Notes In Hindi

SUBJECT

INTERNET OF THINGS

PREPARED BY

TEAM NOTESINHINDI

Notesinhindi.com

2026 Edition

Unit 1: Fundamentals of IoT

Introduction to IoT (Definition, Characteristics, Applications, 5-Layer Architecture, Challenges), IoT Devices/Sensors/Actuators/Controllers, Evolution of IoT, Enabling Technologies, M2M Communication, और IoT vs M2M vs WSN

Syllabus — Unit Overview

Unit	Topics and Sub-topics
Unit 1: Fundamentals of IoT	• **Introduction to IoT** — Definition, Characteristics, Applications, 5-layer architecture: Perception, Network, Middleware, Application, Business, IoT challenges • **IoT Devices, Sensor Actuator और Controllers** — IoT devices, sensors, types of sensors, actuators, types of actuators, Controllers (Arduino, Raspberry Pi) • **Evolution of Internet of Things**, Enabling Technologies, M2M Communication, IoT vs M2M vs WSN

Is Unit ki Structure:

1) Introduction to IoT | 2) IoT Devices, Sensors, Actuators और Controllers | 3) Evolution of IoT, Enabling Technologies, M2M Communication, IoT vs M2M vs WSN

1. Introduction to IoT

> **Internet of Things (IoT)** एक ऐसी technology है जिसमें रोज़मर्रा की physical चीज़ों (devices) को internet से जोड़ दिया जाता है, ताकि वो आपस में **data collect, share और exchange** कर सकें — बिना किसी human की सीधी मदद के। आसान भाषा में कहें तो, यह एक तरीका है जिससे साधारण objects (जैसे bulb, fan, watch, fridge) "smart" बन जाते हैं और internet के through एक-दूसरे से बात कर पाते हैं।

- पहले सिर्फ computers और mobile phones ही internet से जुड़ते थे, लेकिन IoT की वजह से अब cars, home appliances, health devices, और industrial machines जैसी चीज़ें भी internet से connect होने लगी हैं।
- हर IoT device में एक **sensor** (जो data collect करता है) और कभी-कभी एक **actuator** (जो physical action perform करता है) लगा होता है।

1.1 Characteristics of IoT

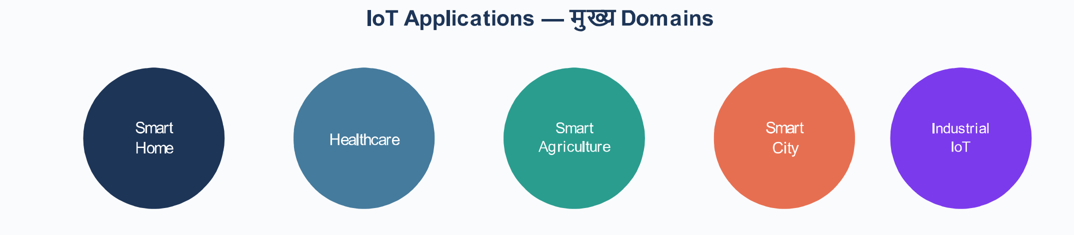
Characteristic	Description
Connectivity	IoT devices हमेशा network से जुड़े रहते हैं, ताकि वो data भेज और प्राप्त कर सकें। बिना connectivity के कोई भी device "smart" नहीं बन सकता।
Intelligence	Devices sensors से collected data को analyze करके smart decisions ले सकते हