश उत्सर्जित (Emit) करती है।
- Light Emitting Diode (LED) is a semiconductor device that emits light when electric current flows through it.
- यह डिवाइस तब प्रकाश निकालती है जब उसमें से Forward Bias में करंट गुजारा जाता है।
- This device emits light when current is passed through it in forward bias.

विशेषताएँ (Features of LED):

- बहुत ही कम बिजली खपत (Low power consumption)
- लंबी लाइफ (Long life span)
- तुरंत जलने वाली (Instant ON)
- अलग-अलग रंगों में उपलब्ध (Red, Green, Blue, White etc.)
- पर्यावरण के अनुकूल (No mercury or UV radiation)

Working:

- यह एक सेमीकंडक्टर डिवाइस है जो इलेक्ट्रॉन्स और होल्स की क्रिया पर आधारित है।
- It is a semiconductor device that works based on the movement of electrons and holes.


Depletion region
p- type
n- type
A
k
hole
Tri-valent
(B, Al)
(hole)
electron
Pentavalent
P → 1-free electron
+
-
holes
Free electrons

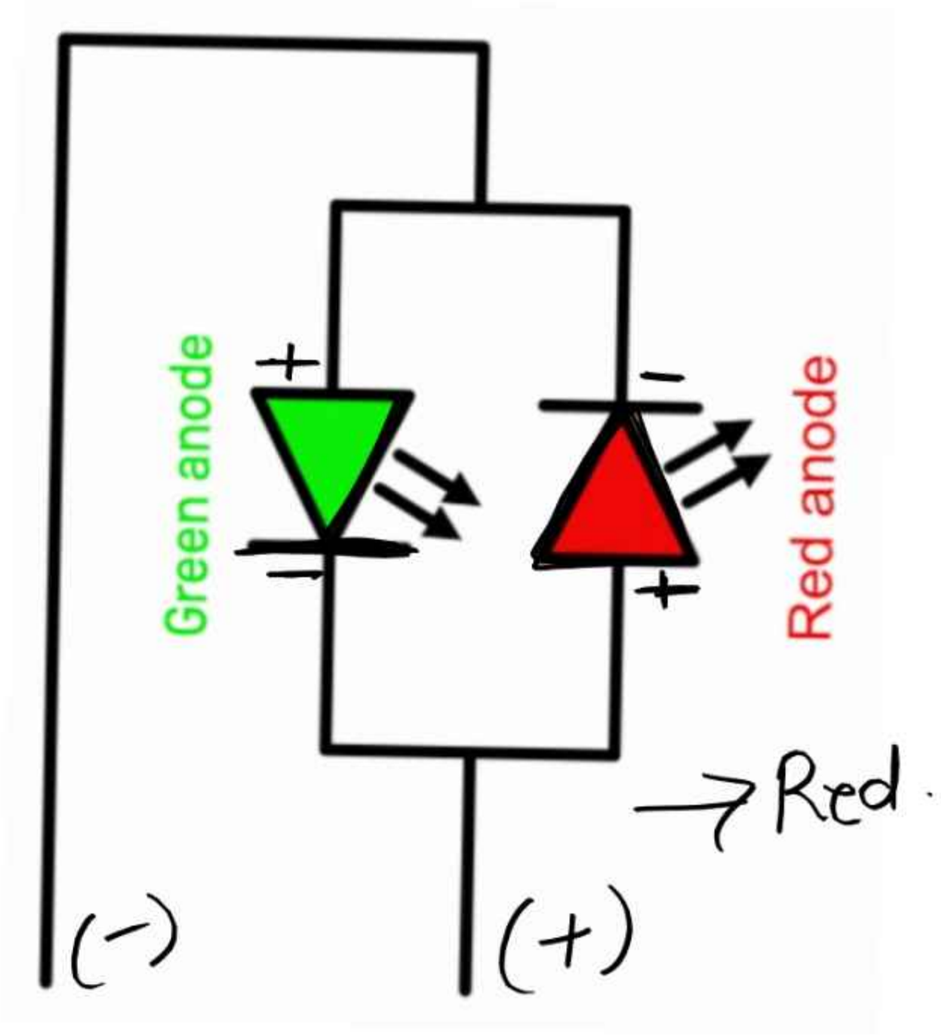
- **Step 1: LED को Forward Bias में जोड़ना:**
  - Positive टर्मिनल → P-side से
  - Negative टर्मिनल → N-side से
  - इसे कहते हैं Forward Bias
- **Step 2: Carrier Movement (कैरेयर की गति):**
  - N-side के Electrons → Junction की ओर आते हैं
  - P-side के Holes → Junction की ओर आते हैं
  - Junction पर Electron + Hole मिलते हैं
- **Step 3: Energy Release & Light Production:**
  - जब कोई Electron, किसी Hole में गिरता है,
  - वह अपनी Extra Energy छोड़ता है
  - यह ऊर्जा Photon (लाइट की कण) के रूप में निकलती है
  - इस प्रक्रिया को कहते हैं: Electroluminescence

Types of LED:

(1): द्वि-रंग एलईडी (Bi- Color LED):

- Bi-Color LED एक ऐसी LED होती है, जो दो अलग-अलग रंगों में प्रकाश उत्सर्जित कर सकती है।
- A Bi-Color LED is an LED that can emit light in two different colors.
- इसे आमतौर पर दो रंगों को दिखाने के लिए उपयोग किया जाता है, जैसे हरा और लाल। Bi-Color LED में दो अलग-अलग LEDs एक ही पैकेज में होती हैं, जो एक ही LED में दो रंगों की सुविधा प्रदान करती हैं।
- It is commonly used to display two colors, such as green and red. Bi-Color LEDs contain two separate LEDs in the same package, providing the convenience of two colors in a single LED.

(1): द्वि-रंग एलईडी (Bi- Color LED):



एक रंग चालू करना (Turning on a color) :

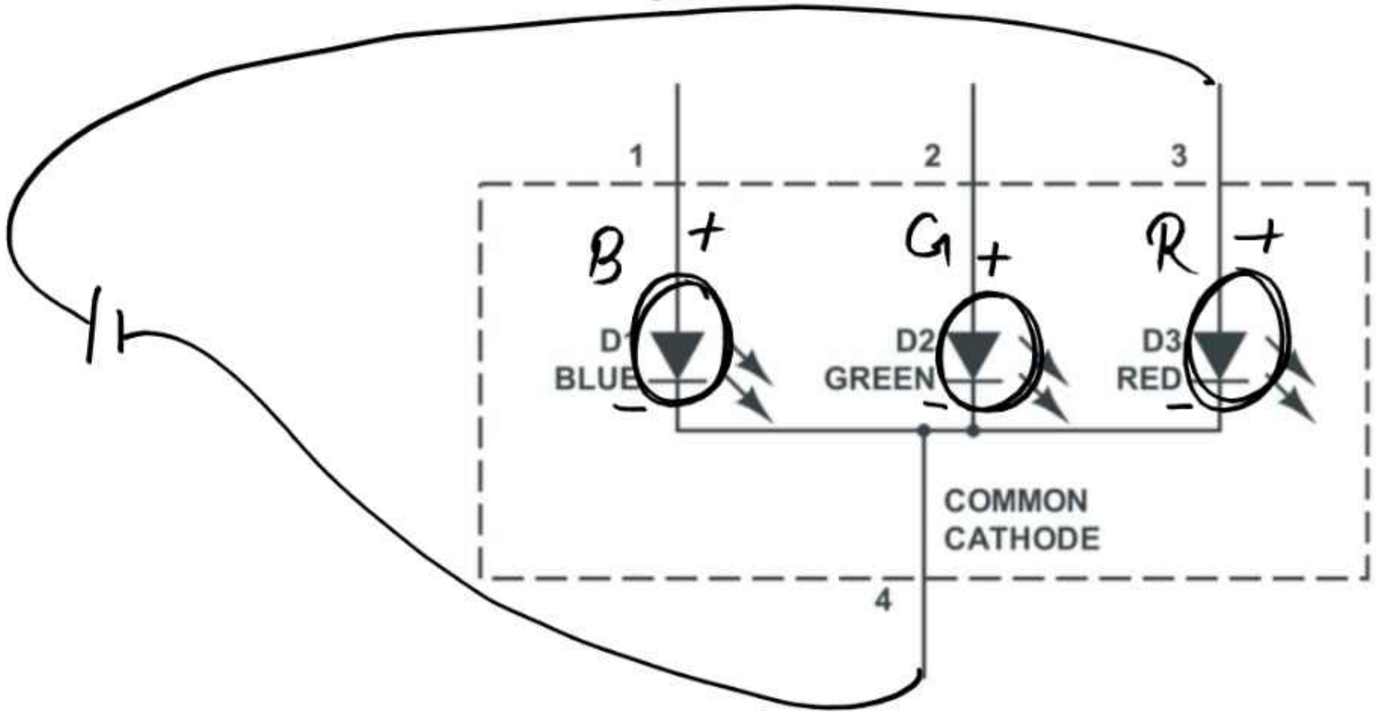
- जब आप पहले पिन को पॉजिटिव वोल्टेज और कॉमन पिन को नेगेटिव वोल्टेज से जोड़ते हैं, तो पहला LED जलता है (जैसे लाल)।
- When you connect the first pin to positive voltage and the common pin to negative voltage, the first LED lights up (e.g. red).

दूसरा रंग चालू करना (Turning on another color:) :

- अगर आप दूसरे पिन को पॉजिटिव वोल्टेज और कॉमन पिन को नेगेटिव वोल्टेज से जोड़ते हैं, तो दूसरा LED जलता है (जैसे हरा)।
- If you connect the other pin to positive voltage and the common pin to negative voltage, then another LED lights up (e.g. green).

(2): त्रि रंग एलईडी(tri Color LED): :

- Tri-Color LED एक ऐसी LED होती है, जो तीन अलग-अलग रंगों में प्रकाश उत्पन्न कर सकती है। आमतौर पर यह तीन रंगों में होती है: लाल (Red), हरा (Green), और नीला (Blue)। यह एक ही पैकेज में तीन अलग-अलग LEDs को जोड़ती है,
- A Tri-Color LED is an LED that can emit light in three different colors, typically red, green, and blue. It combines three separate LEDs in one package



Working:

1. प्रत्येक LED को नियंत्रित करना (Controlling Each LED):

- जब आप लाल LED वाले पिन को पॉजिटिव वोल्टेज से जोड़ते हैं और कॉमन पिन को नेगेटिव से, तो लाल रंग का प्रकाश निकलता है।
- When you apply positive voltage to the red pin and connect the common pin to the negative, the red LED lights up.
- इसी तरह, हरा पिन और नीला पिन अलग-अलग वोल्टेज से जोड़कर हरा और नीला प्रकाश उत्पन्न किया जा सकता है।
- Similarly, you can light up the green and blue LEDs by applying voltage to their respective pins.

LED के उपयोग (Applications of LED)

1. घरेलू लाइटिंग (Home Lighting).
2. इलेक्ट्रॉनिक डिवाइस में इंडिकेटर.
3. डिस्प्ले स्क्रीन (Display Panels).
4. ऑटोमोबाइल्स (वाहनों में).
5. डिजिटल घड़ियों और कैलकुलेटर में.
6. सड़क और ट्रैफिक लाइट्स में.
7. इमरजेंसी और टॉर्च लाइट्स.
8. डेकोरेशन और त्योहारों में.

1. Home Lighting.
2. Indicators in electronic devices.
3. Display Panels.
4. Automobiles (vehicles).
5. In digital watches and calculators.
6. In street and traffic lights.
7. Emergency and torch lights.
8. In decoration and festivals.

रेजिस्टर (Resistor):

- रेजिस्टर एक ऐसा इलेक्ट्रॉनिक उपकरण है, जो विद्युत धारा (electric current) के प्रवाह को नियंत्रित या सीमित करता है। इसे एक प्रतिरोधक कहा जाता है क्योंकि यह करंट के प्रवाह का प्रतिरोध करता है।
- A resistor is an electronic device that controls or limits the flow of electric current. It is called a resistor because it resists the flow of current.

- रेजिस्टर का मुख्य कार्य सर्किट में करंट और वोल्टेज को नियंत्रित करना है ताकि सर्किट और उपकरण सुरक्षित रहें।
- The main function of a resistor is to control the current and voltage in a circuit so that the circuit and equipment remain safe.

रेजिस्टर का सिद्धांत (Working Principle of a Resistor):

- रेजिस्टर का काम ओम के नियम (Ohm's Law) पर आधारित होता है, जो कहता है:
- The operation of a resistor is based on Ohm's Law, which states:
- $V = I * R$

  जहाँ(where)
  - V = वोल्टेज (Voltage)
  - I = विद्युत धारा (Current)
  - R = रेजिस्टेंस (Resistance)

इस नियम के अनुसार(According to this law):

- अगर रेजिस्टेंस बढ़ता है, तो करंट कम होता है।
- अगर रेजिस्टेंस घटता है, तो करंट बढ़ता है।
- If the resistance (R) increases, the current (I) decreases.
- If the resistance (R) decreases, the current (I) increases.

रेजिस्टर की विशेषताएँ (Characteristics of Resistor):

1. Resistance (प्रतिरोध):

- "Resistance एक भौतिक गुण है, जो विद्युत धारा के प्रवाह का विरोध करता है। यह पदार्थ की लंबाई, क्षेत्रफल और सामग्री पर निर्भर करता है।"
- "Resistance is a physical property that opposes the flow of electric current. It depends on the length, area and material of the substance."

2. Tolerance (टॉलरेंस):

- टॉलरेंस यह दर्शाता है कि रजिस्टर की वास्तविक वैल्यू उसकी नामांकित (लिखी गई) वैल्यू से कितनी ऊपर या नीचे हो सकती है।
- Tolerance indicates how much the actual value of the register can be above or below its nominated (written) value.

3. Power Rating (पावर रेटिंग):

- "Power rating वह अधिकतम शक्ति होती है, जिसे रजिस्टर बिना खराब हुए या जलने के सहन कर सकता है।"
- "Power rating is the maximum power the register can withstand without damaging or burning out."

4. Temperature Coefficient (तापमान गुणांक):

- "Temperature coefficient दर्शाता है कि तापमान में एक डिग्री सेल्सियस परिवर्तन पर रजिस्टर की प्रतिरोधकता कितने ppm (parts per million) से बदलती है।"
- "Temperature coefficient shows how much ppm (parts per million) the resistance of the register changes per one degree Celsius change in temperature."

5. Noise (शोर):

- "Noise वह अनचाहा (Unwanted) इलेक्ट्रॉनिक सिग्नल है, जो रजिस्टर के अंदर के इलेक्ट्रॉनिक मूवमेंट से उत्पन्न होता है और सर्किट में विक्षेप पैदा करता है।"
- "Noise is the unwanted electronic signal which is generated by the electronic movement inside the register and creates deflection in the circuit."

6. Linearity (रेखीयता):

- "Linearity उस गुण को कहते हैं जिसमें रजिस्टर के वोल्टेज और करंट के बीच सीधा (प्रत्यक्ष) संबंध होता है और वह हमेशा Ohm's Law (V = IR) का पालन करता है।"
- "Linearity is the property in which there is a direct relationship between the voltage and current of the register and it always follows Ohm's Law (V = IR)."

7. Physical Size (भौतिक आकार):

- "Physical Size रजिस्टर की बनावट और पावर रेटिंग पर निर्भर करता है। उच्च पावर वाले रजिस्टर बड़े होते हैं क्योंकि उन्हें ज्यादा गर्मी सहनी होती है।"
- "Physical size depends on the design and power rating of the register. Higher power registers are larger because they have to dissipate more heat."

6. Linearity (रेखीयता):

- "Linearity उस गुण को कहते हैं जिसमें रजिस्टर के वोल्टेज और करंट के बीच सीधा (प्रत्यक्ष) संबंध होता है और वह हमेशा Ohm's Law (V = IR) का पालन करता है।"
- "Linearity is the property in which there is a direct relationship between the voltage and current of the register and it always follows Ohm's Law (V = IR)."

Sensor (सेंसर):

- "Sensor एक ऐसा इलेक्ट्रॉनिक डिवाइस होता है जो किसी भौतिक Quantity (जैसे – तापमान, दबाव, प्रकाश, गति, आदि) को मापता है और उसे इलेक्ट्रिकल सिग्नल में बदल देता है।"
- "A sensor is an electronic device that measures a physical quantity (such as temperature, pressure, light, motion, etc.) and converts it into an electrical signal."

Example:

- Temperature sensor कमरे का तापमान मापता है और उसे डिजिटल डेटा में बदलता है।
- The Temperature sensor measures the room temperature and converts it into digital data.
- Light sensor यह बताता है कि कितना प्रकाश मौजूद है।
- The Light sensor measures how much light is present.
- Motion sensor यह पता लगाता है कि कोई चीज़ हिली या नहीं।
- The Motion sensor detects whether something has moved or not.

इलेक्ट्रॉनिक्स में विभिन्न प्रकार के सेंसर (Different types of sensor in electronics):

- इलेक्ट्रॉनिक्स में विभिन्न प्रकार के सेंसर होते हैं, जिनमें से प्रत्येक को विशिष्ट भौतिक मापदंडों का पता लगाने के लिए डिज़ाइन किया गया है।
- In electronics, there are various types of sensors, each designed to detect specific physical parameters.

- Temperature Sensor
- Ultrasonic Sensor
- Proximity Sensor
- Level Sensor
- Force Sensor
- IR Sensor
- Touch Sensor
- Pressure Sensor
- Smoke And Gas Sensor
- Humidity Sensor

1:-तापमान सेंसर (Temperature Sensor):

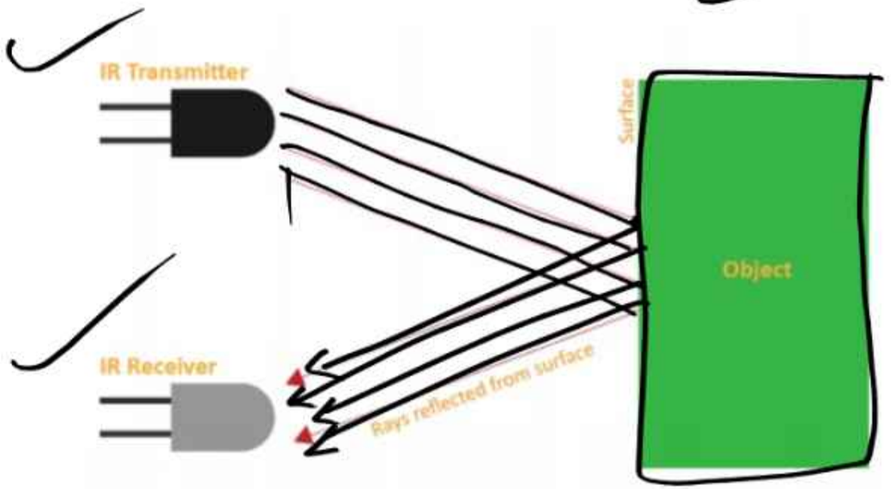
- एक ऐसा उपकरण है जो तापमान को मापने का काम करता है। यह सेंसर किसी वस्तु या स्थान के गर्म या ठंडे होने का पता लगाकर उस तापमान को मापता है और इसे एक संकेत (सिग्नल) में बदल देता है जिसे मशीन या उपकरण समझ सकता है।
- A sensor is a device that measures temperature. This sensor measures the temperature of an object or place by detecting whether it is hot or cold and converts it into a signal that a machine or device can understand.

Example: थर्मिस्टर (Thermistor), थर्मोकपल (Thermocouple),
आरटीडी (RTD - Resistance Temperature Detector),
इंफ्रारेड (Infrared) तापमान सेंसर. etc

2:-इन्फ्रारेड सेंसर (IR Sensor)

- इन्फ्रारेड सेंसर एक ऐसा उपकरण है जो इन्फ्रारेड विकिरण (Infrared Radiation) को महसूस करता है। यह सेंसर बिना संपर्क के किसी वस्तु का तापमान माप सकता है या उसके आसपास की हलचल का पता लगा सकता है।
- An infrared sensor is a device that senses infrared radiation. This sensor can measure the temperature of an object without contact or detect movement around it.

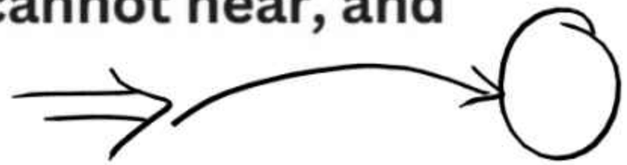
Example: पैसिव इन्फ्रारेड सेंसर (PIR Sensor), एक्टिव इन्फ्रारेड सेंसर (Active IR Sensor).



3: अल्ट्रासोनिक सेंसर (Ultrasonic Sensor):

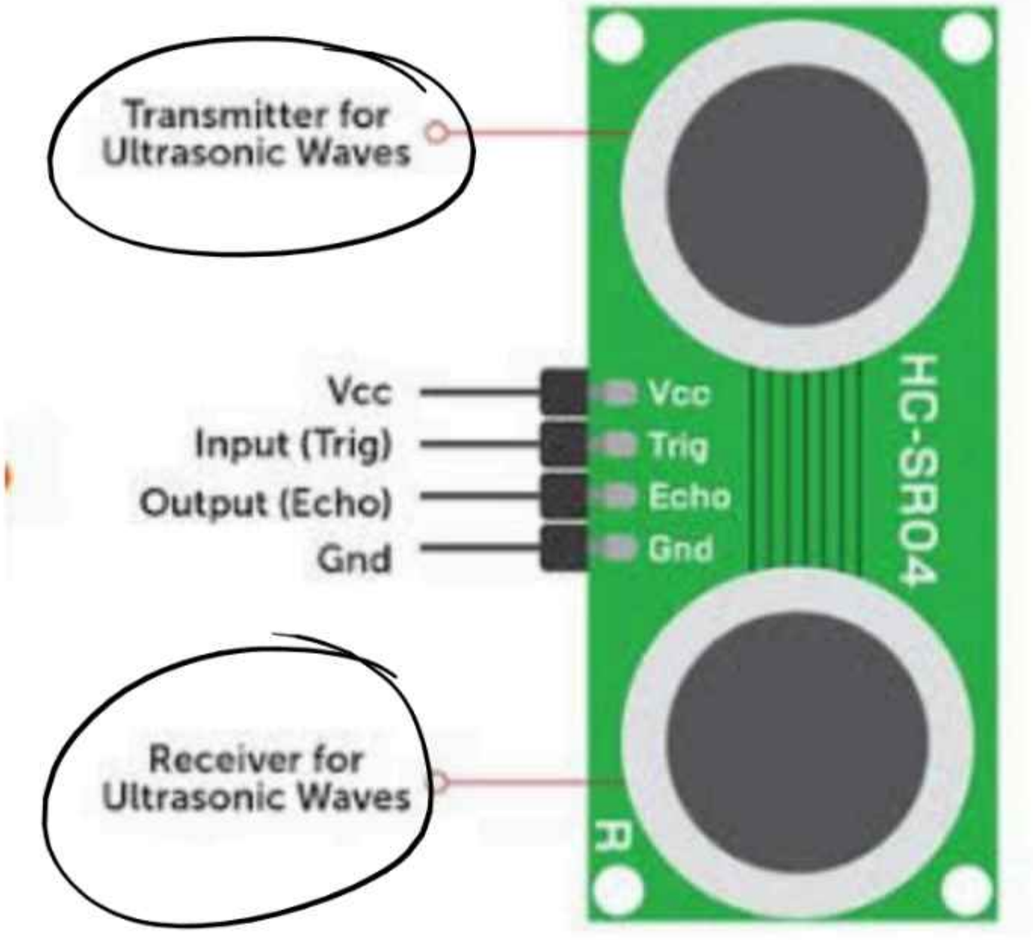
- अल्ट्रासोनिक सेंसर एक उपकरण है जो ध्वनि तरंगों का उपयोग करके दूरी मापता है।
- An ultrasonic sensor is a device that measures distance using sound waves.

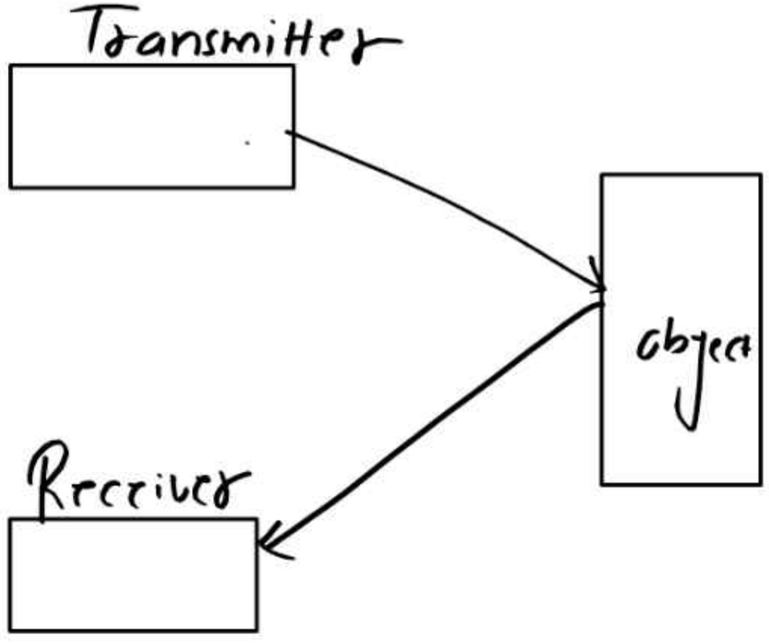
- यह सेंसर ध्वनि की एक विशेष तरंग, जिसे हम सुन नहीं सकते, को भेजता है और उस तरंग के वापस आने में लगने वाले समय का मापन करता है।
- This sensor sends out a special wave of sound, which we cannot hear, and measures the time it takes for that wave to return.



- यह तकनीक विभिन्न अनुप्रयोगों में उपयोग की जाती है, जैसे कि वस्तुओं की दूरी मापने, गति पहचानने, और ऑब्जेक्ट डिटेक्शन में।
- This technology is used in various applications, such as measuring the distance of objects, motion detection, and object detection.

3: अल्ट्रासोनिक सेंसर (Ultrasonic Sensor):





4: टच सेंसर (Touch Sensor):

- टच सेंसर एक ऐसा उपकरण है जो किसी वस्तु या व्यक्ति के स्पर्श (टच) का पता लगाता है। यह सेंसर तब सक्रिय होता है जब इसे कोई व्यक्ति या वस्तु छूता है या उस पर हल्का दबाव डालता है। टच सेंसर का उपयोग आधुनिक उपकरणों और स्मार्टफोन, टैबलेट, और लैपटॉप जैसी डिवाइसों में बड़े पैमाने पर किया जाता है।
- A touch sensor is a device that detects the touch of an object or person. This sensor is activated when a person or object touches it or applies slight pressure on it. Touch sensors are used extensively in modern gadgets and devices like smartphones, tablets, and laptops.

5: प्रेशर सेंसर (Pressure Sensor):

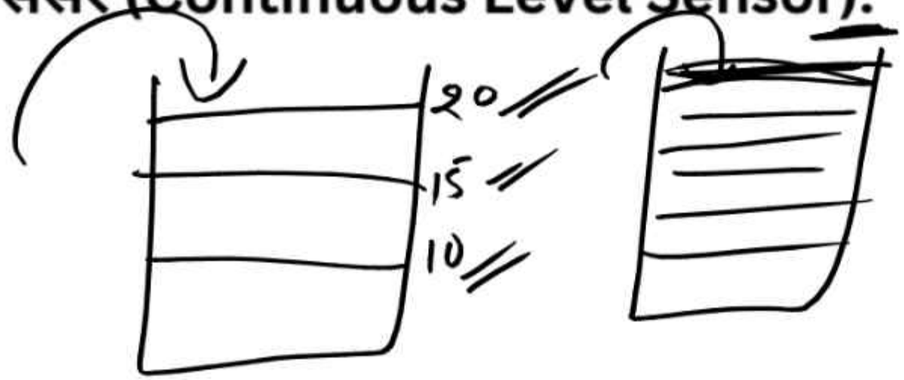
- प्रेशर सेंसर एक ऐसा उपकरण है जो गैस या तरल पदार्थ के दबाव (प्रेशर) को मापता है। यह सेंसर दबाव में होने वाले परिवर्तन को पहचानकर उसे एक इलेक्ट्रिकल सिग्नल में बदल देता है। इसका उपयोग विभिन्न उपकरणों और मशीनों में दबाव को मापने और नियंत्रित करने के लिए किया जाता है।
- A pressure sensor is a device that measures the pressure of a gas or liquid. This sensor detects the change in pressure and converts it into an electrical signal. It is used to measure and control pressure in various devices and machines.

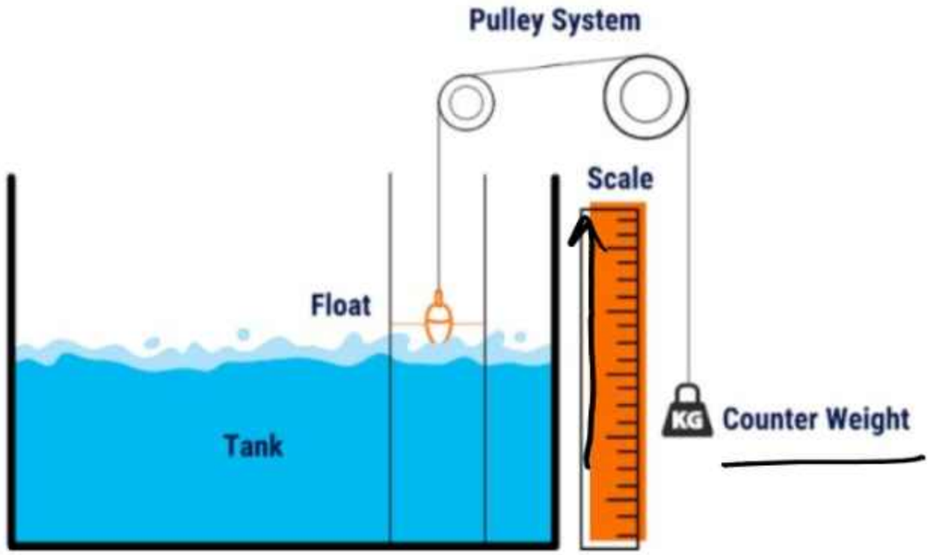
7: लेवल सेंसर (Level Sensor):

- लेवल सेंसर एक ऐसा उपकरण है जो किसी कंटेनर, टैंक, या रिवर जैसी जगहों में तरल (liquid) या ठोस पदार्थ (solid) के स्तर (लेवल) का पता लगाता है। इसका उपयोग यह मापने के लिए किया जाता है कि किसी जगह पर कितना तरल या ठोस पदार्थ भरा हुआ है और कब उसे भरने या खाली करने की जरूरत है।
- A level sensor is a device that detects the level of a liquid or solid in a container, tank, or river. It is used to measure how much liquid or solid is filled in a space and when it needs to be filled or emptied.

Types of Level Sensor :

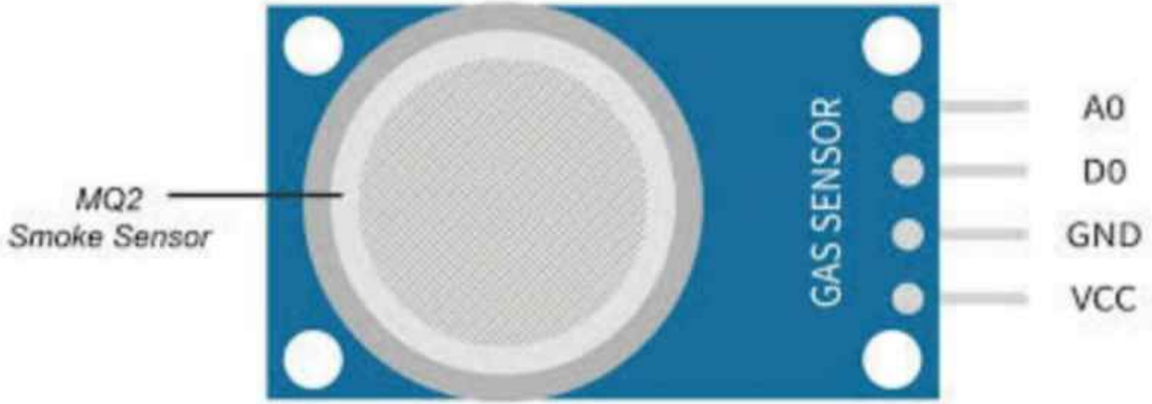
- पॉइंट लेवल सेंसर (Point Level Sensor):
- कंटिन्युअस लेवल सेंसर (Continuous Level Sensor):





8: स्मोक और गैस सेंसर (Smoke and Gas Sensor):

- स्मोक और गैस सेंसर ऐसे उपकरण हैं जो हवा में धुआं (स्मोक) या हानिकारक गैसों की उपस्थिति का पता लगाते हैं और इसकी जानकारी देते हैं। इन सेंसरों का उपयोग सुरक्षा के लिए किया जाता है ताकि धुआं, जहरीली गैस, या आग जैसी आपातकालीन स्थितियों के बारे में समय रहते अलार्म मिल सके।
- Smoke and gas sensors are devices that detect and report the presence of smoke or harmful gases in the air. These sensors are used for safety purposes to provide timely alarms about emergency situations such as smoke, toxic gases, or fire.



9: ह्यूमिडिटी सेंसर (Humidity Sensor):

- ह्यूमिडिटी सेंसर एक ऐसा उपकरण है जो वातावरण में मौजूद नमी (ह्यूमिडिटी) की मात्रा को मापता है। इसे "नमी सेंसर" भी कहा जाता है। यह सेंसर हवा में पानी के वाष्प (वॉटर वेपर) की मात्रा का पता लगाता है और इसे इलेक्ट्रिकल सिग्नल में बदल देता है, जिससे हम नमी की सही माप ले सकते हैं। इसका उपयोग मौसम विज्ञान, कृषि, घरों और उद्योगों में किया जाता है।
- A humidity sensor is a device that measures the amount of moisture present in the atmosphere. It is also called a "moisture sensor". This sensor detects the amount of water vapor in the air and converts it into an electrical signal, allowing us to take accurate measurements of humidity. It is used in meteorology, agriculture, homes and industries.

एक्चुएटर (Actuator):

- "Actuator एक इलेक्ट्रो-मैकेनिकल डिवाइस होता है जो इलेक्ट्रिकल सिग्नल को मैकेनिकल मूवमेंट (गति, धक्का, घुमाव, खिंचाव) में बदलता है।"
- "An actuator is an electro-mechanical device that converts electrical signals into mechanical movement (pulling, pushing, twisting, pulling)."

2. Actuator का कार्य (Function of Actuator)

- Controller या Sensor से सिग्नल प्राप्त करता है
- Receives signal from controller or sensor
- उस सिग्नल के अनुसार फिजिकल मूवमेंट करता है
- Performs physical movement according to that signal
- मशीन या सिस्टम में Output क्रिया करता है (जैसे पंखा चलाना, गेट खोलना)
- Performs output action in the machine or system (eg running a fan, opening a gate)

एक्चुएटर के प्रकार (Types of Actuator):

(1): इलेक्ट्रिक एक्चुएटर (Electric Actuator):

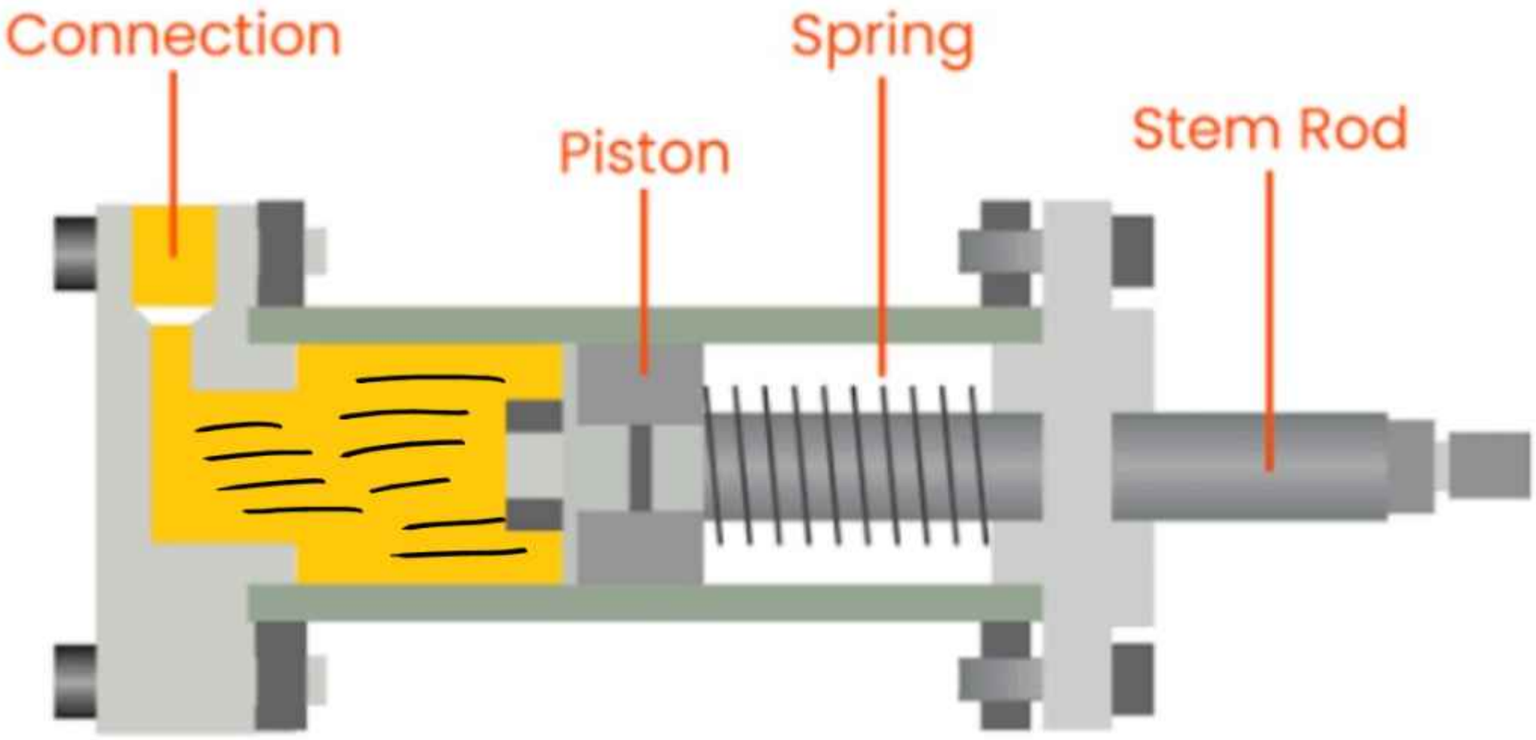
- यह एक्चुएटर बिजली (इलेक्ट्रिसिटी) का उपयोग करके काम करता है। जब इसे बिजली का सिग्नल मिलता है, तो यह एक मोटर या सोलनॉइड की मदद से आवश्यक गति या क्रिया उत्पन्न करता है।
- This actuator works using electricity. When it receives an electrical signal, it produces the required motion or action with the help of a motor or solenoid.
- उपयोग: यह रोबोटिक्स, ऑटोमेशन सिस्टम, और मशीनों में गति या स्थिति को नियंत्रित करने के लिए उपयोग होता है।
- Uses: It is used to control motion or position in robotics, automation systems, and machines.

(2): हाइड्रोलिक एक्चुएटर (Hydraulic Actuator):

- यह एक्चुएटर तरल पदार्थ (हाइड्रोलिक फ्लुइड) के दबाव का उपयोग करके काम करता है। जब इसे सिग्नल मिलता है, तो यह हाइड्रोलिक फ्लुइड को एक पिस्टन में भेजता है, जिससे बल उत्पन्न होता है और आवश्यक क्रिया होती है।
- This actuator works by using the pressure of the fluid (hydraulic fluid). When it receives a signal, it sends hydraulic fluid to a piston, producing force and performing the required action.

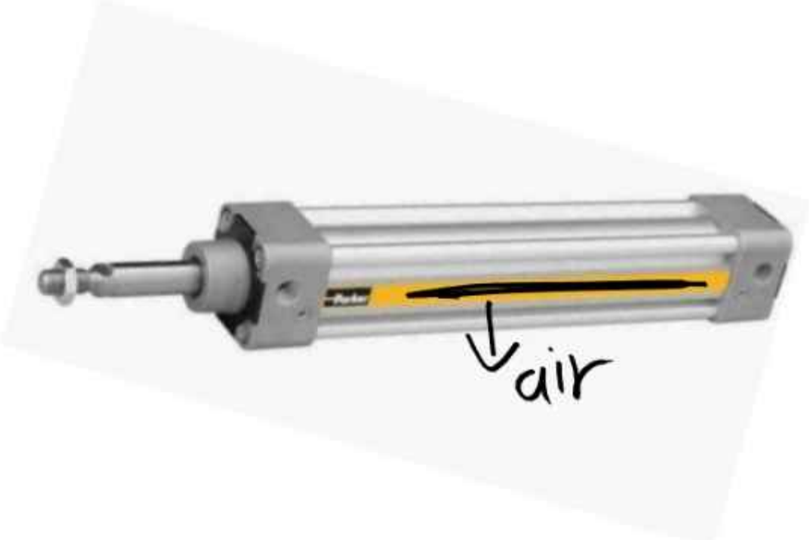
- उपयोग: भारी मशीनरी, क्रेन, और निर्माण उपकरणों में इसका उपयोग होता है
- Uses: Used in heavy machinery, cranes, and construction equipment

(2): हाइड्रोलिक एक्चुएटर (Hydraulic Actuator):



3. न्यूमैटिक एक्चुएटर (Pneumatic Actuator):

- यह एक्चुएटर संपीड़ित हवा (कंप्रेस्ड एयर) का उपयोग करके काम करता है। हवा का दबाव एक पिस्टन को चलाता है, जिससे गति या क्रिया होती है।
- This actuator works by using compressed air. The air pressure drives a piston, causing movement or action.
- उपयोग: इसका उपयोग हल्के कार्यों के लिए किया जाता है, जैसे स्वचालित दरवाजे, प्रोडक्शन लाइन की मशीनें, आदि।
- Uses: It is used for light applications, like automatic doors, production line machines, etc.



Sensor vs Actuator:

बिंदु (Points)	बिंदु (Points)	Actuator (एक्चुएटर)
परिभाषा (Definition)	वातावरण से डेटा को मापता/महसूस करता है	किसी आदेश या डेटा के आधार पर फिजिकल क्रिया करता है
कार्य (Function)	Input प्रदान करता है (जैसे तापमान मापना)	Output देता है (जैसे दरवाज़ा खोलना, मोटर चलाना)
प्रकार (Types)	Temperature, Light, Motion, Gas, Humidity आदि	Electric, Hydraulic, Pneumatic, Thermal आदि
इनपुट/आउटपुट	Input डिवाइस है	Output डिवाइस है
उदाहरण (Example)	DHT11 (Temperature), IR Sensor, LDR	DC Motor, Servo Motor, Relay, Solenoid

डाटा का रूप	Analog या Digital Data को भेजता है	Received Signal को Physical Action में बदलता है
उपयोग (Use)	वातावरण को समझने के लिए	मशीनों को क्रियाशील बनाने के लिए
दिशा (Direction)	Environment → Processor/Controller	Controller → मशीन या डिवाइस
Role in IoT	Data Collection	Action Execution
ऊर्जा खपत (Power)	कम होती है (Passive Component भी हो सकता है)	अधिक हो सकती है (Active Component होता है)

Chapter No-2 (IoT Devices)

Q 1: What is a bread board? Also show the connections with a clear diagram.
Q 1: Bread board क्या है? स्पष्ट चित्र के द्वारा Connections को भी दर्शाइये।

Q 2: What are Tri-color and Bi-color LEDs? Explain.
Q 2:Tri-color तथा Bi color LED क्या है? समझाइये।

Q 3: What is the difference between Sensors and Actuators?
Q 3:Sensor एवं Actuators में क्या अन्तर है?

Q 4: What is a sensor? Explain its different types?
Q 4: Sensor क्या है? इसके विभिन्न प्रकारों की व्याख्या कीजिए।

Q 5 What is a Actuator ? Explain its different types?
Q 5:Tri-color तथा Bi color LED क्या है? समझाइये।

Chapter No: 1 Introduction to (IoT)

syllabus:

Introduction to the components of basic IoT networks, the types of network connections and how data travels through them, and the role of Internet Protocols. Basic under standing of micro controllers/ Arduino and communication protocols

Introduction to the components of basic IoT Network:
(बुनियादी IoT नेटवर्क के घटकों का परिचय):

- An IoT (Internet of Things) network allows different devices to connect through the internet and exchange data. A basic IoT network has four main components:
- IoT (Internet of Things) नेटवर्क का मतलब है कि हम इंटरनेट के ज़रिए अलग-अलग डिवाइसेज को कनेक्ट करके आपस में डेटा का आदान-प्रदान कर सकते हैं। एक बेसिक IoT नेटवर्क में मुख्यत: 4 कंपोनेंट्स होते हैं:

(1) Devices:

- ये वे स्मार्ट डिवाइसेज होते हैं जो इंटरनेट से कनेक्ट होते हैं, जैसे सेंसर, स्मार्टफ़ोन, स्मार्ट लाइट्स आदि। ये डिवाइसेज डेटा को कलेक्ट करते हैं और आगे भेजते हैं।
- These are the smart devices that connect to the internet, such as sensors, smartphones, smart lights, etc. These devices collect data and send it for processing.

2. नेटवर्क (Network):

- नेटवर्क वह माध्यम है जो डिवाइसेज और सर्वर के बीच कनेक्शन बनाता है। इसमें WiFi, Bluetooth, Cellular, आदि शामिल हो सकते हैं, जो डेटा को डिवाइसेज से क्लाउड या सर्वर तक पहुंचाते हैं। 4G, 5G,
- The network is the medium that connects devices to the server. This can include WiFi, Bluetooth, Cellular, and more, which helps in transferring data from the devices to the cloud or server.

3. क्लाउड (Cloud):

- क्लाउड एक सर्वर होता है जहाँ डिवाइसेज से आने वाला डेटा स्टोर और प्रोसेस होता है। इससे डेटा को रियल टाइम में एक्सेस करना और प्रोसेस करना आसान हो जाता है।
- The cloud is a server where the data from the devices is stored and processed. It makes it easy to access and analyze data in real-time.

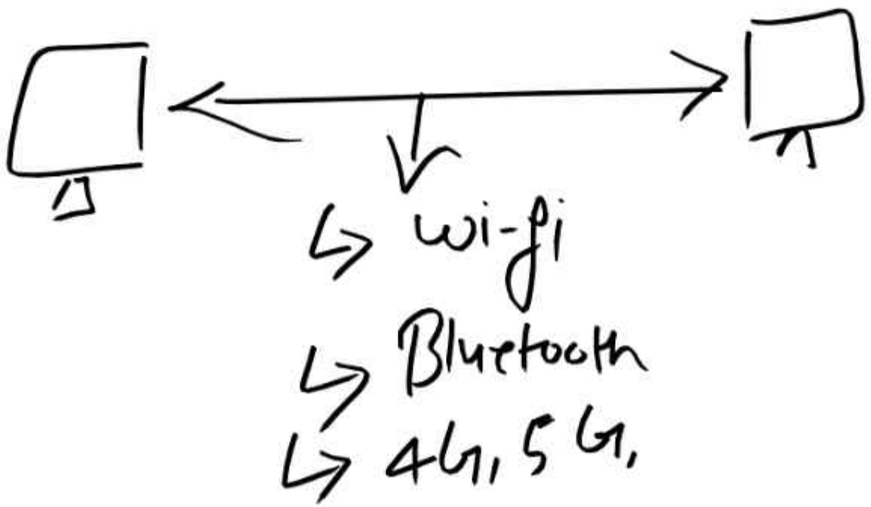
→ 24X7 → any device → Remote access.

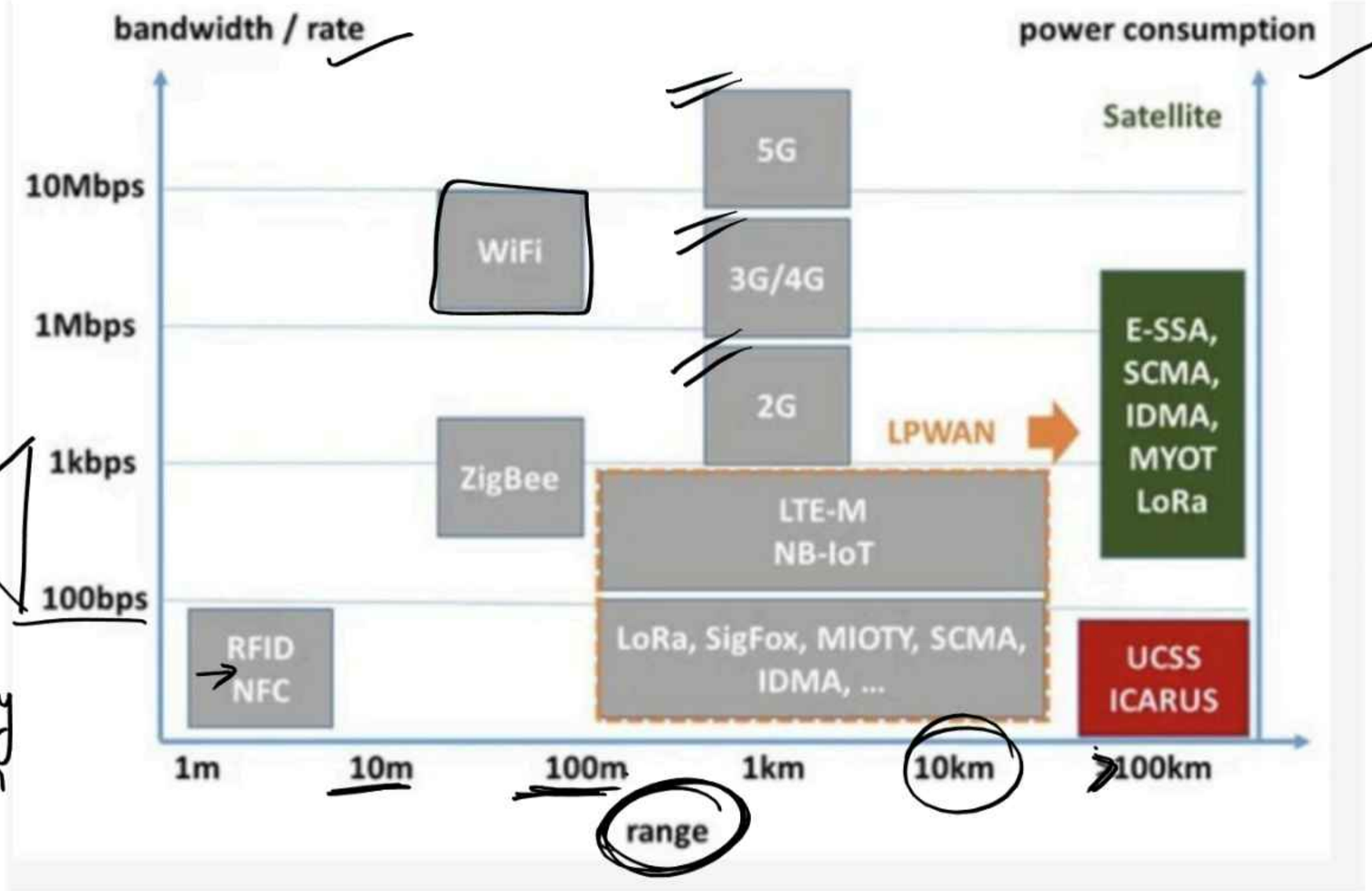
4. Application:

- एप्लिकेशन का काम होता है यूज़र को डिवाइसेज का डेटा समझना और उन्हें कंट्रोल करना। उदाहरण के लिए, स्मार्टफ़ोन ऐप्स के ज़रिए हम घर के डिवाइसेज को कंट्रोल कर सकते हैं।
- The application allows users to understand and control the data from the devices. For example, through smartphone apps, we can control smart home devices.

नेटवर्क कनेक्शन के प्रकार (Type of network connections):

- IoT नेटवर्क में डेटा को एक जगह से दूसरी जगह भेजने के लिए कई प्रकार के नेटवर्क कनेक्शन होते हैं। आइए हम कुछ मुख्य प्रकार के नेटवर्क कनेक्शन को समझते हैं और कैसे डेटा इनमें यात्रा करता है
- an IoT network, data travels from one place to another using various types of network connections. Let's go through some main types of network connections and understand how data travels through them





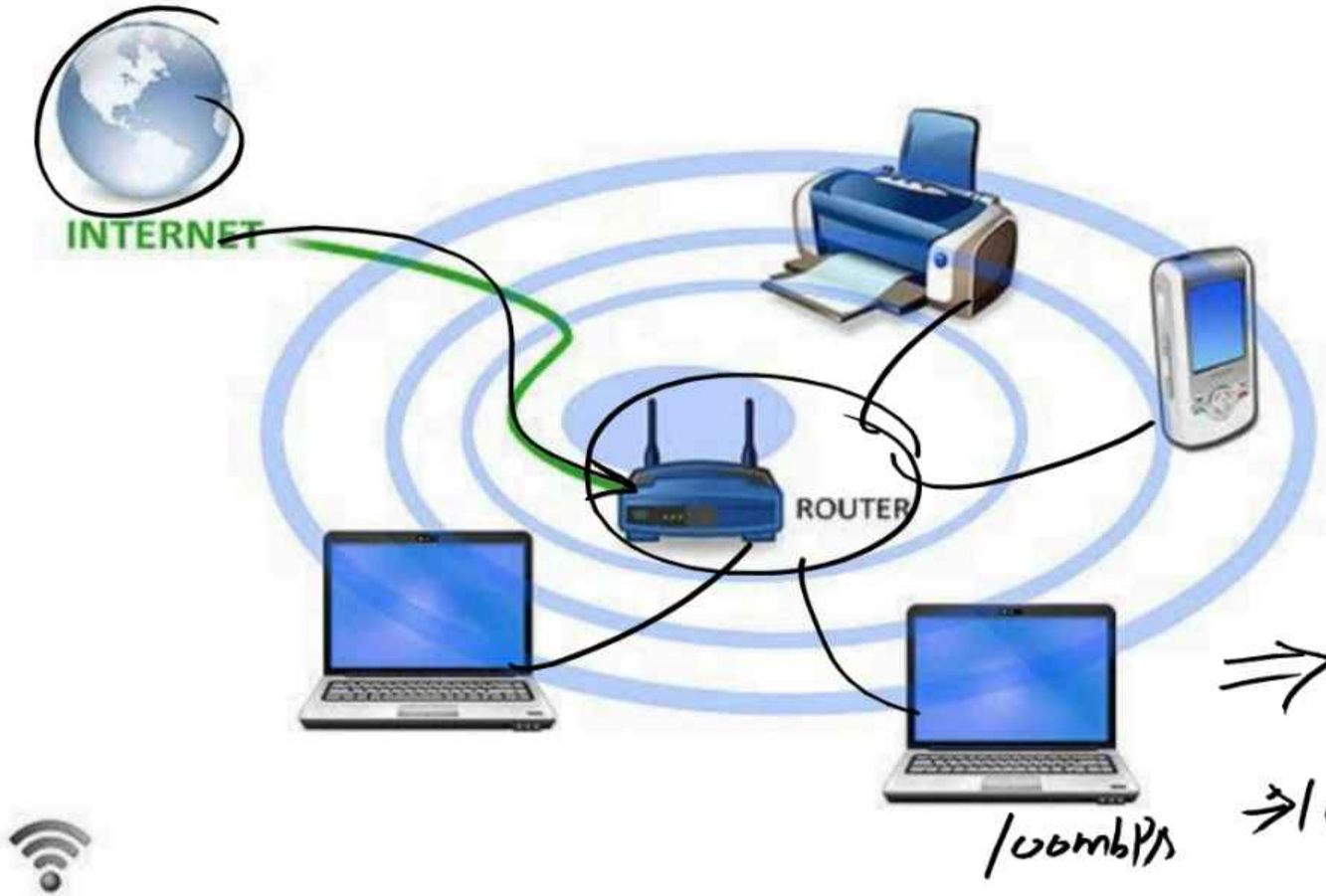
Radio frequency identification

(1) WiFi :

- वाई-फाई नेटवर्क आपके घर, ऑफिस, या पब्लिक जगहों पर वायरलेस तरीके से इंटरनेट कनेक्शन उपलब्ध कराता है। एक राउटर (Router) होता है जो इंटरनेट से जुड़ा होता है और डिवाइसेज को इंटरनेट से कनेक्ट करता है।
- WiFi provides a wireless internet connection in homes, offices, or public areas. A router connects to the internet and links various devices to it.

डेटा flow:

- जब कोई डिवाइस (जैसे स्मार्टफोन या कंप्यूटर) वाई-फाई से कनेक्ट होता है, तो डेटा उस डिवाइस से राउटर तक जाता है और फिर इंटरनेट के माध्यम से सर्वर तक पहुँचता है।
- When a device (such as a smartphone or computer) connects to Wi-Fi, data travels from that device to the router and then to a server via the Internet.



⇒ 10m → 100m

⇒ 10kbps →

(2) Bluetooth :

- ब्लूटूथ एक शॉर्ट-रेंज वायरलेस कनेक्शन है, जो आमतौर पर मोबाइल डिवाइसेज, हेडफ़ोन और अन्य गैजेट्स को जोड़ने के लिए उपयोग किया जाता है।
- Bluetooth is a short-range wireless connection, commonly used to connect mobile devices, headphones, and other gadgets.

डेटा flow:

- ब्लूटूथ डेटा को सीधे जुड़े डिवाइसेज के बीच भेजता है, जैसे कि आपका मोबाइल और हेडफ़ोन। इसमें इंटरनेट की ज़रूरत नहीं होती।
- Bluetooth sends data directly between connected devices, like your phone and headphones, without needing the internet.

(3) Low Power Wide Area Network (LPWAN):

- एक वायरलेस नेटवर्क है जो IoT डिवाइसेज के लिए डिज़ाइन किया गया है। इसका उपयोग लंबी दूरी तक कम ऊर्जा में छोटे डेटा पैकेट्स भेजने के लिए किया जाता है। यह बैटरी से चलने वाले सेंसर जैसे डिवाइसेज को लंबे समय तक काम करने में मदद करता है।
- Is a wireless network designed for IoT devices. It is used to send small data packets over long distances with minimal power consumption, allowing battery-operated sensors and devices to work for extended periods.

Key Features:

1. Low Power Consumption – Batteries can last for several years.
2. Long Range – Reaches remote areas.
3. Low Data Rate – Suitable for small data transfers.

100m → 10Km
↳ Data → 200KB

1. कम ऊर्जा खपत – बैटरी कई सालों तक चल सकती है।
2. लंबी दूरी – दूरस्थ स्थानों तक डेटा पहुंचाता है।
3. कम डेटा रेट – छोटे डेटा ट्रांसफर के लिए उपयुक्त।

(4) LoRaWAN (LoRa Wide Area Network):

- लोराWAN लंबी दूरी के लिए लो पावर (कम ऊर्जा) कनेक्शन प्रोवाइड करता है। यह विशेषकर IoT डिवाइसेज के लिए बनाया गया है, जैसे कि स्मार्ट सिटी में सेंसर।
- LoRaWAN provides a long-range, low-power connection, especially for IoT devices like sensors in smart cities.

Data flow:

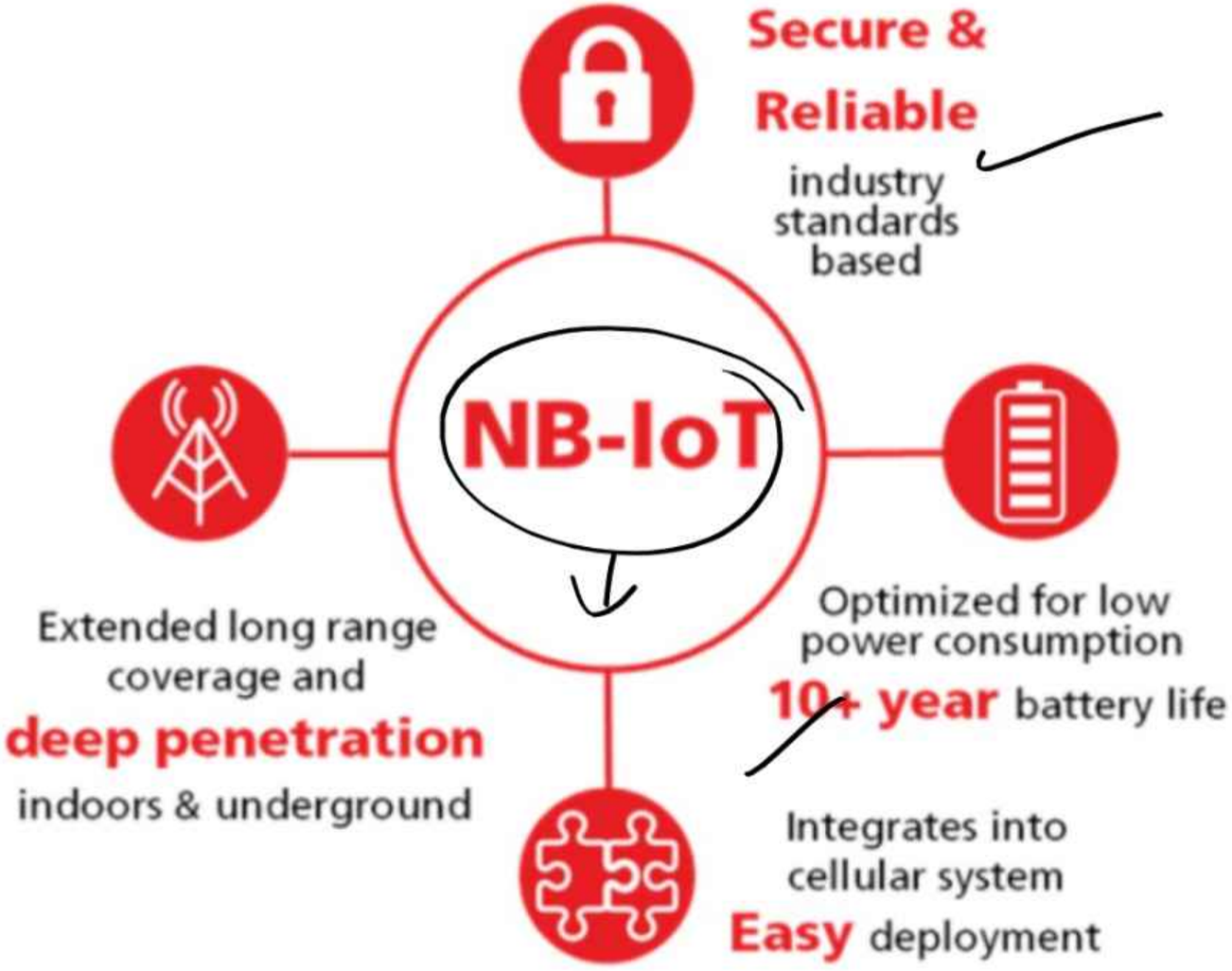
- यह भी सेलुलर नेटवर्क की तरह काम करता है, जहाँ डेटा पहले बेस स्टेशन तक पहुँचता है और फिर इंटरनेट के माध्यम से सर्वर तक।
- In a LoRa network, data goes from a device to a LoRa gateway, and then to the server. It’s used for low data rate connections in remote areas.

(5) NB-IoT (Narrowband IoT):

- एनबी-आईओटी, लो पावर और वाइड एरिया कनेक्शन है जो विशेष रूप से IoT के लिए डिज़ाइन किया गया है। यह छोटे डेटा पैकेट्स को लंबे समय तक भेजने में सहायक है।
- NB-IoT is a low-power, wide-area connection specifically designed for IoT. It is useful for sending small data packets over a long period.

Data flow:

- यह भी सेलुलर नेटवर्क की तरह काम करता है, जहाँ डेटा पहले बेस स्टेशन तक पहुँचता है और फिर इंटरनेट के माध्यम से सर्वर तक।
- NB-IoT works like a cellular network, where data travels first to a base station and then to the server through the internet.


Secure &
Reliable
industry
standards
based
NB-IoT
Extended long range
coverage and
deep penetration
indoors & underground
Optimized for low
power consumption
10+ year battery life
Integrates into
cellular system
Easy deployment

IoT में संचार प्रौद्योगिकी (Communication Technology in IoT):

- IoT (Internet of Things) में अलग-अलग डिवाइसेज एक-दूसरे से कनेक्ट होते हैं और डेटा का आदान-प्रदान करते हैं। इसके लिए कई तरह की कम्युनिकेशन टेक्नोलॉजीज का इस्तेमाल होता है, जिनका चुनाव डेटा ट्रांसफर की दूरी, स्पीड, पावर और एप्लिकेशन के अनुसार किया जाता है।
- In IoT (Internet of Things), various devices connect and exchange data. Different communication technologies are used based on factors like data transfer distance, speed, power, and the application needs.

1. WiFi (वाई-फाई):

- वाई-फाई एक वायरलेस नेटवर्क है जो डिवाइसेज को इंटरनेट से कनेक्ट करता है। इसका इस्तेमाल घर, ऑफिस, और स्मार्ट होम डिवाइसेज में किया जाता है।
- WiFi is a wireless network that connects devices to the internet. It's commonly used in homes, offices, and for smart home devices.
- Data Speed : 600 Mbps maximum, but 150-200 Mbps is more typical,
- Speed: 2.4 GHz and 5 GHz bands
- Range: About ~~50 m~~ 100m
- डेटा दरें: 600 एमबीपीएस अधिकतम, लेकिन 150-200 एमबीपीएस अधिक विशिष्ट है,
- आवृत्ति: 2.4 गीगाहर्ट्ज और 5 गीगाहर्ट्ज बैंड
- रेंज: लगभग 50 मी0

2. Bluetooth (ब्लूटूथ):

- ब्लूटूथ शॉर्ट रेंज वायरलेस टेक्नोलॉजी है, जो आमतौर पर मोबाइल और स्मार्ट डिवाइसेज को कनेक्ट करने के लिए इस्तेमाल होती है।
- Bluetooth is a short-range wireless technology often used to connect mobile and smart devices.
  - Frequency (आवृत्ति): 2.4 GHz (ISM)
  - Range (रेंज) : 50-150 M (Smart/BLE)
  - Data rates (डेटा दरें) : 1 Mbps (Smart/BLE)

3. Zigbee (ज़िगबी):

- Zigbee एक कम पावर और लो डेटा रेट वायरलेस टेक्नोलॉजी है, जो स्मार्ट होम, इंडस्ट्रियल ऑटोमेशन और सेंसर नेटवर्क्स में इस्तेमाल होती है।
- Zigbee is a low-power, low-data-rate wireless technology used in smart homes, industrial automation, and sensor networks.
- Standard (मानक:): ZigBee 3.0 based on IEEE802.15.4
- Frequency (आवृत्ति): 2.4 GHz
- Range (रेंज): 10-100 m

ZigBee®

4. LoRaWAN (लोरा वैन):

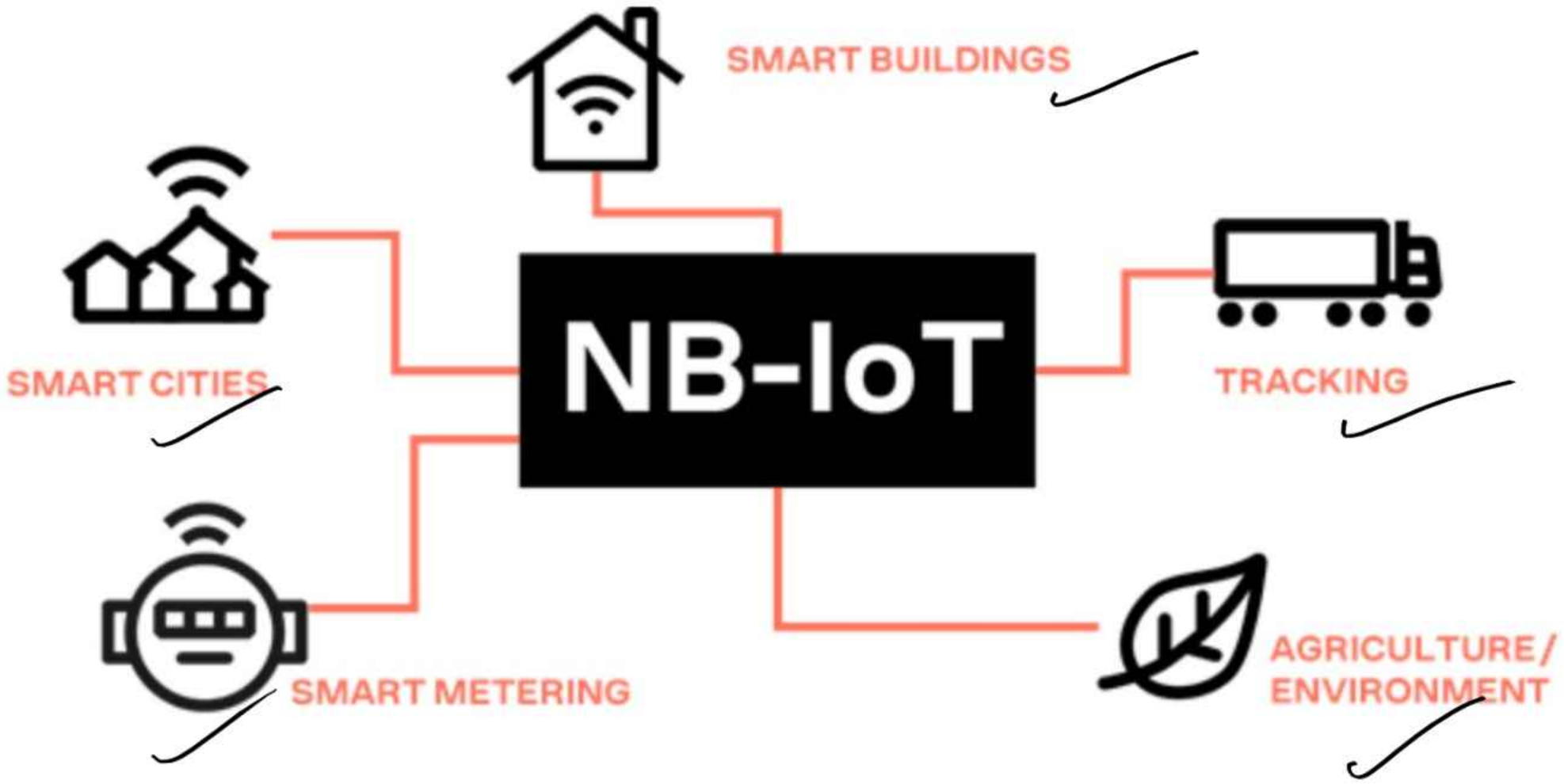
- LoRaWAN लंबी दूरी के लिए कम पावर का उपयोग करता है, और इसे खासकर IoT के लिए डिज़ाइन किया गया है। इसका इस्तेमाल स्मार्ट सिटी और कृषि में किया जाता है।
- LoRaWAN provides low-power, long-range connectivity and is designed specifically for IoT. It's used in applications like smart cities and agriculture.

- Standard(मानक): LoRaWAN
- Frequency (आवृत्ति): Various
- Range (रेंज): 2-5 km (urban environment), 15 km (suburban environment)
- Data rates(डेटा दरें) : 0.3-50 kbps.

5. NB-IoT (Narrowband IoT):

- NB-IoT एक सेल्युलर नेटवर्क पर आधारित तकनीक है, जो IoT डिवाइसेज को लंबी दूरी तक जोड़ता है। यह टेलीकॉम कंपनियों द्वारा प्रदान किया जाता है। लंबी दूरी और अच्छी नेटवर्क कवरेज, बैटरी से चलने वाले डिवाइसेज के लिए उपयुक्त।
- NB-IoT is a cellular-based technology that connects IoT devices over long distances. It's provided by telecom companies. Long-range and good network coverage, suitable for battery-operated devices.

- Various Frequencies: Typically operates on licensed frequency bands, such as 700 MHz, 800 MHz, and 900 MHz.
- Range:
  - 10-15 km in urban environments
  - Up to 35 km in rural areas
- Data Rates :Data Rates: 20-250 kbps


SMART BUILDINGS
SMART CITIES
NB-IoT
TRACKING
SMART METERING
AGRICULTURE / ENVIRONMENT

6. Cellular (सेल्युलर - 2G, 3G, 4G, 5G):

- यह मोबाइल नेटवर्क है जो डिवाइसेज को इंटरनेट से जोड़ता है। 5G आने से IoT में डेटा ट्रांसफर बहुत तेज़ हो गया है।हाई स्पीड डेटा ट्रांसफर और लंबी दूरी तक कनेक्टिविटी।
- Cellular networks connect devices to the internet. With 5G, data transfer in IoT has become much faster. High-speed data transfer and long-range connectivity.

- **Standard:** GSM/GPRS/EDGE (2G), UMTS/HSPA (3G), LTE (4G)
- **Frequency:** 900/1800/1900/2100 MHz
- **Range:** 35 km max for GSM; 200 km max for HSPA
- **Data rates (typical download):**
  - 35-170 kps (GPRS), 120-384 kbps (EDGE), 384 kbps-2 Mbps (UMTS), 600 kbps-10 Mbps (HSPA), 3-10 Mbps (LTE)

7: RFID (Radio Frequency Identification):

- RFID टेक्नोलॉजी टैग्स का उपयोग करती है, जिन्हें RFID रीडर्स पढ़ते हैं। इसका इस्तेमाल ट्रैकिंग और आइडेंटिफिकेशन में होता है, जैसे कि मॉल्स या वेयरहाउस में।
- RFID technology uses tags that can be read by RFID readers. It's used for tracking and identification, like in malls or warehouses.

- फ़ायदे: कम पावर और सटीक ट्रैकिंग।
- सीमाएँ: रेंज बहुत कम होती है, और केवल पहचान के लिए उपयोगी है।
- Advantages: Low power and precise tracking.
- Limitations: Very short range, mainly useful for identification.

8. SigFox:-

- Sigfox एक लो-पावर वाइड एरिया नेटवर्क (LPWAN) तकनीक है, जो विशेष रूप से IoT (इंटरनेट ऑफ थिंग्स) डिवाइसों के लिए डिज़ाइन की गई है। यह छोटी मात्रा में डेटा को लंबी दूरी पर ट्रांसमिट करने में सक्षम है।
- Sigfox is a Low Power Wide Area Network (LPWAN) technology specifically designed for IoT (Internet of Things) devices. It enables the transmission of small amounts of data over long distances.

Sigfox की विशेषताएँ (Key Features of Sigfox):

- **Low Power Consumption:**
- **Long Range:** 10 to 50 kilometers, especially in rural areas.
- **Low Data Traffic transfer Rate**: Sigfox has a data limit of 12-140 bytes per message
- **Direct Connectivity:**

Bluetooth और WiFi के बीच अंतर (Difference Between Bluetooth and WiFi)

- Bluetooth का कोई full form नहीं है।
- कनेक्टिविटी के लिए इसे सभी उपकरणों पर Bluetooth एडाप्टर की आवश्यकता होती है
- Bluetooth कम बिजली की खपत करता है।
- WiFi की तुलना में Bluetooth की सुरक्षा कम है।
- इसे सीमित उपयोगकर्ता उपयोग कर सकते हैं।
- Bluetooth की रेडियो सिग्नल रेंज दस मीटर है।
- Bluetooth को कम बैंडविड्थ की आवश्यकता होती है।

- जबकि WiFi की full form वायरलेस फिडेलिटी है।
- जबकि इसे कनेक्टिविटी के लिए सभी उपकरणों पर वायरलेस एडेप्टर और वायरलेस राउटर की आवश्यकता होती है
- जबकि यह उच्च शक्ति का उपभोग करता है।
- जबकि यह Bluetooth से बेहतर सुरक्षा प्रदान करता है।
- जबकि WiFi को बड़ी मात्रा में उपयोगकर्ता उपयोग कर सकते हैं।
- जबकि WiFi में यह रेंज सौ मीटर है।
- जबकि इसके लिए उच्च बैंडविड्थ की आवश्यकता होती है।

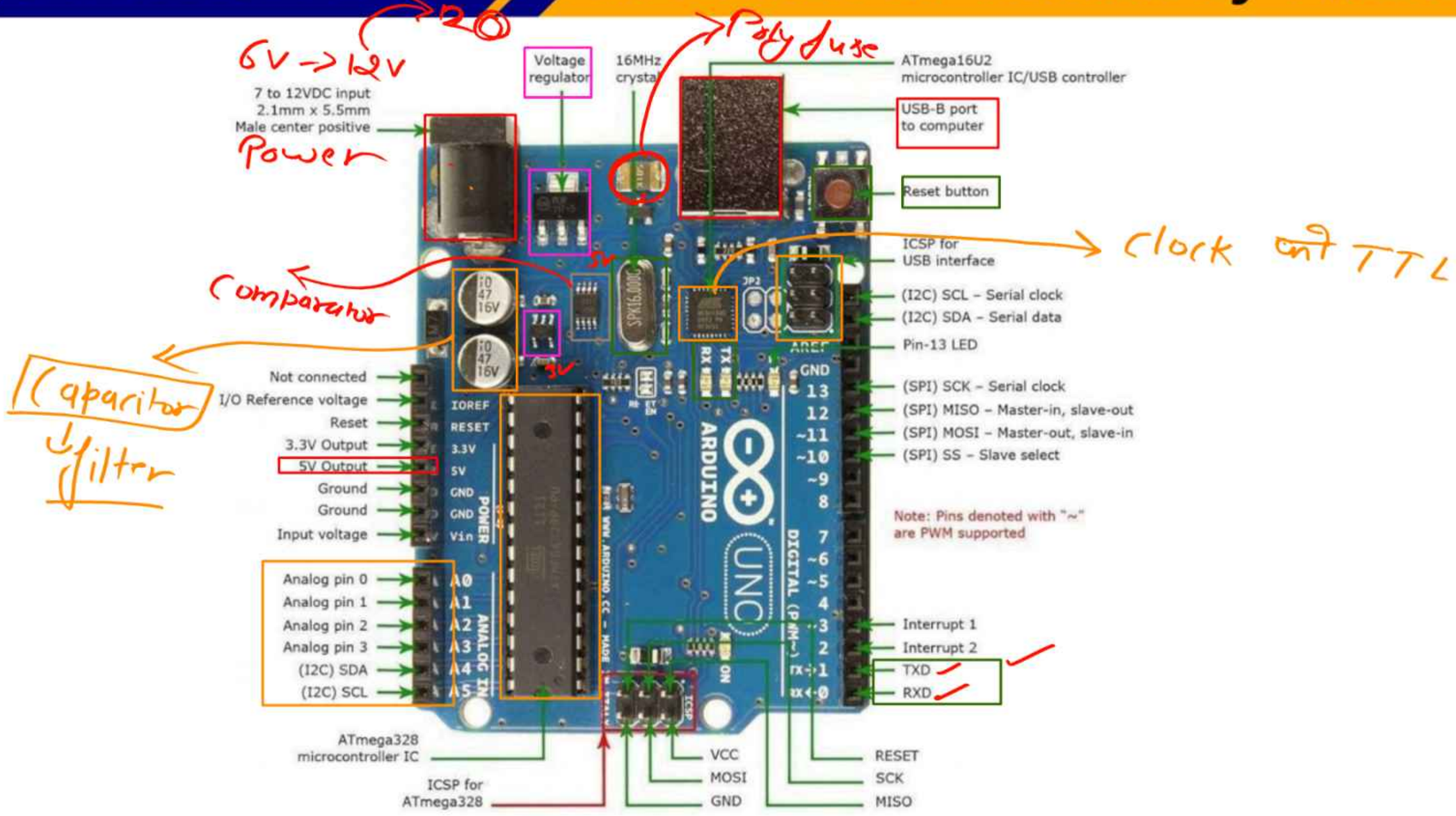
Bluetooth और WiFi के बीच अंतर (Difference Between Bluetooth and WiFi

There is no full form of Bluetooth.	the full form of WiFi is Wireless Fidelity.
It requires Bluetooth adapter on all devices for connectivity.	While it requires wireless adapters and wireless routers on all devices for connectivity.
Bluetooth consumes less power.	While it consumes high power.
Bluetooth is less secure than WiFi.	While it provides better security than Bluetooth.
It can be used by limited users.	While WiFi can be used by a large number of users.
The radio signal range of Bluetooth is ten meters.	While the range in WiFi is hundred meters.
Bluetooth requires less bandwidth.	While it requires high bandwidth.

Basic under standing of micro controllers/ Arduino:

(माइक्रोकंट्रोलर/ Arduino की बुनियादी समझ):

- Arduino एक ओपन-सोर्स प्लेटफॉर्म है, जो इलेक्ट्रॉनिक प्रोजेक्ट्स बनाने में मदद करता है। यह एक माइक्रोकंट्रोलर बोर्ड (जैसे Arduino Uno) और इसके साथ एक प्रोग्रामिंग वातावरण (Arduino IDE) से बना होता है। (C/C++)
- Arduino is an open-source platform that helps create electronic projects. It consists of a microcontroller board (such as the Arduino Uno) and a programming environment (Arduino IDE) that comes with it.
- माइक्रोकंट्रोलर एक छोटा कंप्यूटर होता है, जिसमें प्रोसेसर, मेमोरी, और इनपुट/आउटपुट (I/O) पिन होते हैं। ये साधारण कार्यों को नियंत्रित करने के लिए डिज़ाइन किए गए होते हैं, जैसे सेंसर डेटा पढ़ना, मोटर कंट्रोल करना, और डिस्प्ले को संचालित करना।
- A microcontroller is a small computer that contains a processor, memory, and input/output (I/O) pins. They are designed to perform simple tasks, such as reading sensor data, controlling motors, and operating displays.


6V -> 12V
7 to 12VDC input
2.1mm x 5.5mm
Male center positive
Power
Voltage regulator
16MHz crystal
Poly fuse
ATmega16U2 microcontroller IC/USB controller
USB-B port to computer
Reset button
ICSP for USB interface
Clock को TTL
Comparator
Capacitor
filter
(I2C) SCL – Serial clock
(I2C) SDA – Serial data
Pin-13 LED
(SPI) SCK – Serial clock
(SPI) MISO – Master-in, slave-out
(SPI) MOSI – Master-out, slave-in
(SPI) SS – Slave select
Note: Pins denoted with "~" are PWM supported
Interrupt 1
Interrupt 2
TXD
RXD
Not connected
I/O Reference voltage
Reset
3.3V Output
5V Output
Ground
Ground
Input voltage
Analog pin 0
Analog pin 1
Analog pin 2
Analog pin 3
(I2C) SDA
(I2C) SCL
ATmega328 microcontroller IC
ICSP for ATmega328
VCC
MOSI
GND
RESET
SCK
MISO

(1.) USB Port (Number 1):

- यह पोर्ट Arduino को कंप्यूटर से कनेक्ट करने के लिए होता है। इसके जरिए हम Arduino को पावर देते हैं और कोड अपलोड करते हैं।
- This port is used to connect the Arduino to a computer. It provides power to the board and allows us to upload code to it.

(2.) Power Jack (Number 2):

- Arduino को बाहरी स्रोत से पावर देने के लिए इस्तेमाल होता है। यह 7-12 वोल्ट के DC पावर को सपोर्ट करता है।
- This is used to supply external power to the Arduino. It supports 7-12 volts of DC power.

(3.) 5V Pin (Number 3):

- यह पिन Arduino से 5V का आउटपुट देता है और सेंसर, LED आदि को पावर देने के लिए इस्तेमाल होता है।
- This pin provides a 5V output from the Arduino, used to power sensors, LEDs, etc.

(4.) 3.3V Pin (Number 4):

- कुछ डिवाइसेस 3.3V पर काम करते हैं, उन्हें पावर देने के लिए इस पिन का उपयोग होता है।
- Some devices operate at 3.3V, and this pin supplies that power.

(5.) GND Pins (Number 5):

- यह ग्राउंड पिन है, जो सभी सर्किट को ग्राउंडिंग प्रदान करता है।
- These are ground pins that provide grounding for all the circuits connected to the Arduino.

(6.) Analog In Pins (Number 6):

- ये पिन्स (A0 से A5 तक) एनालॉग सेंसर से इनपुट प्राप्त करने के लिए होते हैं। इन पिन्स से हम एनालॉग सिग्नल (जैसे तापमान, प्रकाश) को पढ़ सकते हैं।
- These pins (A0 to A5) are used to receive input from analog sensors. Through these, we can read analog signals (like temperature, light).

(7.) Digital Pins (Number 7-13):

- ये पिन्स डिजिटल इनपुट और आउटपुट के लिए उपयोग होते हैं। इन पिन्स पर हम सेंसर, मोटर, और अन्य इलेक्ट्रॉनिक डिवाइसेस को कनेक्ट कर सकते हैं।
- These pins are for digital input and output. Sensors, motors, and other electronic devices can be connected to these pins.

(8.) PWM Pins (Number 8):

- कुछ डिजिटल पिन्स (जैसे 3, 5, 6, 9, 10, और 11) PWM आउटपुट देते हैं, जिनका उपयोग LED की ब्राइटनेस कंट्रोल करने और मोटर की स्पीड कंट्रोल करने के लिए होता है।
- Some digital pins (such as 3, 5, 6, 9, 10, and 11) provide PWM (Pulse Width Modulation) output, used for controlling LED brightness and motor speed.

(9.) RESET Button (Number 10):

- इस बटन को दबाने से Arduino का प्रोग्राम रीसेट हो जाता है और फिर से शुरू होता है।
- Pressing this button resets the Arduino, restarting the code from the beginning.

(10.) Power LED (Number 11):

- यह LED तब जलती है जब Arduino को पावर मिल रही होती है। यह इंडिकेटर है कि बोर्ड चालू है।
- This LED lights up when the Arduino is receiving power, indicating that the board is on.

(11.) TX and RX LEDs (Number 12):

- ये LEDs डेटा ट्रांसमिशन (TX) और रिसेप्शन (RX) को इंडिकेट करती हैं। जब Arduino और कंप्यूटर के बीच डेटा का आदान-प्रदान हो रहा होता है, तो ये LEDs ब्लिंक करती हैं।
- These LEDs indicate data transmission (TX) and reception (RX). They blink when data is being exchanged between the Arduino and the computer.

(12.) Microcontroller (Number 13):

- यह Arduino का मुख्य हिस्सा है। Arduino Uno में ATmega328P माइक्रोकंट्रोलर होता है, जो सारे कार्यों को प्रोसेस करता है।
- This is the main part of the Arduino. The Arduino Uno uses the ATmega328P microcontroller, which processes all the tasks.

(13) ICSP Header (Number 14):

- यह प्रोग्रामिंग के लिए होता है और माइक्रोकंट्रोलर को सीधा प्रोग्राम करने में मदद करता है।
- This is for programming the microcontroller directly and is useful for advanced programming.

संचार प्रोटोकॉल (Communication Protocols):

- IoT (Internet of Things) में विभिन्न डिवाइसेस को कनेक्ट करने और डेटा का आदान-प्रदान करने के लिए कुछ खास प्रकार के प्रोटोकॉल्स का उपयोग किया जाता है। हर प्रोटोकॉल का अपना उद्देश्य और उपयोग होता है।
- In IoT (Internet of Things), various types of protocols are used to connect devices and exchange data. Each protocol has a specific purpose and use case.

- नेटवर्क प्रोटोकॉल्स (Network Protocols)
- डेटा ट्रांसफर प्रोटोकॉल्स (Data Transfer Protocols
- सेल्युलर प्रोटोकॉल्स (Cellular Protocols)
- इन्फ्रास्ट्रक्चर प्रोटोकॉल्स (Infrastructure Protocols)
- सुरक्षा प्रोटोकॉल्स (Security Protocols)

(1) नेटवर्क प्रोटोकॉल्स (Network Protocols):

- नेटवर्क प्रोटोकॉल्स का उपयोग IoT डिवाइसेस को नेटवर्क पर जोड़ने और डेटा ट्रांसफर करने के लिए होता है।
- Network protocols are used to connect IoT devices to a network and allow them to transmit data.

Examples:

Wi-Fi:

- छोटे और घरेलू IoT डिवाइसेस के लिए उपयोगी। इसका उपयोग स्मार्ट होम डिवाइसेस में अधिक होता है।
- Useful for small and home-based IoT devices. It is widely used in smart home devices.

Bluetooth:

- छोटे रेंज में कम पावर कनेक्शन के लिए उपयुक्त। स्मार्ट वॉच, फिटनेस बैंड जैसे डिवाइसेस में इसका उपयोग होता है।
- Suitable for short-range, low-power connections. It's commonly used in devices like smartwatches and fitness bands.

Zigbee:

- लो-पावर और छोटे क्षेत्र के IoT नेटवर्क्स के लिए, जैसे स्मार्ट लाइट्स और स्मार्ट लॉक्स।
- Designed for low-power, short-range IoT networks, like smart lights and smart locks.

LoRaWAN:

- लंबी दूरी के लिए, कम पावर में बड़े एरिया में डिवाइसेस को कनेक्ट करने के लिए, जैसे कृषि में उपयोग।
- For long-range connections, it's used to connect devices over large areas with low power, such as in agriculture.

(2.) डेटा ट्रांसफर प्रोटोकॉल्स (Data Transfer Protocols):

- डेटा ट्रांसफर प्रोटोकॉल्स का उपयोग IoT डिवाइसेस और सर्वर के बीच डेटा के आदान-प्रदान के लिए किया जाता है।
- These protocols enable the transfer of data between IoT devices and a server.

Example:

HTTP (HyperText Transfer Protocol):

- यह वेब पर डेटा भेजने के लिए एक सामान्य प्रोटोकॉल है। इसका उपयोग उन IoT प्रोजेक्ट्स में होता है जहां डेटा की विश्वसनीयता जरूरी होती है।
- A common protocol for transferring data over the web. It is used in IoT projects where data reliability is essential.

MQTT (Message Queuing Telemetry Transport):

- यह हल्का और तेज प्रोटोकॉल है, जो कम बैंडविड्थ और कम पावर डिवाइसेस के लिए होता है। इसका उपयोग सेंसर और छोटे IoT डिवाइसेस में होता है।
- A lightweight, fast protocol, ideal for low-bandwidth and low-power devices like sensors.

CoAP (Constrained Application Protocol):

- यह भी एक हल्का प्रोटोकॉल है और HTTP जैसा है, लेकिन छोटे और लो-पावर डिवाइसेस के लिए उपयुक्त है।
- Similar to HTTP but lighter, making it suitable for small and low-power devices.

WebSocket:

- यह रियल-टाइम कम्युनिकेशन के लिए होता है, जिसमें सर्वर और क्लाइंट के बीच तुरंत डेटा का आदान-प्रदान होता है।
- Used for real-time communication, allowing immediate data exchange between the server and the client.

(3) सेल्युलर प्रोटोकॉल्स (Cellular Protocols):

- Cellular Protocols वे तकनीकी नियम और मानक (rules and standards) होते हैं जिनके द्वारा मोबाइल डिवाइसेज़ (जैसे मोबाइल फोन, टैबलेट आदि) सेल्युलर नेटवर्क के माध्यम से एक-दूसरे से और इंटरनेट से संचार (communication) करते हैं।
- Cellular Protocols are the technical rules and standards by which mobile devices (such as mobile phones, tablets, etc.) communicate with each other and the Internet via cellular networks.

Example:

- (GSM) Global System for Mobile Communication (2G).
- (GMTS) Universal Mobile Telecommunication System.
- (LTE) Long Term Evolution (4G).
- 5G NR 5th Generation New Radio.

(4) इन्फ्रास्ट्रक्चर प्रोटोकॉल्स (Infrastructure Protocols):

- Infrastructure Protocols वे संचार प्रोटोकॉल होते हैं जो नेटवर्क की आधारभूत संरचना (infrastructure) को बनाने, नियंत्रित करने और बनाए रखने के लिए इस्तेमाल किए जाते हैं। ये प्रोटोकॉल नेटवर्क के रूटिंग, addressing, switching, और data transfer जैसे कार्यों में सहायता करते हैं।
- Infrastructure Protocols are communication protocols used to create, control, and maintain the network's infrastructure. These protocols assist in network functions such as routing, addressing, switching, and data transfer.

Example:

- IP, DHCP, DNS, ARP ,RARP etc.

(4) इन्फ्रास्ट्रक्चर प्रोटोकॉल्स (Infrastructure Protocols):

- Security Protocols वे नियम और मानक (rules and standards) होते हैं जो कंप्यूटर नेटवर्क, वेबसाइट्स, ईमेल, और अन्य डिजिटल संचार में डेटा को सुरक्षित (secure) रखने के लिए बनाए जाते हैं।
- Security protocols are rules and standards designed to keep data secure in computer networks, websites, email, and other digital communications.

Example:

- HTTPS, SSL etc.

Chapter No-3 (IoT Networks)

Q 1: What are the features of Arduino Uno board?
Q 1 आर्डिनो Uno board की विशेषतायें क्या हैं?

Q 2: Compare between IoT networks and wireless networks.
Q 2:IoT नेटवर्क और वायरलेस नेटवर्क के बीच तुलना करें।

Q 3: What is IOT Network? Explain different types of IOT Network Connections.
Q 3:IOT Network क्या है विभिन्न प्रकार के IOT Network Connections को समझाइये।

Q 4: Explain all internet protocols and all data protocols.
Q 4: सभी इंटरनेट प्रोटोकॉल तथा सभी डेटा प्रोटोकॉल की व्याख्या करें।

Q 5: Explain communication Technologies in IoT?
Q 5:IoT में संचार प्रौद्योगिकियों की व्याख्या करें

Chapter No-3 (IoT Networks)

Q 6: Difference between WiFi and Bluetooth.):

Q 6: वाईफाई और ब्लूटूथ के बीच अंतर।

3-unit complete

Chapter No. 4: (IoT Arduino

Syllabus :

- Feature of arduino device,
- Ardunino device introduction,
- Components of Arduino board ,
- C Arduino Programming Language,
- Understanding of basic of Arduino IDE,
- function, control statement, loops, datatype, variables: (Language)

Introduction of Arduino:

- Arduino is an open-source electronics platform that helps create projects using hardware and software. It is designed to program devices based on microcontrollers.
- Arduino एक ओपन-सोर्स इलेक्ट्रॉनिक्स प्लेटफ़ॉर्म है, जो हार्डवेयर और सॉफ़्टवेयर का उपयोग करके प्रोजेक्ट्स बनाने में मदद करता है। इसे माइक्रोकंट्रोलर पर आधारित डिवाइस प्रोग्राम करने के लिए डिज़ाइन किया गया है।
- Arduino is used in robotics, automation, IoT, and other electronics projects.
- Arduino का उपयोग रोबोटिक्स, ऑटोमेशन, IoT, और अन्य इलेक्ट्रॉनिक्स प्रोजेक्ट्स में किया जाता है।

Characterstics of Arduino Device:

1. Open-source platform(ओपन-सोर्स प्लेटफ़ॉर्म):

- Both the Arduino hardware and software are open-source, meaning anyone can use and customize them for free.
- Arduino हार्डवेयर और सॉफ़्टवेयर दोनों ओपन-सोर्स हैं, यानी इन्हें हर कोई फ्री में इस्तेमाल और कस्टमाइज़ कर सकता है।

2. Easy Programming(आसान प्रोग्रामिंग):

- The Arduino IDE is very simple to use. It is based on C and C++ language, which makes it easy for coding.
- Arduino IDE का उपयोग करना बहुत सरल है। यह C और C++ भाषा पर आधारित है, जो इसे कोडिंग के लिए आसान बनाता है।

3. Plug and Play(प्लग एंड प्ले):

- Arduino can be easily connected to a computer via USB and programmed instantly.
- Arduino को कंप्यूटर से USB के जरिए आसानी से कनेक्ट किया जा सकता है और तुरंत प्रोग्राम किया जा सकता है।

4. Sensor and Module Support(सेंसर और मॉड्यूल सपोर्ट):

- Arduino supports a wide variety of sensors (e.g. temperature, light) and modules (e.g. Wi-Fi, Bluetooth), allowing it to be used in a variety of projects.
- Arduino कई प्रकार के सेंसर (जैसे तापमान, लाइट) और मॉड्यूल (जैसे वाई-फाई, ब्लूटूथ) को सपोर्ट करता है, जिससे इसे अलग-अलग प्रोजेक्ट्स में इस्तेमाल किया जा सकता है।

5. Low power requirement(कम पावर की आवश्यकता):

- Arduino consumes low power, making it ideal for battery-based projects.
- Arduino कम बिजली खपत करता है, जिससे यह बैटरी आधारित प्रोजेक्ट्स के लिए आदर्श बनता है।

6. Digital and Analog Pins(डिजिटल और एनालॉग पिन्स):

- The Arduino has digital and analog pins, which help connect devices such as sensors, motors, and LEDs.
- Arduino में डिजिटल और एनालॉग पिन्स होते हैं, जो सेंसर, मोटर्स, और एलईडी जैसी डिवाइस को जोड़ने में मदद करते हैं।

7. Flexible and Customizable(लचीला और कस्टमाइज़ेबल):

- Arduino can be customized and modified to suit your needs.
- Arduino को आपकी जरूरत के हिसाब से कस्टमाइज़ और मॉडिफाई किया जा सकता है।

8.Large Community Support(बड़ी समुदाय सहायता):

- Arduino has a large and active community, which guides new users and provides solutions.
- Arduino का एक बड़ा और सक्रिय समुदाय है, जो नए उपयोगकर्ताओं को गाइड करता है और समाधान प्रदान करता है।

Arduino IDE (Integrated Development Environment):

- It is a software which is used to program the Arduino device. It plays a very important role in IoT (Internet of Things) projects.
- एक सॉफ़्टवेयर है, जिसका उपयोग Arduino डिवाइस को प्रोग्राम करने के लिए किया जाता है। यह IoT (Internet of Things) प्रोजेक्ट्स में बहुत महत्वपूर्ण भूमिका निभाता है।

Arduino IDE की मुख्य विशेषताएँ:

1. Programming Interface(प्रोग्रामिंग इंटरफ़ेस):

- Arduino IDE is used to write, edit code and upload it to the Arduino device.
- It is based on C/C++ language but is simplified for beginners.
- Arduino IDE का उपयोग कोड लिखने, एडिट करने और Arduino डिवाइस पर अपलोड करने के लिए किया जाता है।
- यह C/C++ भाषा पर आधारित है, लेकिन इसे शुरुआती लोगों के लिए सरल बनाया गया है।

2.Pre-built Library Support(प्री-बिल्ट लाइब्रेरी सपोर्ट):

- There are pre-built libraries available for sensors, modules (e.g. Wi-Fi, Bluetooth) used in IoT projects.
- Example: "ESP8266" library for WiFi module.
- IoT प्रोजेक्ट्स में उपयोग होने वाले सेंसर, मॉड्यूल (जैसे वाई-फाई, ब्लूटूथ) के लिए पहले से तैयार लाइब्रेरी उपलब्ध हैं।
- उदाहरण: WiFi मॉड्यूल के लिए "ESP8266" लाइब्रेरी।

3. Serial Monitor(सीरियल मॉनिटर):

- Serial Monitor is a tool that allows you to view data from sensors in real-time.
- It is very useful for testing and debugging devices in IoT.
- सीरियल मॉनिटर एक टूल है, जो सेंसर से डेटा को रियल-टाइम में देखने की सुविधा देता है।
- यह IoT में डिवाइस की टेस्टिंग और डिबगिंग के लिए बहुत उपयोगी है।

4. Uploading the code in simple steps:

- Connect the Arduino to the computer via USB, and send the code directly to the device by pressing the “Upload” button.
- This helps to setup the IoT device quickly and easily.
- Arduino को कंप्यूटर से USB के जरिए कनेक्ट करें, और कोड को “अपलोड” बटन दबाकर सीधे डिवाइस में भेज दें।
- यह IoT डिवाइस को जल्दी और आसानी से सेटअप करने में मदद करता है।

5. Cloud Programming Support:

- Arduino IDE now also supports cloud programming, allowing coding and projects to be accessed and modified from anywhere.
- Arduino IDE अब क्लाउड प्रोग्रामिंग का भी सपोर्ट करता है, जिससे कोडिंग और प्रोजेक्ट को कहीं से भी एक्सेस और मॉडिफाई किया जा सकता है।

6. Useful for IoT projects:

- Arduino IDE is used to build IoT projects such as smart homes, smart lights, and data monitoring systems.
- Arduino IDE का उपयोग IoT प्रोजेक्ट्स जैसे स्मार्ट होम, स्मार्ट लाइट, और डेटा मॉनिटरिंग सिस्टम बनाने के लिए किया जाता है।

How does the Arduino IDE work (Arduino IDE कैसे काम करता है):

1. Writing the code (Sketch)

- A simple text editor is provided to write code in Arduino IDE.
- This code is called "Sketch".
- This code is based on C/C++ programming language.
- Arduino IDE में कोड लिखने के लिए एक सिंपल टेक्स्ट एडिटर दिया गया है।
- इस कोड को "Sketch" कहा जाता है।
- यह कोड C/C++ प्रोग्रामिंग भाषा पर आधारित होता है।

2. Using Library:

- Pre-built libraries for different sensors or modules (e.g. Wi-Fi, Bluetooth) can be included as per the requirement.
- Libraries are added using "Include Library" option.
- जरूरत के हिसाब से अलग-अलग सेंसर या मॉड्यूल (जैसे Wi-Fi, Bluetooth) के लिए पहले से बनी हुई लाइब्रेरी को शामिल किया जा सकता है।
- लाइब्रेरी को "Include Library" ऑप्शन से जोड़ा जाता है।

3. Verify/Compile the code:

- After writing the code, click on the "Verify" button to check if there is any error in the code or not.
- It converts the code into machine language (Hex Code), which the Arduino board can understand.
- कोड लिखने के बाद "Verify" बटन पर क्लिक करके चेक किया जाता है कि कोड में कोई गलती (Error) है या नहीं।
- यह कोड को मशीन भाषा (Hex Code) में बदलता है, जिसे Arduino बोर्ड समझ सकता है।

4. Connection to Arduino:

- The Arduino board is connected to the computer via USB.
- The correct board and port are selected in the “Tools” menu.

- Arduino बोर्ड को कंप्यूटर से USB के जरिए कनेक्ट किया जाता है।
- सही बोर्ड और पोर्ट का चयन “Tools” मेन्यू में किया जाता है।

5. Uploading the code:

- The code is uploaded to the Arduino board by clicking on the “Upload” button.
- After the code is uploaded, the Arduino device starts working accordingly.

- “Upload” बटन पर क्लिक करके कोड को Arduino बोर्ड में अपलोड किया जाता है।
- कोड अपलोड होने के बाद Arduino डिवाइस उसी अनुसार काम करने लगता है।

6. Use of Serial Monitor:

- "Serial Monitor" is used in Arduino IDE to read and monitor data from the Arduino board.
- It helps in testing and debugging the device in real-time.
- Arduino IDE में "Serial Monitor" का उपयोग किया जाता है, जिससे Arduino बोर्ड से डेटा पढ़ा और मॉनिटर किया जा सकता है।
- यह रियल-टाइम में डिवाइस की टेस्टिंग और डिबगिंग के लिए मदद करता है।

Arduino Programming Language:

- The Arduino programming language, also known as "Arduino Language", is actually a simplified version of the C and C++ programming languages.
- It is specifically designed to program electronics projects and microcontroller-based systems.
- It is written and uploaded through the Arduino IDE (Integrated Development Environment).
- Arduino प्रोग्रामिंग लैंग्वेज, जिसे “Arduino Language” भी कहा जाता है, असल में C और C++ प्रोग्रामिंग लैंग्वेज का एक सरल वर्जन है।
- इसे खास तौर पर इलेक्ट्रॉनिक्स प्रोजेक्ट्स और माइक्रोकंट्रोलर बेस्ड सिस्टम्स को प्रोग्राम करने के लिए डिज़ाइन किया गया है।
- इसे Arduino IDE (Integrated Development Environment) के जरिए लिखा और अपलोड किया जाता है।

Structure of Arduino program:

- The Arduino code is divided into two main parts:
- Arduino कोड दो मुख्य भागों में बंटा होता है:

(a): setup(). (b): loop()

(1): setup():

- This is a function that runs only once, when the program starts. It contains things that only need to be set at the beginning, like making pins inputs or outputs.
- यह एक फंक्शन है जो केवल एक बार, प्रोग्राम शुरू होने पर चलता है। इसमें उन चीजों को लिखा जाता है जिन्हें सिर्फ शुरुआत में सेट करना होता है, जैसे पिन्स को इनपुट या आउटपुट बनाना।

Name

```
void setup() {
  pinMode(13, OUTPUT); // Setting pin number 13 to output
}
```

Return type

PinMode(12, input);

Return type – fn. Name (Parameter)
{
}

(2): loop():

- This function runs repeatedly as long as the Arduino board is powered on. It contains code that should be executed repeatedly.
- यह फंक्शन बार-बार चलता रहता है जब तक Arduino बोर्ड चालू है। इसमें वह कोड लिखा जाता है जो बार-बार एक्सीक्यूट होना चाहिए।

Name)

RT

```
void loop() {
  digitalWrite(13, HIGH); // Switch pin number 13 ON
  delay(1000);        // wait 1 second
  digitalWrite(13, LOW); // Switch pin no. 13 off
  delay(1000);        // wait 1 second
}
```

ON
1sec
off

Data Types in Arduino:

- Data Types means what type of data (eg: numbers, letters, words) can be stored in a variable. Data types in Arduino tell the program how much memory the variable will have and what type of data will be stored in it.
- डेटा प्रकार (Data Types) का मतलब है, किसी वैरिएबल में किस प्रकार का डेटा (जैसे: नंबर, अक्षर, शब्द) स्टोर किया जा सकता है। Arduino में डेटा प्रकार प्रोग्राम को यह बताते हैं कि वैरिएबल की मेमोरी कितनी होगी और उसमें किस प्रकार का डेटा स्टोर किया जाएगा।

Main data types and their uses:

1. void

- It is used in the function definition.
- It indicates that the function will not return anything.

- इसका उपयोग फंक्शन की परिभाषा (function definition) में किया जाता है।
- यह बताता है कि फंक्शन कुछ भी वापस (return) नहीं करेगा।

```
void setup() {
  Serial.begin(9600);  // The function does not return anything
}
```

2. int (integer)

- **Meaning**: Whole Numbers.
- **Range**: -32,768 to 32,767.
- **Memory**: 2 bytes.
- It is used for general calculations and variables.

- अर्थ: पूर्णांक (Whole Numbers)।
- रेंज: -32,768 से 32,767 तक।
- मेमोरी: 2 बाइट।
- इसका उपयोग सामान्य गणनाओं और वैरिएबल्स के लिए होता है।

16 → 1byte ⇒ 8bit
2 byte ⇒ 8×2
= 16bit

(*) $\boxed{-2^{n-1} \text{ to } +2^{n-1}-1}$

$\Rightarrow -2^{16-1} \text{ to } +2^{16-1}-1$

$\Rightarrow -2^{15} \text{ to } 2^{15}-1$

$\Rightarrow -32768 \text{ to } 32768-1 = 32767$

3. float (Floating-point number)

- **Meaning**: Decimal numbers.
- **Range**: -3.4028235E+38 to 3.4028235E+38.
- **Memory**: 4 bytes.
- It is used when precision is required.
- अर्थ: दशमलव वाले नंबर (Decimal Numbers)।
- रेंज: -3.4028235E+38 से 3.4028235E+38।
- मेमोरी: 4 बाइट।
- इसका उपयोग तब किया जाता है, जब सटीकता (precision) की जरूरत हो।

⇒ 32.25

⇒ -32.25 - float ⟹ 4 byte

4. char (Character)

- **Meaning**: A letter or symbol (Single Character).
- **Memory**: 1 byte.
- It is used to store a single character.
- अर्थ: एक अक्षर या प्रतीक (Single Character)।
- मेमोरी: 1 बाइट।
- इसका उपयोग सिंगल कैरेक्टर स्टोर करने के लिए होता

5. String

- **Meaning**: A sequence of characters.
- It is used to store sentences, names, or any other text.
- अर्थ: अक्षरों या शब्दों का समूह (Sequence of Characters)।
- इसका उपयोग वाक्य, नाम, या किसी अन्य टेक्स्ट को स्टोर करने के लिए होता

6. boolean

- **Meaning**: true or false.
- **Memory**: 1 byte.
- It is used in decision making code.
- अर्थ: सही (true) या गलत (false)।
- मेमोरी: 1 बाइट।
- इसका उपयोग निर्णय लेने वाले कोड में किया जाता

Char c → ' ' / 'a' / 'A'

Char c = '11' / 'ab' / 'AB' ✗

Function:

- In Arduino, a function is a block of code that can be used again and again. This helps in making the program easy and organized.
- Arduino में Function एक ऐसा कोड का टुकड़ा (block of code) होता है, जिसे बार-बार उपयोग किया जा सकता है। यह प्रोग्राम को आसान और व्यवस्थित बनाने में मदद करता है।

Work of function :

- Making the code short and clean.
- Writing frequently used code in one place.
- Divide a large program into smaller parts.
- कोड को छोटा और साफ बनाना।
- बार-बार उपयोग होने वाले कोड को एक ही जगह लिखना।
- बड़े प्रोग्राम को छोटे हिस्सों में बांटना।

```
main( )
{
  add(a,b);
  add(c,d);
  add(e,y);
}
```

```
main( ){
  int,a,b
  Binty (a+b)
  int c,d;
  Bint (c+d);
```

Types of function :

- There are 2 types of functions in Arduino:
- Arduino में Function के 2 प्रकार होते हैं:

(1): Predefine function.

(2): User define function.

(1): Predefine function.

- These are already available inside the Arduino, such as:
- setup(): Runs only once when the program starts.
- loop(): Runs repeatedly as long as the board is on.
- ये Arduino के अंदर पहले से उपलब्ध होते हैं, जैसे:
- setup(): प्रोग्राम शुरू होने पर सिर्फ एक बार चलता है।
- loop(): बार-बार चलता रहता है जब तक बोर्ड ऑन रहता है।

Syntax:

```
return_type function_name(parameters) {
// Function code
return value; // (if any value is to be returned
}
```

(2): User defined function

- This is code that you define to suit your needs. Instead of writing the same code every time you work with an IoT device (you can put it into a custom function and call it repeatedly as needed.
- यह ऐसा कोड होता है जिसे आप अपनी जरूरत के हिसाब से परिभाषित (define) करते हैं। जब आप IoT डिवाइस के साथ काम करते हैं, तो हर बार वही कोड लिखने की बजाय, आप उसे एक Custom Function में डाल सकते हैं और जरूरत पड़ने पर बार-बार कॉल कर सकते हैं।

Suntax:

```
return_type function_name(parameters) {
// Function code
return value; // (if any value is to be returned
}
```

Types of Control Statements:

- Control Statements are the instructions that control the flow of execution of the program. They are used to decide which code will run when and how many times.
- Control Statements वे निर्देश (instructions) होते हैं जो प्रोग्राम के निष्पादन (execution) के प्रवाह (flow) को नियंत्रित करते हैं। इनका उपयोग यह तय करने के लिए किया जाता है कि कौन-सा कोड कब और कितनी बार चलेगा।

1. Conditional Statements (शर्त वाले बयान):

- These help the program to take decisions.
- Example: if, if-else, switch-case.
- ये प्रोग्राम को निर्णय लेने में मदद करते हैं।
- उदाहरण: if, if-else, switch-case।

Syntax:

```
if (condition) {
  // This code will run if the condition is true
}
```

20==15

20

```
int a=15;
if (a==15)
Printf("Hello");
```

true

O/P => Hello

2. Looping Statements (दोहराव वाले बयान):

(a) for Loop:

- It is used to repeat the code a certain number of times.
- यह एक निश्चित संख्या में कोड को दोहराने के लिए उपयोग होता है।

Syntax:

```
for (initialization; condition; increment) {
// this code will run
}
```

5 < 5

a i

0 < 5
1 < 5
2 < 5
3 < 5
4 < 5

```
for (int i=0; a<5; a++){
    Printf( i);
```

0 1 2 3 4

(b) While Loop:

- While loop is a type of looping statement that runs the code repeatedly as long as a given condition remains true.
- While loop एक प्रकार का लूपिंग स्टेटमेंट है जो तब तक कोड को बार-बार चलाता है जब तक दी गई शर्त (condition) सही (true) रहती है।

Syntax:

```
while (condition) {
// This code will continue to run as long as the condition is true.
}
```

```
int i=0;
while(i<5){
    Printf(i)
    i++;
}
```

(c) do-while Loop (पहले करो-फिर देखो लूप) :

- This loop runs at least once, then checks the condition.
- यह लूप कम से कम एक बार जरूर चलता है, फिर शर्त को चेक करता है।

Syntax:

```
do {
//This code will run first then the condition will be checked.
} while (condition);
```

Chapter No-4 (IoT ARDUINO)

Q 1:What is Arduino IDE ? How does Arduino IDE work?
Q 1: Arduino IDE क्या है? Arduino IDE कैसे काम करता है?

Q 2:What is Arduino device? Explain the various pins available in it.
Q 2:Arduino device क्या है? इसमें उपलब्ध विभिन्न pins की व्याख्या कीजिए।

Q 3: Describe the Arduino programming language.
Q 3:आर्डिनो प्रोग्रामिंग-भाषा का वर्णन कीजिये।

Q 4:What is a Control Statement? Give various examples also.
Q 4:Control Statement क्या है? विभिन्न उदाहरण भी दीजिए।

7:30

Q 5: Explain the operators used in Arduino IDE.
Q 5:Arduino IDE में प्रयुक्त ऑपरेटरों को समझाएं।

Introduction to Machine 2 Machine:-

- Machine to Machine (M2M) in IoT means that two or more machines exchange information with each other without any human intervention. This process is completely automated
- Machine to Machine (M2M) in IoT का मतलब है कि दो या दो से अधिक मशीनें बिना किसी मानव के हस्तक्षेप के एक-दूसरे से जानकारी का आदान-प्रदान करती हैं। ये प्रक्रिया पूरी तरह से स्वचालित होती है
- In M2M, one machine (such as a sensor) sends data to another machine (such as a server). This data can be sent via the Internet.
- M2M में एक मशीन (जैसे एक सेंसर) दूसरी मशीन (जैसे सर्वर) को डेटा भेजती है। ये डेटा इंटरनेट के माध्यम से भेजा जा सकता है।

M2M के प्रमुख घटक (Key Components of M2M):

- **Sensors**: These are needed to collect data, such as temperature, humidity, motion, etc.
- सेंसर: ये डेटा इकट्ठा करने के लिए जरूरी हैं, जैसे तापमान, नमी, गति आदि।
- **Connectivity**: The Internet, Wi-Fi, or other networks are used to send data from one machine to another.
- कनेक्टिविटी: डेटा को एक मशीन से दूसरी मशीन तक भेजने के लिए इंटरनेट, Wi-Fi, या अन्य नेटवर्क का उपयोग किया जाता है।
- **Data Processing:** Once the data is received it is processed so that necessary decisions can be taken.
- डेटा प्रोसेसिंग: डेटा प्राप्त होने के बाद उसे प्रोसेस किया जाता है ताकि आवश्यक निर्णय लिया जा सके।

Actuators: These are the devices that take actions after the data is processed, like turning on/off lights, starting/stopping motors, etc.

एक्चुएटर्स: ये वो डिवाइस हैं जो डेटा प्रोसेस होने के बाद एक्शन लेते हैं, जैसे लाइट्स को ऑन/ऑफ करना, मोटर्स को चालू/बंद करना आदि।

M2M (मशीन टू मशीन) के उपयोग (Uses of M2M (Machine to Machine)

- Uses of M2M (Machine to Machine) technology can be seen in various fields where machines work automatically and share information with each other. Examples:

- M2M (मशीन टू मशीन) तकनीक का उपयोग विभिन्न क्षेत्रों में देखा जा सकता है जहाँ मशीनें स्वचालित रूप से काम करती हैं और एक-दूसरे के साथ जानकारी share करती हैं। उदाहरण:

1. स्मार्ट होम्स (Smart Homes):

- M2M technology is widely used in smart home devices like lights, thermostats, and security cameras.
- For example, a smart thermostat measures the temperature through sensors and automatically turns on the heater or air conditioner as needed.

- घरों में M2M तकनीक का उपयोग स्मार्ट डिवाइसेस जैसे लाइट्स, थर्मोस्टेट, और सिक्योरिटी कैमरों में किया जाता है।
- उदाहरण के लिए, आपका स्मार्ट थर्मोस्टेट अपने आप तापमान को सेंसर से मापता है और जरूरत के हिसाब से हीटर या एयर कंडीशनर को चालू करता है।

2. स्वास्थ्य सेवाएं (Healthcare)

- Health devices like heart rate monitors and blood pressure monitors use M2M technology.
- These devices automatically send patient data to the hospital or doctor's server, giving doctors real-time information.
- हेल्थ डिवाइसेस जैसे हार्ट रेट मॉनिटर और ब्लड प्रेशर मॉनिटर M2M का उपयोग करते हैं।
- ये डिवाइसेस अपने आप मरीज का डेटा अस्पताल या डॉक्टर के सर्वर पर भेज देते हैं ताकि डॉक्टर को रियल-टाइम जानकारी मिल सके।

3. स्मार्ट सिटी (Smart City):

- M2M is used to make traffic lights, smart parking systems, and water supply systems more efficient and automated.
- For instance, smart traffic lights use data from traffic sensors to adjust based on traffic flow, helping reduce traffic jams.
- ट्रैफिक लाइट्स, स्मार्ट पार्किंग सिस्टम, और पानी की आपूर्ति को स्वचालित और कुशल बनाने के लिए M2M का उपयोग किया जाता है।
- उदाहरण के लिए, स्मार्ट ट्रैफिक लाइट्स ट्रैफिक सेंसर से डेटा लेकर ट्रैफिक के हिसाब से लाइट्स बदलती हैं ताकि ट्रैफिक जाम कम हो।

4. Industrial Applications:

- In factories, machines are connected through M2M to improve production processes.
- If there's an issue with a machine, it can automatically notify other machines or the server for immediate repair.
- फैक्ट्री में मशीनें M2M का उपयोग करके एक-दूसरे से जुड़ी होती हैं ताकि उत्पादन प्रक्रिया में सुधार हो सके।
- यदि किसी मशीन में गड़बड़ी हो, तो वह अपने आप अन्य मशीनों या सर्वर को सूचित करती है ताकि तुरंत मरम्मत हो सके।

5. ऊर्जा और यूटिलिटी (Energy and Utilities):

- Smart electricity meters use M2M to send data directly to the electricity company.
- This provides the company with accurate information about power usage, allowing them to generate bills without manual readings.
- बिजली के स्मार्ट मीटर M2M का उपयोग करते हैं और सीधे इलेक्ट्रिसिटी कंपनी को डेटा भेजते हैं।
- इससे कंपनियों को बिजली के उपयोग की सही जानकारी मिलती है और वे बिना किसी मैन्युअल रीडिंग के ही बिल बना सकती हैं।

6. परिवहन और लॉजिस्टिक्स (Transportation and Logistics):

- Trucks and other vehicles have sensors that automatically send information about location, fuel level, and other details.
- This helps companies track their vehicles' real-time location and status, making management easier.
- ट्रक और अन्य गाड़ियों में लगे सेंसर अपने आप लोकेशन, फ्यूल लेवल और अन्य जानकारी भेजते हैं।
- इससे कंपनियों को गाड़ियों का सही लोकेशन और स्थिति पता चलता है, जिससे उन्हें ट्रैक करना आसान होता है।

Difference between IoT and M2M

- M2M is about communication between machines.
- It supports point-to-point communication.
- The devices do not necessarily rely on an internet connection.
- M2M is mostly a hardware-based technology.
- Machines usually communicate with only one machine at a time.
- A device can be connected via a mobile or other network

- IoT is about sensor automation and internet platform.
- It supports cloud communication.
- The devices depend on internet connection.
- IoT is hardware and software based technology.
- Many users can access the internet at a time.
- Data distribution depends on Internet protocol (IP) network.

Difference between IoT and M2M

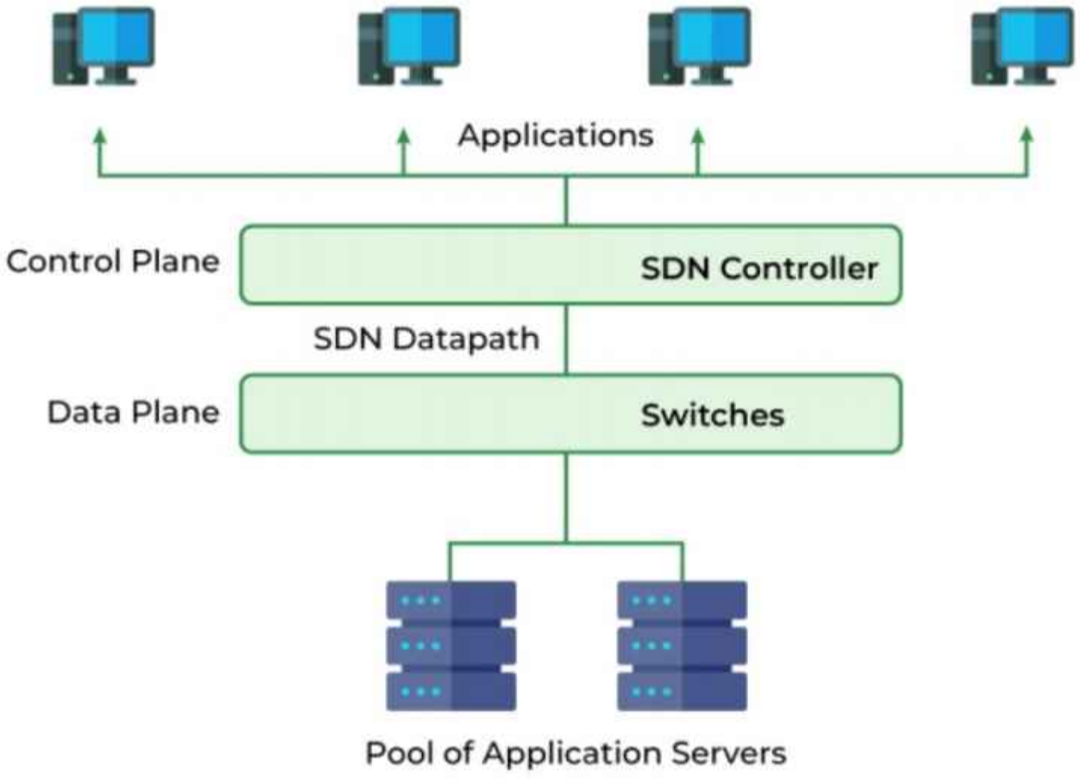
- M2M मशीनों के बीच संचार के बारे में है।
- यह Point-से-Point संचार का समर्थन करता है।
- जरूरी नहीं कि डिवाइस इंटरनेट कनेक्शन पर निर्भर हों।
- M2M ज्यादातर हार्डवेयर आधारित तकनीक है।
- मशीनें आमतौर पर एक बार में एक ही मशीन से संवाद करती हैं।
- एक डिवाइस को मोबाइल या अन्य नेटवर्क के माध्यम से कनेक्ट किया जा सकता है

- IoT सेंसर ऑटोमेशन और इंटरनेट प्लेटफॉर्म के बारे में है।
- यह क्लाउड संचार का समर्थन करता है।
- डिवाइस इंटरनेट कनेक्शन पर निर्भर करते हैं।
- IoT हार्डवेयर और सॉफ्टवेयर आधारित तकनीक है।
- कई उपयोगकर्ता एक समय में इंटरनेट पर पहुंच सकते हैं।
- डेटा वितरण Internet protocol (IP) नेटवर्क पर निर्भर करता है।

Software Defined Networking (SDN):

- SDN stands for Software Defined Networking. Its main goal is to separate the Control Plane and the Data Plane in a network.
- SDN का पूरा नाम Software Defined Networking है। इसका मुख्य उद्देश्य नेटवर्क के कंट्रोल प्लेन (Control Plane) और डेटा प्लेन (Data Plane) को अलग करना है
- Control Plane: This part makes decisions about where data should go.
- यह नेटवर्क के निर्णय लेने का हिस्सा होता है, यानी यह तय करता है कि डेटा कहां और कैसे भेजा जाएगा।
- Data Plane: This part is responsible for actually sending data from one device to another.
- यह वह हिस्सा होता है जो असल में डेटा को एक डिवाइस से दूसरी डिवाइस तक भेजता है।

Software Defined Networking (SDN)



SDN का कार्य सिद्धांत (Working of SDN):

- SDN networks have a software controller that manages all network devices. This controller makes decisions for the network, like which route data should take, which policies to apply, and so on.
- SDN नेटवर्क में सॉफ्टवेयर कंट्रोलर होता है जो नेटवर्क के सभी डिवाइसेस को मैनेज करता है। इस कंट्रोलर का काम है कि वह सभी नेटवर्क डिवाइसेस के लिए निर्णय लेता है, जैसे डेटा किस रूट से भेजा जाएगा, किस प्रकार की पॉलिसीज़ लागू होंगी, आदि।

(1): Separation:

- SDN separates control and data planes, meaning data transmission and network decisions happen independently.
- SDN में कंट्रोल और डेटा प्लेन को अलग कर दिया जाता है। इससे डेटा ट्रांसमिशन और नेटवर्क निर्णय अलग-अलग होते हैं।

(2) Centralized Control:

- A single controller manages the whole network, making it easier to make changes and updates across the network.
- नेटवर्क कंट्रोलर पूरे नेटवर्क को एक जगह से नियंत्रित करता है। इससे नेटवर्क में बदलाव या अपडेट्स करना आसान हो जाता है।

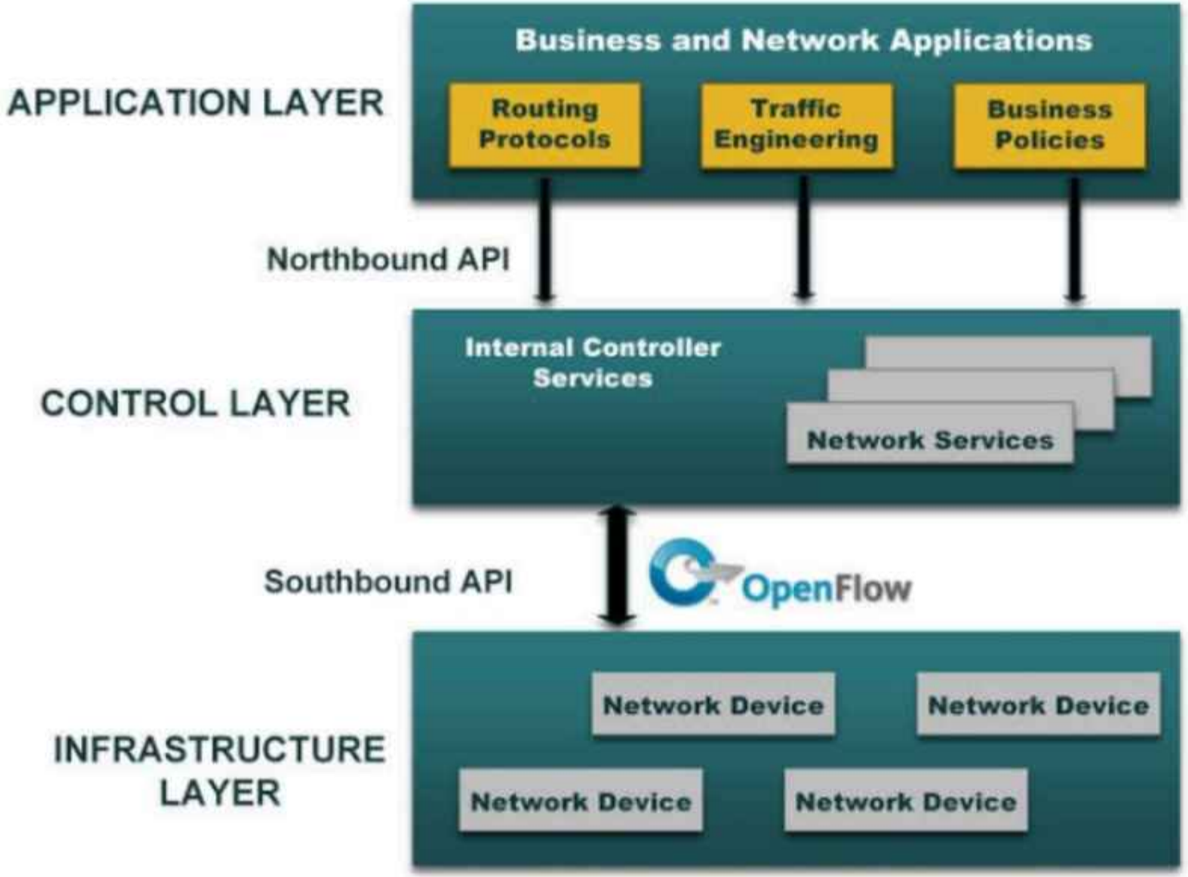
(3): Programmable Network:

- SDN enables network administrators to use software to change network behavior. They can update data flows, security policies, and routing with ease.
- SDN में सॉफ्टवेयर प्रोग्रामिंग का उपयोग करके नेटवर्क के व्यवहार को बदला जा सकता है। इससे एडमिनिस्ट्रेटर को नेटवर्क के फ्लो, सिक्योरिटी पॉलिसी, और रूटिंग को बदलने की सुविधा मिलती है।

SDN architecture :

- SDN (Software Defined Networking) Architecture is divided into three main layers: Infrastructure Layer, Control Layer, and Application Layer. This architecture makes the network flexible, programmable, and centrally managed.
- SDN (Software Defined Networking) आर्किटेक्चर में नेटवर्क को तीन मुख्य परतों (Layers) में विभाजित किया गया है: इंफ्रास्ट्रक्चर लेयर, कंट्रोल लेयर, और एप्लिकेशन लेयर। यह आर्किटेक्चर नेटवर्क को लचीला, प्रोग्रामेबल और सेंट्रलाइज्ड तरीके से प्रबंधित करने की सुविधा देता है।

SDN architecture :



1. इंफ्रास्ट्रक्चर लेयर (Infrastructure Layer):

- Also known as the Data Plane.
- This layer includes physical network devices like routers, switches, and other hardware.
- Its primary function is to send and receive data packets based on the instructions it receives from the controller.
- इसे डेटा प्लेन भी कहा जाता है।
- इसमें नेटवर्क के भौतिक उपकरण जैसे राउटर, स्विच, और अन्य हार्डवेयर डिवाइसेस शामिल होते हैं।
- इस लेयर का काम डेटा पैकेट को भेजना और रिसीव करना होता है। यह कंट्रोलर से निर्देश प्राप्त करती है कि डेटा कैसे और कहाँ भेजा जाए।

2. कंट्रोल लेयर (Control Layer):

- The Control Layer is the brain of the SDN architecture and contains the SDN Controller.
- This layer manages and controls the entire network by making decisions on data routing, policy implementation, and security.
- The Controller decides the direction and handling of network traffic, acting as an intermediary between the Application Layer and the Infrastructure Layer.
- कंट्रोल लेयर SDN आर्किटेक्चर का "मस्तिष्क" होता है और इसमें SDN कंट्रोलर शामिल होता है।
- यह लेयर पूरे नेटवर्क को प्रबंधित और नियंत्रित करती है, डेटा रूटिंग, पॉलिसी लागू करने और सुरक्षा पर निर्णय लेकर।
- कंट्रोलर नेटवर्क ट्रैफिक की दिशा और प्रबंधन का निर्णय करता है, और एप्लिकेशन लेयर और इंफ्रास्ट्रक्चर लेयर के बीच एक मध्यस्थ के रूप में कार्य करता है।

3. एप्लिकेशन लेयर (Application Layer):

- This layer contains various network applications and services, such as firewalls, traffic management, and security policies.
- This layer tells the controller what kind of services and policies to implement.
- The job of the application layer is to communicate with the controller to enhance the functionality of the network.
- इस लेयर में विभिन्न नेटवर्क एप्लिकेशन और सेवाएं होती हैं, जैसे फायरवॉल, ट्रैफिक मैनेजमेंट, और सिक्योरिटी पॉलिसीज़।
- यह लेयर कंट्रोलर को बताती है कि उसे किस तरह की सेवाएं और पॉलिसीज़ लागू करनी हैं।
- एप्लिकेशन लेयर का काम है कि वह नेटवर्क की कार्यक्षमता को बढ़ाने के लिए कंट्रोलर के साथ संपर्क बनाए रखे।

2. कंट्रोल लेयर (Control Layer):

- The Control Layer is the brain of the SDN architecture and contains the SDN Controller.
- This layer manages and controls the entire network by making decisions on data routing, policy implementation, and security.
- The Controller decides the direction and handling of network traffic, acting as an intermediary between the Application Layer and the Infrastructure Layer.
- कंट्रोल लेयर SDN आर्किटेक्चर का "मस्तिष्क" होता है और इसमें SDN कंट्रोलर शामिल होता है।
- यह लेयर पूरे नेटवर्क को प्रबंधित और नियंत्रित करती है, डेटा रूटिंग, पॉलिसी लागू करने और सुरक्षा पर निर्णय लेकर।
- कंट्रोलर नेटवर्क ट्रैफिक की दिशा और प्रबंधन का निर्णय करता है, और एप्लिकेशन लेयर और इंफ्रास्ट्रक्चर लेयर के बीच एक मध्यस्थ के रूप में कार्य करता है।

3. एप्लिकेशन लेयर (Application Layer):

- This layer contains various network applications and services, such as firewalls, traffic management, and security policies.
- This layer tells the controller what kind of services and policies to implement.
- The job of the application layer is to communicate with the controller to enhance the functionality of the network.
- इस लेयर में विभिन्न नेटवर्क एप्लिकेशन और सेवाएं होती हैं, जैसे फायरवॉल, ट्रैफिक मैनेजमेंट, और सिक्योरिटी पॉलिसीज़।
- यह लेयर कंट्रोलर को बताती है कि उसे किस तरह की सेवाएं और पॉलिसीज़ लागू करनी हैं।
- एप्लिकेशन लेयर का काम है कि वह नेटवर्क की कार्यक्षमता को बढ़ाने के लिए कंट्रोलर के साथ संपर्क बनाए रखे।

SDN के फायदे (Benefits of SDN):

1. लचीलापन (Flexibility):

- SDN allows network policies and routing to be easily changed through software, making network updates and improvements simpler.
- SDN में सॉफ्टवेयर द्वारा नेटवर्क की पॉलिसीज़ और रूटिंग को आसानी से बदला जा सकता है। यह नेटवर्क को अपडेट करने और सुधारने में मदद करता है।

2. केंद्रकृत नियंत्रण (Centralized Management):

- With SDN, the entire network can be controlled from a single point, simplifying monitoring and management.
- SDN में नेटवर्क का नियंत्रण एक ही सेंट्रल प्वाइंट से होता है। इससे नेटवर्क का मॉनिटरिंग और मैनेजमेंट आसान हो जाता है।

3. कम लागत (Cost Savings):

- SDN reduces dependency on specialized hardware devices, lowering costs, as everything is controlled by software.
- SDN में हार्डवेयर डिवाइसेस पर निर्भरता कम होती है। इससे नेटवर्किंग की लागत में बचत होती है, क्योंकि सब कुछ सॉफ्टवेयर से नियंत्रित होता है।

4. स्केलेबिलिटी (Scalability):

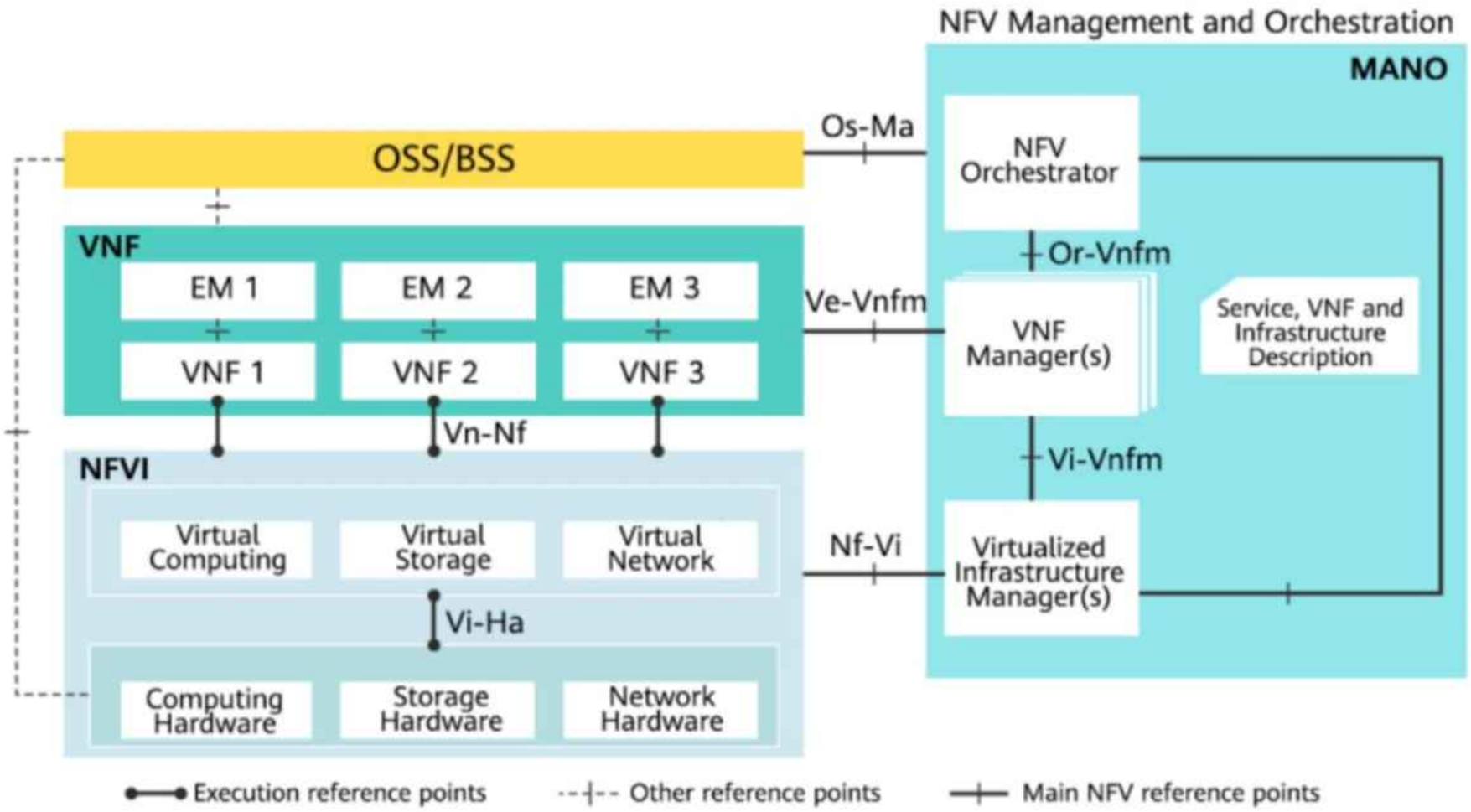
- SDN makes it easy to expand networks by adding new devices and connections.
- SDN में नेटवर्क को बड़े पैमाने पर बढ़ाया जा सकता है। नए डिवाइसेस और कनेक्शन्स को जोड़ना आसान होता है।

5. सिक्योरिटी (Security):

- Centralized control in SDN allows for stronger network security. Security policies can be implemented and managed from a single location.
- SDN में सेंट्रल कंट्रोल की वजह से नेटवर्क को अधिक सुरक्षित बनाया जा सकता है। सिक्योरिटी पॉलिसीज़ को एक ही जगह से लागू और मैनेज किया जा सकता है।

Network Function Virtualization (NFV):

- The main purpose of NFV is to virtualize various network functions. Traditionally, network devices like routers, firewalls, and load balancers are physical hardware, but with NFV, these are run as software on virtual machines (VMs) or containers.
- NFV का मुख्य उद्देश्य नेटवर्क के विभिन्न कार्यों को वर्चुअलाइज करना है। आमतौर पर नेटवर्क में राउटर, फायरवॉल, लोड बैलेंसर जैसे डिवाइसेस फिजिकल (हार्डवेयर) रूप में होते हैं, लेकिन NFV में इन्हें सॉफ्टवेयर के रूप में वर्चुअल मशीनों (VMs) या कंटेनरों में चलाया जाता है।
- This makes managing the network easier, allowing new features to be added, upgraded, and scaled with flexibility.
- इससे नेटवर्क का प्रबंधन आसान हो जाता है और नए फीचर्स जोड़ना, अपग्रेड करना और स्केलिंग करना सरल हो जाता है।


NFV Management and Orchestration
MANO
OSS/BSS
Os-Ma
NFV Orchestrator
Or-Vnfm
VNF
EM 1
EM 2
EM 3
VNF 1
VNF 2
VNF 3
Ve-Vnfm
VNF Manager(s)
Service, VNF and Infrastructure Description
Vn-Nf
NFVI
Vi-Vnfm
Virtual Computing
Virtual Storage
Virtual Network
Nf-Vi
Virtualized Infrastructure Manager(s)
Vi-Ha
Computing Hardware
Storage Hardware
Network Hardware
Execution reference points
Other reference points
Main NFV reference points

1. NFVI (Network Function Virtualization Infrastructure):

- The first layer of NFVI consists of hardware resources (CPU), storage resources (Hard disk), and Network resources (Router, Switch, and Firewalls).
- The second layer of NFVI is the virtualization layer, which separates hardware and replaces it with software.
- The third layer of NFVI is virtualized resources such as virtual computers, virtual networks, and virtual storage.
- NFVI की पहली परत में हार्डवेयर संसाधन (CPU), स्टोरेज संसाधन (हार्ड डिस्क) और नेटवर्क संसाधन (राउटर, स्विच और फ़ायरवॉल) शामिल हैं।
- NFVI की दूसरी परत वर्चुअलाइजेशन परत है, जो हार्डवेयर को अलग करती है और इसे सॉफ़्टवेयर से बदल देती है।
- NFVI की तीसरी परत वर्चुअलाइज्ड संसाधन जैसे वर्चुअल कंप्यूटर, वर्चुअल नेटवर्क और वर्चुअल स्टोरेज है।

2. VNFs (Virtual Network Functions)

- VNFs are virtual network functions, such as virtual routers, virtual firewalls, and virtual load balancers.
- These functions are run in separate virtual machines or containers and provide network services.
- Depending on NFVI, VNFs can be quickly configured, deleted, or replaced.
- VNFs वर्चुअल नेटवर्क फंक्शन्स हैं, जैसे कि वर्चुअल राउटर, वर्चुअल फायरवॉल, और वर्चुअल लोड बैलेंसर।
- इन फंक्शन्स को अलग-अलग वर्चुअल मशीनों या कंटेनरों में चलाया जाता है और ये नेटवर्क सेवाएं प्रदान करते हैं।
- NFVI पर निर्भर करते हुए, VNFs को जल्दी से कॉन्फ़िगर किया जा सकता है, हटाया जा सकता है या बदला जा सकता है।

3. NFV MANO (Management and Orchestration)

1. NFV Orchestrator: It plans and manages the entire NFV system. It deploys new VNFs, monitors their status, and scales the system.

2. VNF Manager: It controls various VNFs, such as their configuration, status monitoring, and updating.

3. Virtualized Infrastructure Manager (VIM): It manages the NFVI, such as managing virtual machines, networking, and storage.

1. NFV Orchestrator: यह पूरे NFV सिस्टम की योजना और प्रबंधन करता है। यह नए VNFs को तैनात करना, उनकी स्थिति की निगरानी करना और सिस्टम को स्केल करने का काम करता है।

2. VNF Manager: यह विभिन्न VNFs को कंट्रोल करता है, जैसे कि उनका कॉन्फ़िगरेशन, स्टेटस , और अपडेट करना।

3. Virtualized Infrastructure Manager (VIM): यह NFVI को प्रबंधित करता है, जैसे कि मशीनों का प्रबंधन, नेटवर्किंग, और स्टोरेज का कंट्रोल।

NFV के फायदे (Benefits of NFV):

1. लागत में कमी (Cost Reduction):

- Reduces the need for physical hardware, saving costs.
- फिजिकल हार्डवेयर की आवश्यकता कम हो जाती है, जिससे लागत में कमी आती है।

2. लचीलापन (Flexibility):

- Makes it easy to add new functions and change policies.
- नए फंक्शन्स जोड़ना और पॉलिसी बदलना आसान हो जाता है।

3. तेजी से सेवाएं उपलब्ध कराना (Rapid Service Deployment):

- New network functions can be quickly deployed and configured.
- नए नेटवर्क फंक्शन्स को जल्दी से तैनात और कॉन्फ़िगर किया जा सकता है।

4. स्केलेबिलिटी (Scalability):

- NFV allows the network to scale easily to handle more traffic.
- NFV से नेटवर्क को आसानी से स्केल किया जा सकता है और अधिक ट्रैफिक को संभाला जा सकता है।

5. उपयोग में आसानी (Ease of Management):

- Managing the network through software simplifies operations..
- सॉफ्टवेयर के माध्यम से नेटवर्क का प्रबंधन करना सरल हो जाता है।

WoT (Web of Things):

- WoT, or the Web of Things, is an important concept within IoT (Internet of Things).
- It aims to make IoT devices connect and communicate more effectively using web technologies.
- The primary goal of WoT is to simplify IoT by providing a standardized way to interact with smart devices over the web.

- WoT, या वेब ऑफ थिंग्स (Web of Things), IoT का एक महत्वपूर्ण हिस्सा है। इसे इस तरह समझा जा सकता है कि यह IoT डिवाइसेस को आपस में कनेक्ट करने और संचार(Communication) करने में सक्षम बनाता है।

- WoT का मुख्य उद्देश्य IoT को वेब तकनीकों के माध्यम से और अधिक आसान और उपयोगी बनाना है।

WoT के मुख्य घटक (Key Components of WoT):

- WoT consists of four main components:
- WoT के चार मुख्य घटक होते हैं:

1. थिंग्स डिस्क्रिप्शन (Things Description):

- This is a document written in JSON-LD that describes the features of an IoT device. It includes details about the device's services, actions it can perform, and the data it generates.
- यह JSON-LD में लिखा एक दस्तावेज़ होता है जो IoT डिवाइसेस का विवरण देता है। इसमें उस डिवाइस की सभी जानकारी होती है जैसे कि क्या सेवाएं देता है, क्या एक्शन ले सकता है, और क्या डेटा उत्पन्न करता है।

2. बाइंडिंग टेम्पलेट्स (Binding Templates):

- These templates help connect devices with various communication protocols (like HTTP, MQTT), allowing them to interact seamlessly.
- ये टेम्पलेट्स विभिन्न प्रोटोकॉल्स (जैसे HTTP, MQTT) के साथ डिवाइस को जोड़ने में मदद करते हैं।

3. सिक्योरिटी (Security):

- Security is crucial in WoT. It includes authentication (verifying the user) and authorization (granting access permissions), ensuring safe data usage.
- यह WoT का महत्वपूर्ण हिस्सा है। इसमें प्रमाणन (authentication) और अनुमति (authorization) शामिल है ताकि डेटा का उपयोग सुरक्षित तरीके से हो।

4. स्क्रिप्टिंग API (Scripting API):

- This API provides developers with a way to interact with and control IoT devices easily, making it simpler to build applications.
- डेवलपर्स के लिए एक इंटरफ़ेस है जो उन्हें विभिन्न IoT डिवाइसेस के साथ इंटरैक्ट करने और काम करने में सक्षम बनाता है।

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Definition	Network of physical devices connected to the internet	A framework to make IoT devices communicate via web standard
Purpose	Connect devices to exchange data	Simplify communication among devices using web technologies
Communication	Uses various, often incompatible protocols	Uses web-based, standardized protocols (like HTTP, JSON-LD)

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Definition	इंटरनेट से जुड़े भौतिक उपकरणों का नेटवर्क है	IoT उपकरणों को वेब मानक के माध्यम से संचार करने के लिए एक ढांचा (Framewok) देता है
Purpose	डेटा का आदान-प्रदान करने के लिए डिवाइस कनेक्ट करते है	वेब प्रौद्योगिकियों का उपयोग करके उपकरणों के बीच संचार को सरल बनाता है
Communication	विभिन्न, अक्सर असंगत(Incompatible) प्रोटोकॉल का उपयोग करता है	वेब-आधारित, standard प्रोटोकॉल (जैसे HTTP, JSON-LD) का उपयोग करता है

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Interoperability	Difficult for different devices to work together	Provides a common platform for easy interaction
Security	Security varies by device and protocol	Emphasizes standardized security features
Ease of Use	May require complex integrations	Easier integration and control of devices

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Interoperability	विभिन्न उपकरणों का एक साथ काम करना कठिन है।	आसान बातचीत के लिए एक साझा platform प्रदान करता है
Security	सुरक्षा डिवाइस और प्रोटोकॉल के अनुसार अलग-अलग होती है	Standard सुरक्षा सुविधाओं पर जोर दिया जाता है।
Ease of Use	जटिल एकीकरण(complex intergration की आवश्यकता हो सकती है	उपकरणों का आसान एकीकरण(integration) और नियंत्रण(Control) करता है।

Chapter No-5 (IoT and M2M)

Q 1:Explain the difference between IoT and M2M.
Q 1:IoT एवं M2M में अन्तर की व्याख्या कीजिये।

Q 2: What is the fundamental difference between IoT and WoT?
Q 2:IoT एवं WoT के बीच मौलिक अन्तर क्या है?

Q 3 What is NFV for IoT? Explain.
Q 3:NFV for IoT क्या है? समझाइये।

Q 4: Explain different types of SDN.
Q 3:SDN के विभिन्न प्रकार समझाइये।

Thank You

CSE/IT

2025 Batch

5th Semester

IOT

Polytechnic

ONE SHOT

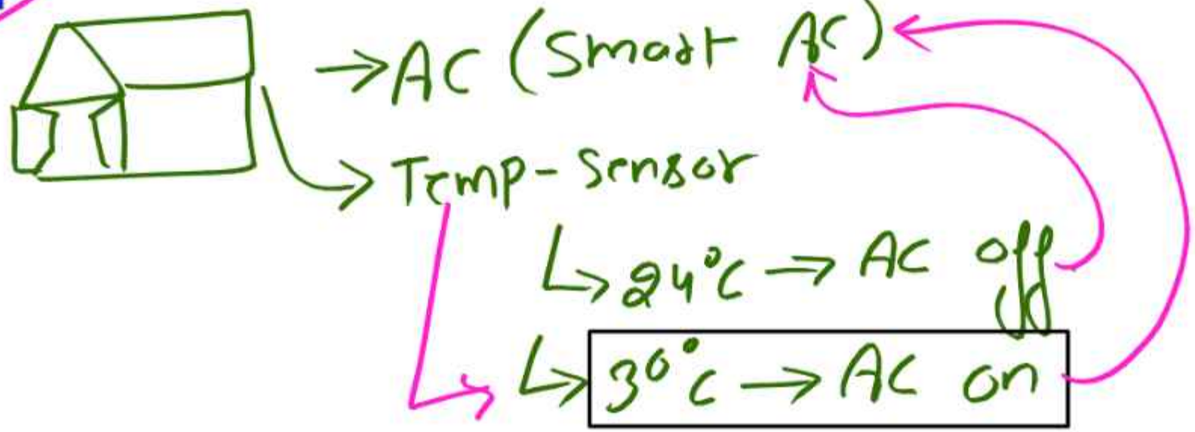
Part-2

IoT

By Vikas Sir

Introduction to Machine 2 Machine:-

- Machine to Machine (M2M) in IoT means that two or more machines exchange information with each other without any human intervention. This process is completely automated
- Machine to Machine (M2M) in IoT का मतलब है कि दो या दो से अधिक मशीनें बिना किसी मानव के हस्तक्षेप के एक-दूसरे से जानकारी का आदान-प्रदान करती हैं। ये प्रक्रिया पूरी तरह से स्वचालित होती है
- In M2M, one machine (such as a sensor) sends data to another machine (such as a server). This data can be sent via the Internet.
- M2M में एक मशीन (जैसे एक सेंसर) दूसरी मशीन (जैसे सर्वर) को डेटा भेजती है। ये डेटा इंटरनेट के माध्यम से भेजा जा सकता है।



M2M के प्रमुख घटक (Key Components of M2M):

- **Sensors**: These are needed to collect data, such as temperature, humidity, motion, etc.
- सेंसर: ये डेटा इकट्ठा करने के लिए जरूरी हैं, जैसे तापमान, नमी, गति आदि।
- **Connectivity**: The Internet, Wi-Fi, or other networks are used to send data from one machine to another.
- कनेक्टिविटी: डेटा को एक मशीन से दूसरी मशीन तक भेजने के लिए इंटरनेट, Wi-Fi, या अन्य नेटवर्क का उपयोग किया जाता है।
- **Data Processing:** Once the data is received it is processed so that necessary decisions can be taken.
- डेटा प्रोसेसिंग: डेटा प्राप्त होने के बाद उसे प्रोसेस किया जाता है ताकि आवश्यक निर्णय लिया जा सके।

Sensor → collect data → Process → Decision → action

Actuators: These are the devices that take actions after the data is processed, like turning on/off lights, starting/stopping motors, etc.

एक्चुएटर्स: ये वो डिवाइस हैं जो डेटा प्रोसेस होने के बाद एक्शन लेते हैं, जैसे लाइट्स को ऑन/ऑफ करना, मोटर्स को चालू/बंद करना आदि।

M2M (मशीन टू मशीन) के उपयोग (Uses of M2M (Machine to Machine)

- Uses of M2M (Machine to Machine) technology can be seen in various fields where machines work automatically and share information with each other. Examples:

- M2M (मशीन टू मशीन) तकनीक का उपयोग विभिन्न क्षेत्रों में देखा जा सकता है जहाँ मशीनें स्वचालित रूप से काम करती हैं और एक-दूसरे के साथ जानकारी share करती हैं। उदाहरण:

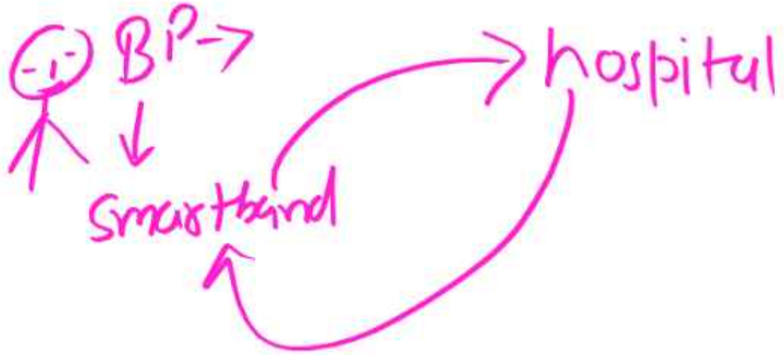
1. स्मार्ट होम्स (Smart Homes):

- M2M technology is widely used in smart home devices like lights, thermostats, and security cameras.
- For example, a smart thermostat measures the temperature through sensors and automatically turns on the heater or air conditioner as needed.

- घरों में M2M तकनीक का उपयोग स्मार्ट डिवाइसेस जैसे लाइट्स, थर्मोस्टेट, और सिक्योरिटी कैमरों में किया जाता है।
- उदाहरण के लिए, आपका स्मार्ट थर्मोस्टेट अपने आप तापमान को सेंसर से मापता है और जरूरत के हिसाब से हीटर या एयर कंडीशनर को चालू करता है।

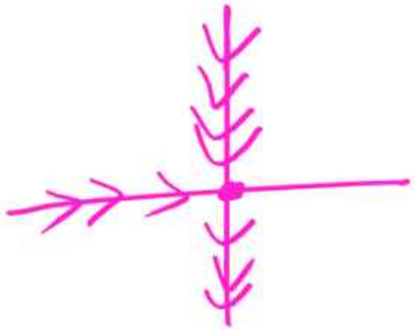
2. स्वास्थ्य सेवाएं (Healthcare)

- Health devices like heart rate monitors and blood pressure monitors use M2M technology.
- These devices automatically send patient data to the hospital or doctor's server, giving doctors real-time information.
- हेल्थ डिवाइसेस जैसे हार्ट रेट मॉनिटर और ब्लड प्रेशर मॉनिटर M2M का उपयोग करते हैं।
- ये डिवाइसेस अपने आप मरीज का डेटा अस्पताल या डॉक्टर के सर्वर पर भेज देते हैं ताकि डॉक्टर को रियल-टाइम जानकारी मिल सके।



3. स्मार्ट सिटी (Smart City):

- M2M is used to make traffic lights, smart parking systems, and water supply systems more efficient and automated.
- For instance, smart traffic lights use data from traffic sensors to adjust based on traffic flow, helping reduce traffic jams.
- ट्रैफिक लाइट्स, स्मार्ट पार्किंग सिस्टम, और पानी की आपूर्ति को स्वचालित और कुशल बनाने के लिए M2M का उपयोग किया जाता है।
- उदाहरण के लिए, स्मार्ट ट्रैफिक लाइट्स ट्रैफिक सेंसर से डेटा लेकर ट्रैफिक के हिसाब से लाइट्स बदलती हैं ताकि ट्रैफिक जाम कम हो।



4. Industrial Applications:

- In factories, machines are connected through M2M to improve production processes.
- If there's an issue with a machine, it can automatically notify other machines or the server for immediate repair.
- फैक्ट्री में मशीनें M2M का उपयोग करके एक-दूसरे से जुड़ी होती हैं ताकि उत्पादन प्रक्रिया में सुधार हो सके।
- यदि किसी मशीन में गड़बड़ी हो, तो वह अपने आप अन्य मशीनों या सर्वर को सूचित करती है ताकि तुरंत मरम्मत हो सके।

5. ऊर्जा और यूटिलिटी (Energy and Utilities):

- Smart electricity meters use M2M to send data directly to the electricity company.
- This provides the company with accurate information about power usage, allowing them to generate bills without manual readings.
- बिजली के स्मार्ट मीटर M2M का उपयोग करते हैं और सीधे इलेक्ट्रिसिटी कंपनी को डेटा भेजते हैं।
- इससे कंपनियों को बिजली के उपयोग की सही जानकारी मिलती है और वे बिना किसी मैन्युअल रीडिंग के ही बिल बना सकती हैं।

6. परिवहन और लॉजिस्टिक्स (Transportation and Logistics):

- Trucks and other vehicles have sensors that automatically send information about location, fuel level, and other details.
- This helps companies track their vehicles' real-time location and status, making management easier.
- ट्रक और अन्य गाड़ियों में लगे सेंसर अपने आप लोकेशन, फ्यूल लेवल और अन्य जानकारी भेजते हैं।
- इससे कंपनियों को गाड़ियों का सही लोकेशन और स्थिति पता चलता है, जिससे उन्हें ट्रैक करना आसान होता है।

Difference between IoT and M2M

- M2M is about communication between machines.
- It supports point-to-point communication.
- The devices do not necessarily rely on an internet connection.
- M2M is mostly a hardware-based technology.
- Machines usually communicate with only one machine at a time.
- A device can be connected via a mobile or other network

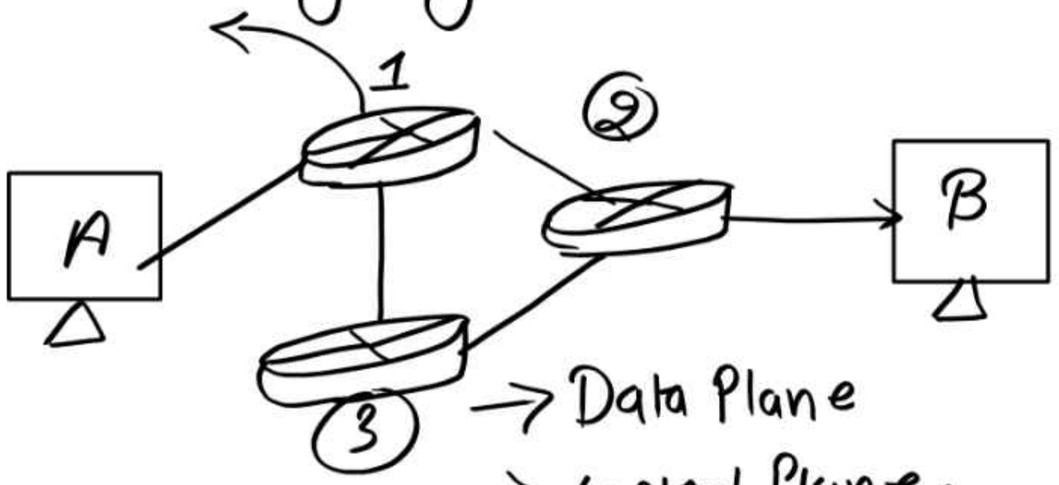
- IoT is about sensor automation and internet platform.
- It supports cloud communication.
- The devices depend on internet connection.
- IoT is hardware and software based technology.
- Many users can access the internet at a time.
- Data distribution depends on Internet protocol (IP) network.

Difference between IoT and M2M

- M2M मशीनों के बीच संचार के बारे में है।
- यह Point-से-Point संचार का समर्थन करता है।
- जरूरी नहीं कि डिवाइस इंटरनेट कनेक्शन पर निर्भर हों।
- M2M ज्यादातर हार्डवेयर आधारित तकनीक है।
- मशीनें आमतौर पर एक बार में एक ही मशीन से संवाद करती हैं।
- एक डिवाइस को मोबाइल या अन्य नेटवर्क के माध्यम से कनेक्ट किया जा सकता है

- IoT सेंसर ऑटोमेशन और इंटरनेट प्लेटफॉर्म के बारे में है।
- यह क्लाउड संचार का समर्थन करता है।
- डिवाइस इंटरनेट कनेक्शन पर निर्भर करते हैं।
- IoT हार्डवेयर और सॉफ्टवेयर आधारित तकनीक है।
- कई उपयोगकर्ता एक समय में इंटरनेट पर पहुंच सकते हैं।
- डेटा वितरण Internet protocol (IP) नेटवर्क पर निर्भर करता है।

Routing algo:->

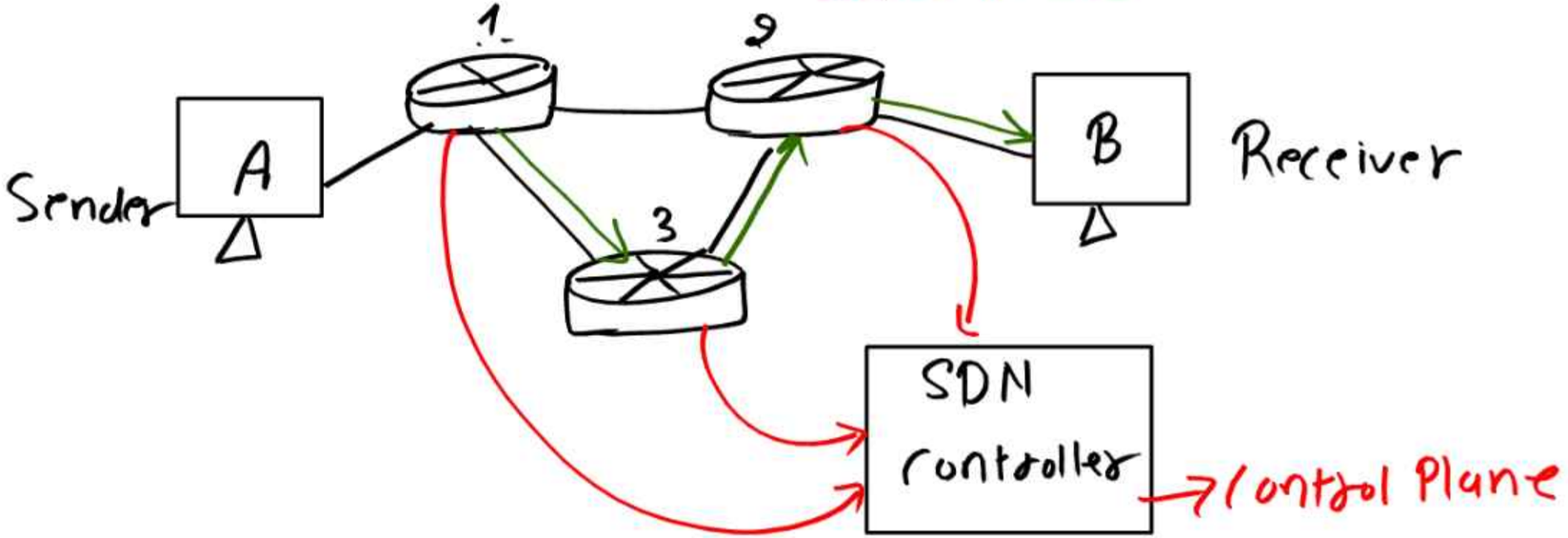


-> Data Plane

-> Control Plane

Traditional method:->

Data Plane



-> Control Plane

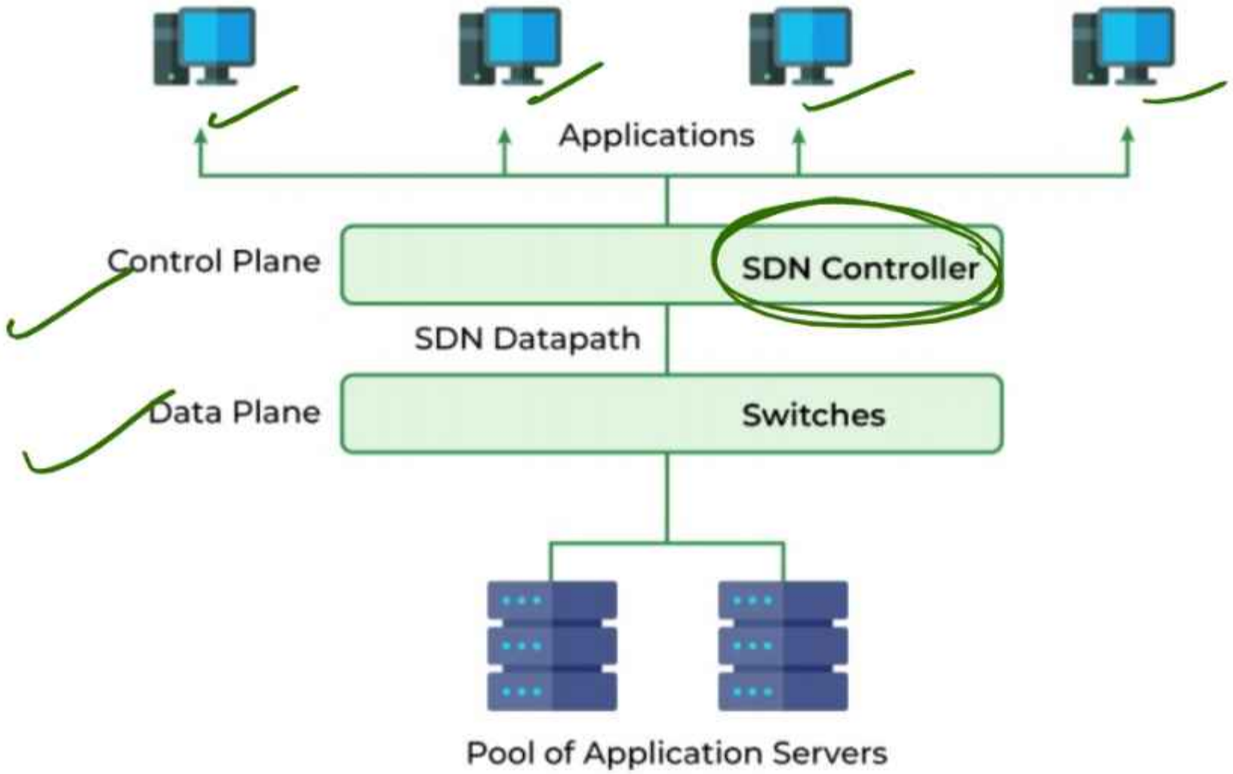
Software Defined Networking (SDN):

- SDN stands for Software Defined Networking. Its main goal is to separate the Control Plane and the Data Plane in a network.
- SDN का पूरा नाम Software Defined Networking है। इसका मुख्य उद्देश्य नेटवर्क के कंट्रोल प्लेन (Control Plane) और डेटा प्लेन (Data Plane) को अलग करना है

- Control Plane: This part makes decisions about where data should go.
- यह नेटवर्क के निर्णय लेने का हिस्सा होता है, यानी यह तय करता है कि डेटा कहां और कैसे भेजा जाएगा।

- Data Plane: This part is responsible for actually sending data from one device to another.
- यह वह हिस्सा होता है जो असल में डेटा को एक डिवाइस से दूसरी डिवाइस तक भेजता है।

Software Defined Networking (SDN)



SDN का कार्य सिद्धांत (Working of SDN):

- SDN networks have a software controller that manages all network devices. This controller makes decisions for the network, like which route data should take, which policies to apply, and so on.
- SDN नेटवर्क में सॉफ्टवेयर कंट्रोलर होता है जो नेटवर्क के सभी डिवाइसेस को मैनेज करता है। इस कंट्रोलर का काम है कि वह सभी नेटवर्क डिवाइसेस के लिए निर्णय लेता है, जैसे डेटा किस रूट से भेजा जाएगा, किस प्रकार की पॉलिसीज़ लागू होंगी, आदि।

(1): Separation:

- SDN separates control and data planes, meaning data transmission and network decisions happen independently.
- SDN में कंट्रोल और डेटा प्लेन को अलग कर दिया जाता है। इससे डेटा ट्रांसमिशन और नेटवर्क निर्णय अलग-अलग होते हैं।

(2) Centralized Control:

- A single controller manages the whole network, making it easier to make changes and updates across the network.
- नेटवर्क कंट्रोलर पूरे नेटवर्क को एक जगह से नियंत्रित करता है। इससे नेटवर्क में बदलाव या अपडेट्स करना आसान हो जाता है।

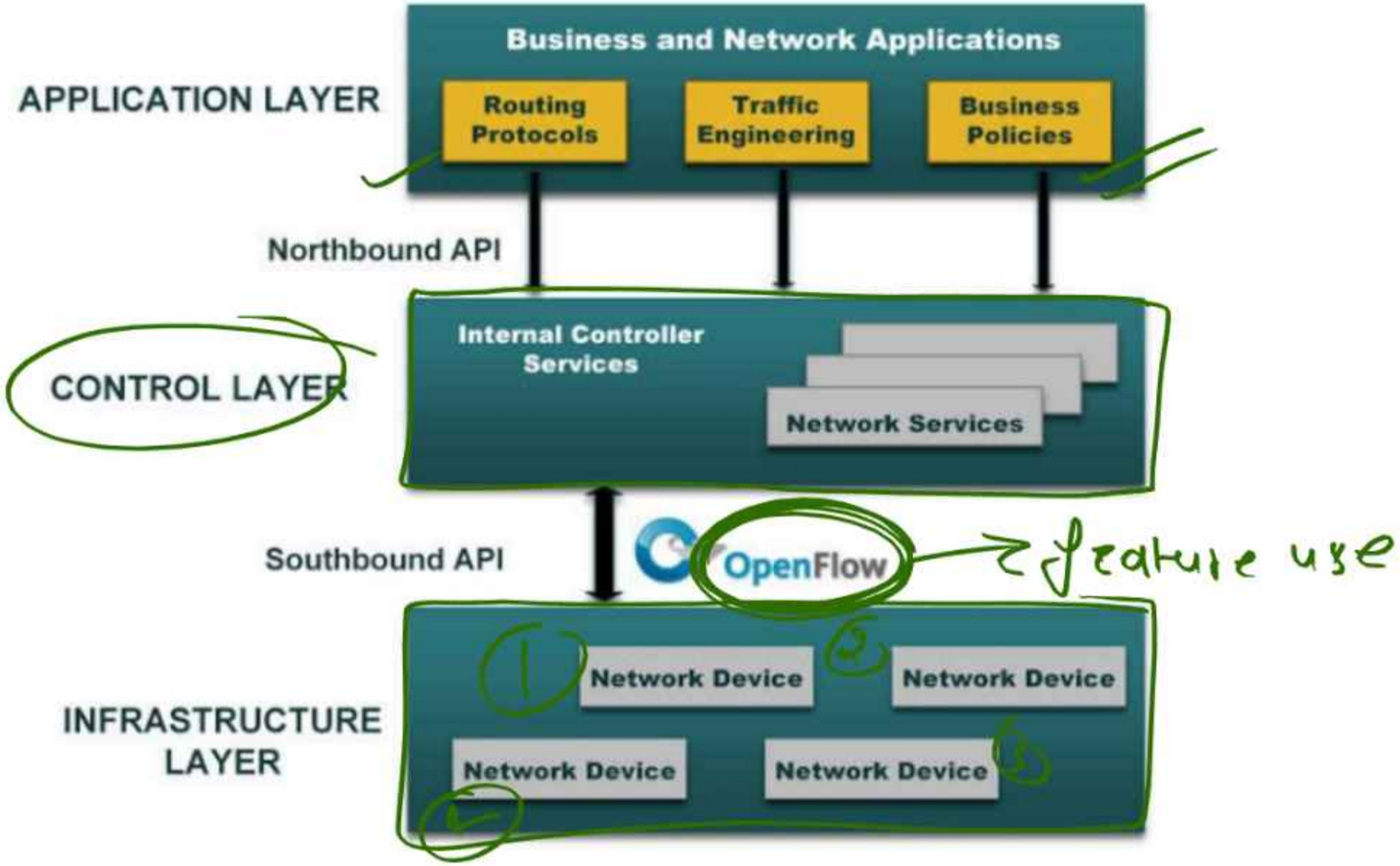
(3): Programmable Network:

- SDN enables network administrators to use software to change network behavior. They can update data flows, security policies, and routing with ease.
- SDN में सॉफ्टवेयर प्रोग्रामिंग का उपयोग करके नेटवर्क के व्यवहार को बदला जा सकता है। इससे एडमिनिस्ट्रेटर को नेटवर्क के फ्लो, सिक्योरिटी पॉलिसी, और रूटिंग को बदलने की सुविधा मिलती है।

SDN architecture :

- SDN (Software Defined Networking) Architecture is divided into three main layers: Infrastructure Layer, Control Layer, and Application Layer. This architecture makes the network flexible, programmable, and centrally managed.
- SDN (Software Defined Networking) आर्किटेक्चर में नेटवर्क को तीन मुख्य परतों (Layers) में विभाजित किया गया है: इंफ्रास्ट्रक्चर लेयर, कंट्रोल लेयर, और एप्लिकेशन लेयर। यह आर्किटेक्चर नेटवर्क को लचीला, प्रोग्रामेबल और सेंट्रलाइज्ड तरीके से प्रबंधित करने की सुविधा देता है।

SDN architecture :



1. इंफ्रास्ट्रक्चर लेयर (Infrastructure Layer):

- Also known as the Data Plane.
- This layer includes physical network devices like routers, switches, and other hardware.
- Its primary function is to send and receive data packets based on the instructions it receives from the controller.
- इसे डेटा प्लेन भी कहा जाता है।
- इसमें नेटवर्क के भौतिक उपकरण जैसे राउटर, स्विच, और अन्य हार्डवेयर डिवाइसेस शामिल होते हैं।
- इस लेयर का काम डेटा पैकेट को भेजना और रिसीव करना होता है। यह कंट्रोलर से निर्देश प्राप्त करती है कि डेटा कैसे और कहाँ भेजा जाए।

2. कंट्रोल लेयर (Control Layer):

- The Control Layer is the brain of the SDN architecture and contains the SDN Controller.
- This layer manages and controls the entire network by making decisions on data routing, policy implementation, and security.
- The Controller decides the direction and handling of network traffic, acting as an intermediary between the Application Layer and the Infrastructure Layer.
- कंट्रोल लेयर SDN आर्किटेक्चर का "मस्तिष्क" होता है और इसमें SDN कंट्रोलर शामिल होता है।
- यह लेयर पूरे नेटवर्क को प्रबंधित और नियंत्रित करती है, डेटा रूटिंग, पॉलिसी लागू करने और सुरक्षा पर निर्णय लेकर।
- कंट्रोलर नेटवर्क ट्रैफिक की दिशा और प्रबंधन का निर्णय करता है, और एप्लिकेशन लेयर और इंफ्रास्ट्रक्चर लेयर के बीच एक मध्यस्थ के रूप में कार्य करता है।

3. एप्लिकेशन लेयर (Application Layer):

- This layer contains various network applications and services, such as firewalls, traffic management, and security policies.
- This layer tells the controller what kind of services and policies to implement.
- The job of the application layer is to communicate with the controller to enhance the functionality of the network.
- इस लेयर में विभिन्न नेटवर्क एप्लिकेशन और सेवाएं होती हैं, जैसे फायरवॉल, ट्रैफिक मैनेजमेंट, और सिक्योरिटी पॉलिसीज़।
- यह लेयर कंट्रोलर को बताती है कि उसे किस तरह की सेवाएं और पॉलिसीज़ लागू करनी हैं।
- एप्लिकेशन लेयर का काम है कि वह नेटवर्क की कार्यक्षमता को बढ़ाने के लिए कंट्रोलर के साथ संपर्क बनाए रखे।

SDN के फायदे (Benefits of SDN):

1. लचीलापन (Flexibility):

- SDN allows network policies and routing to be easily changed through software, making network updates and improvements simpler.
- SDN में सॉफ्टवेयर द्वारा नेटवर्क की पॉलिसीज़ और रूटिंग को आसानी से बदला जा सकता है। यह नेटवर्क को अपडेट करने और सुधारने में मदद करता है।

2. केंद्रकृत नियंत्रण (Centralized Management):

- With SDN, the entire network can be controlled from a single point, simplifying monitoring and management.
- SDN में नेटवर्क का नियंत्रण एक ही सेंट्रल प्वाइंट से होता है। इससे नेटवर्क का मॉनिटरिंग और मैनेजमेंट आसान हो जाता है।

3. कम लागत (Cost Savings):

- SDN reduces dependency on specialized hardware devices, lowering costs, as everything is controlled by software.
- SDN में हार्डवेयर डिवाइसेस पर निर्भरता कम होती है। इससे नेटवर्किंग की लागत में बचत होती है, क्योंकि सब कुछ सॉफ्टवेयर से नियंत्रित होता है।

4. स्केलेबिलिटी (Scalability):

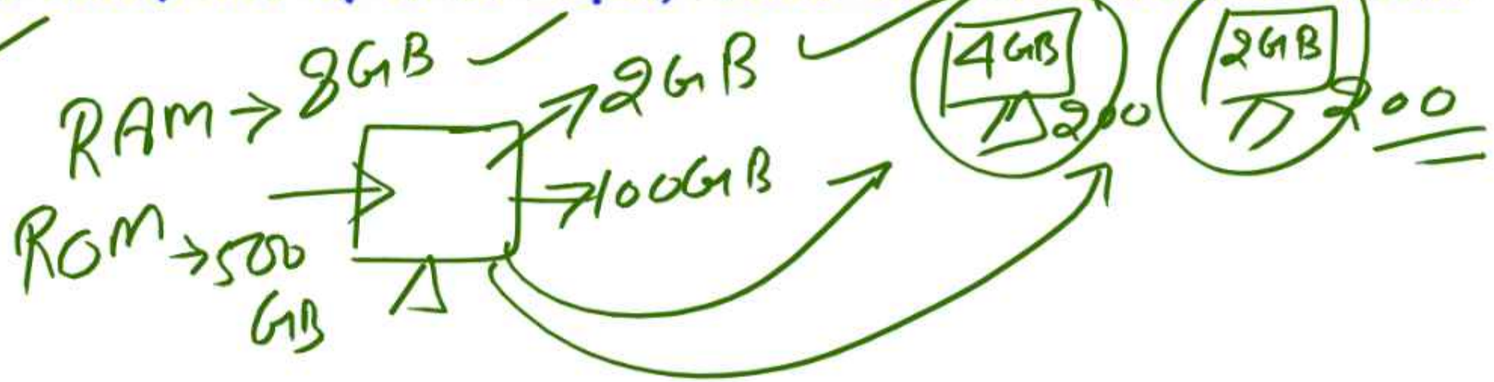
- SDN makes it easy to expand networks by adding new devices and connections.
- SDN में नेटवर्क को बड़े पैमाने पर बढ़ाया जा सकता है। नए डिवाइसेस और कनेक्शन्स को जोड़ना आसान होता है।

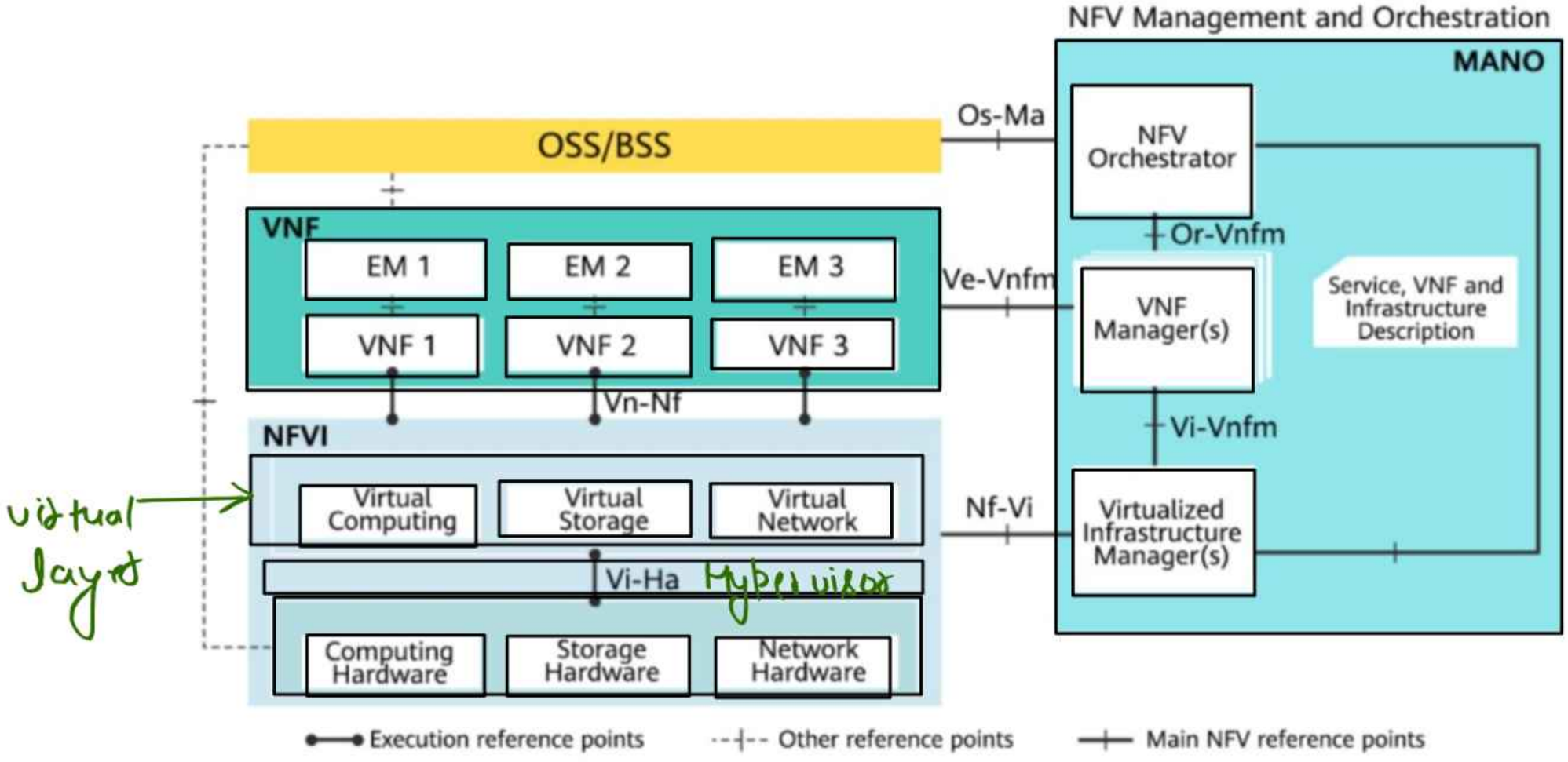
5. सिक्योरिटी (Security):

- Centralized control in SDN allows for stronger network security. Security policies can be implemented and managed from a single location.
- SDN में सेंट्रल कंट्रोल की वजह से नेटवर्क को अधिक सुरक्षित बनाया जा सकता है। सिक्योरिटी पॉलिसीज़ को एक ही जगह से लागू और मैनेज किया जा सकता है।

Network Function Virtualization (NFV):

- The main purpose of NFV is to virtualize various network functions. Traditionally, network devices like routers, firewalls, and load balancers are physical hardware, but with NFV, these are run as software on virtual machines (VMs) or containers.
- NFV का मुख्य उद्देश्य नेटवर्क के विभिन्न कार्यों को वर्चुअलाइज करना है। आमतौर पर नेटवर्क में राउटर, फायरवॉल, लोड बैलेंसर जैसे डिवाइसेस फिजिकल (हार्डवेयर) रूप में होते हैं, लेकिन NFV में इन्हें सॉफ्टवेयर के रूप में वर्चुअल मशीनों (VMs) या कंटेनरों में चलाया जाता है।
- This makes managing the network easier, allowing new features to be added, upgraded, and scaled with flexibility.
- इससे नेटवर्क का प्रबंधन आसान हो जाता है और नए फीचर्स जोड़ना, अपग्रेड करना और स्केलिंग करना सरल हो जाता है।




NFV Management and Orchestration
MANO
OSS/BSS
Os-Ma
NFV Orchestrator
Or-Vnfm
VNF
EM 1
EM 2
EM 3
VNF 1
VNF 2
VNF 3
Ve-Vnfm
VNF Manager(s)
Service, VNF and Infrastructure Description
Vn-Nf
NFVI
Vi-Vnfm
Virtual Computing
Virtual Storage
Virtual Network
Nf-Vi
Virtualized Infrastructure Manager(s)
Vi-Ha
Hypervisor
Computing Hardware
Storage Hardware
Network Hardware
virtual layer
Execution reference points
Other reference points
Main NFV reference points

1. NFVI (Network Function Virtualization Infrastructure):

- The first layer of NFVI consists of hardware resources (CPU), storage resources (Hard disk), and Network resources (Router, Switch, and Firewalls).
- The second layer of NFVI is the virtualization layer, which separates hardware and replaces it with software.
- The third layer of NFVI is virtualized resources such as virtual computers, virtual networks, and virtual storage.
- NFVI की पहली परत में हार्डवेयर संसाधन (CPU), स्टोरेज संसाधन (हार्ड डिस्क) और नेटवर्क संसाधन (राउटर, स्विच और फ़ायरवॉल) शामिल हैं।
- NFVI की दूसरी परत वर्चुअलाइजेशन परत है, जो हार्डवेयर को अलग करती है और इसे सॉफ़्टवेयर से बदल देती है।
- NFVI की तीसरी परत वर्चुअलाइज्ड संसाधन जैसे वर्चुअल कंप्यूटर, वर्चुअल नेटवर्क और वर्चुअल स्टोरेज है।

2. VNFs (Virtual Network Functions)

- VNFs are virtual network functions, such as virtual routers, virtual firewalls, and virtual load balancers.
- These functions are run in separate virtual machines or containers and provide network services.
- Depending on NFVI, VNFs can be quickly configured, deleted, or replaced.
- VNFs वर्चुअल नेटवर्क फंक्शन्स हैं, जैसे कि वर्चुअल राउटर, वर्चुअल फायरवॉल, और वर्चुअल लोड बैलेंसर।
- इन फंक्शन्स को अलग-अलग वर्चुअल मशीनों या कंटेनरों में चलाया जाता है और ये नेटवर्क सेवाएं प्रदान करते हैं।
- NFVI पर निर्भर करते हुए, VNFs को जल्दी से कॉन्फ़िगर किया जा सकता है, हटाया जा सकता है या बदला जा सकता है।

3. NFV MANO (Management and Orchestration)

1. **NFV Orchestrator:** It plans and manages the entire NFV system. It deploys new VNFs, monitors their status, and scales the system.

2. **VNF Manager:** It controls various VNFs, such as their configuration, status monitoring, and updating.

3. **Virtualized Infrastructure Manager (VIM):** It manages the NFVI, such as managing virtual machines, networking, and storage.

1. **NFV Orchestrator:** यह पूरे NFV सिस्टम की योजना और प्रबंधन करता है। यह नए VNFs को तैनात करना, उनकी स्थिति की निगरानी करना और सिस्टम को स्केल करने का काम करता है।

2. **VNF Manager:** यह विभिन्न VNFs को कंट्रोल करता है, जैसे कि उनका कॉन्फ़िगरेशन, स्टेटस , और अपडेट करना।

3. **Virtualized Infrastructure Manager (VIM):** यह NFVI को प्रबंधित करता है, जैसे कि मशीनों का प्रबंधन, नेटवर्किंग, और स्टोरेज का कंट्रोल।

NFV के फायदे (Benefits of NFV):

1. लागत में कमी (Cost Reduction):

- Reduces the need for physical hardware, saving costs.
- फिजिकल हार्डवेयर की आवश्यकता कम हो जाती है, जिससे लागत में कमी आती है।

2. लचीलापन (Flexibility):

- Makes it easy to add new functions and change policies.
- नए फंक्शन्स जोड़ना और पॉलिसी बदलना आसान हो जाता है।

3. तेजी से सेवाएं उपलब्ध कराना (Rapid Service Deployment):

- New network functions can be quickly deployed and configured.
- नए नेटवर्क फंक्शन्स को जल्दी से तैनात और कॉन्फ़िगर किया जा सकता है।

4. स्केलेबिलिटी (Scalability):

- NFV allows the network to scale easily to handle more traffic.
- NFV से नेटवर्क को आसानी से स्केल किया जा सकता है और अधिक ट्रैफिक को संभाला जा सकता है।

5. उपयोग में आसानी (Ease of Management):

- Managing the network through software simplifies operations..
- सॉफ्टवेयर के माध्यम से नेटवर्क का प्रबंधन करना सरल हो जाता है।

WoT (Web of Things):

- WoT, or the Web of Things, is an important concept within IoT (Internet of Things).
- It aims to make IoT devices connect and communicate more effectively using web technologies.
- The primary goal of WoT is to simplify IoT by providing a standardized way to interact with smart devices over the web.

- WoT, या वेब ऑफ थिंग्स (Web of Things), IoT का एक महत्वपूर्ण हिस्सा है। इसे इस तरह समझा जा सकता है कि यह IoT डिवाइसेस को आपस में कनेक्ट करने और संचार(Communication) करने में सक्षम बनाता है।

- WoT का मुख्य उद्देश्य IoT को वेब तकनीकों के माध्यम से और अधिक आसान और उपयोगी बनाना है।

WoT के मुख्य घटक (Key Components of WoT):

- WoT consists of four main components:
- WoT के चार मुख्य घटक होते हैं:

1. थिंग्स डिस्क्रिप्शन (Things Description):

- This is a document written in JSON-LD that describes the features of an IoT device. It includes details about the device's services, actions it can perform, and the data it generates.
- यह JSON-LD में लिखा एक दस्तावेज़ होता है जो IoT डिवाइसेस का विवरण देता है। इसमें उस डिवाइस की सभी जानकारी होती है जैसे कि क्या सेवाएं देता है, क्या एक्शन ले सकता है, और क्या डेटा उत्पन्न करता है।

2. बाइंडिंग टेम्पलेट्स (Binding Templates):

- These templates help connect devices with various communication protocols (like HTTP, MQTT), allowing them to interact seamlessly.
- ये टेम्पलेट्स विभिन्न प्रोटोकॉल्स (जैसे HTTP, MQTT) के साथ डिवाइस को जोड़ने में मदद करते हैं।

3. सिक्योरिटी (Security):

- Security is crucial in WoT. It includes authentication (verifying the user) and authorization (granting access permissions), ensuring safe data usage.
- यह WoT का महत्वपूर्ण हिस्सा है। इसमें प्रमाणन (authentication) और अनुमति (authorization) शामिल है ताकि डेटा का उपयोग सुरक्षित तरीके से हो।

4. स्क्रिप्टिंग API (Scripting API):

- This API provides developers with a way to interact with and control IoT devices easily, making it simpler to build applications.
- डेवलपर्स के लिए एक इंटरफ़ेस है जो उन्हें विभिन्न IoT डिवाइसेस के साथ इंटरैक्ट करने और काम करने में सक्षम बनाता है।

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Definition	Network of physical devices connected to the internet	A framework to make IoT devices communicate via web standard
Purpose	Connect devices to exchange data	Simplify communication among devices using web technologies
Communication	Uses various, often incompatible protocols	Uses web-based, standardized protocols (like HTTP, JSON-LD)

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Definition	इंटरनेट से जुड़े भौतिक उपकरणों का नेटवर्क है	IoT उपकरणों को वेब मानक के माध्यम से संचार करने के लिए एक ढांचा (Framewok) देता है
Purpose	डेटा का आदान-प्रदान करने के लिए डिवाइस कनेक्ट करते है	वेब प्रौद्योगिकियों का उपयोग करके उपकरणों के बीच संचार को सरल बनाता है
Communication	विभिन्न, अक्सर असंगत(Incompatible) प्रोटोकॉल का उपयोग करता है	वेब-आधारित, standard प्रोटोकॉल (जैसे HTTP, JSON-LD) का उपयोग करता है

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Interoperability	Difficult for different devices to work together	Provides a common platform for easy interaction
Security	Security varies by device and protocol	Emphasizes standardized security features
Ease of Use	May require complex integrations	Easier integration and control of devices

IoT और WoT के बीच अंतर (Difference between IoT and WoT)

Aspect	Internet of Things	Web of Things
Interoperability	विभिन्न उपकरणों का एक साथ काम करना कठिन है।	आसान बातचीत के लिए एक साझा platform प्रदान करता है
Security	सुरक्षा डिवाइस और प्रोटोकॉल के अनुसार अलग-अलग होती है	Standard सुरक्षा सुविधाओं पर जोर दिया जाता है।
Ease of Use	जटिल एकीकरण(complex intergration की आवश्यकता हो सकती है	उपकरणों का आसान एकीकरण(integration) और नियंत्रण(Control) करता है।

Chapter No-1 (introduction (IoT)

Q 1: What do you mean by IoT? What are the features of IoT?
Q 1:IoT से आपका क्या अभिप्राय है? IoT की विशेषताएँ क्या हैं?

Q 2: What do you mean by protocol? Explain IoT protocol.
Q 2:प्रोटोकॉल से आपका क्या तात्पर्य है? IoT प्रोटोकॉल की व्याख्या करें।

Q 3: Explain logical and physical design in IoT with their components?
Q 3:उनके घटकों के साथ IoT में तार्किक और भौतिक (logical and physical) डिजाइन की व्याख्या करें?

Q 4: What is REST in REST based communication API?
Q 4:REST आधारित संचार एपीआई में REST क्या है?

Q 5: Name the IoT enabling technology.
Q 5:IoT सक्रिय-प्रौद्योगिकी नामित कीजिये।

Chapter No-2 (IoT Devices)

Q 1: What is a bread board? Also show the connections with a clear diagram.
Q 1: Bread board क्या है? स्पष्ट चित्र के द्वारा Connections को भी दर्शाइये।

Q 2: What are Tri-color and Bi-color LEDs? Explain.
Q 2:Tri-color तथा Bi color LED क्या है? समझाइये।

Q 3: What is the difference between Sensors and Actuators?
Q 3:Sensor एवं Actuators में क्या अन्तर है?

Q 4: What is a sensor? Explain its different types?
Q 4: Sensor क्या है? इसके विभिन्न प्रकारों की व्याख्या कीजिए।

Q 5 What is a Actuator ? Explain its different types?
Q 5:Tri-color तथा Bi color LED क्या है? समझाइये।

Chapter No-3 (IoT Networks)

Q 1: What are the features of Arduino Uno board?
Q 1 आर्डिनो Uno board की विशेषतायें क्या हैं?

Q 2: Compare between IoT networks and wireless networks.
Q 2:IoT नेटवर्क और वायरलेस नेटवर्क के बीच तुलना करें।

Q 3: What is IOT Network? Explain different types of IOT Network Connections.
Q 3:IOT Network क्या है विभिन्न प्रकार के IOT Network Connections को समझाइये।

Q 4: Explain all internet protocols and all data protocols.
Q 4: सभी इंटरनेट प्रोटोकॉल तथा सभी डेटा प्रोटोकॉल की व्याख्या करें।

Q 5: Explain communication Technologies in IoT?
Q 5:IoT में संचार प्रौद्योगिकियों की व्याख्या करें

Chapter No-3 (IoT Networks)

Q 6: Difference between WiFi and Bluetooth.):

Q 6: वाईफाई और ब्लूटूथ के बीच अंतर।

Chapter No-4 (IoT ARDUINO)

Q 1:What is Arduino IDE ? How does Arduino IDE work?
Q 1: Arduino IDE क्या है? Arduino IDE कैसे काम करता है?

Q 2:What is Arduino device? Explain the various pins available in it.
Q 2:Arduino device क्या है? इसमें उपलब्ध विभिन्न pins की व्याख्या कीजिए।

Q 3: Describe the Arduino programming language.
Q 3:आर्डिनो प्रोग्रामिंग-भाषा का वर्णन कीजिये।

Q 4:What is a Control Statement? Give various examples also.
Q 4:Control Statement क्या है? विभिन्न उदाहरण भी दीजिए।

Q 5: Explain the operators used in Arduino IDE.
Q 5:Arduino IDE में प्रयुक्त ऑपरेटरों को समझाएं।

Chapter No-5 (IoT and M2M)

Q 1:Explain the difference between IoT and M2M.
Q 1:IoT एवं M2M में अन्तर की व्याख्या कीजिये।

Q 2: What is the fundamental difference between IoT and WoT?
Q 2:IoT एवं WoT के बीच मौलिक अन्तर क्या है?

Q 3 What is NFV for IoT? Explain.
Q 3:NFV for IoT क्या है? समझाइये।

Q 4: Explain different types of SDN.
Q 3:SDN के विभिन्न प्रकार समझाइये।

Thank You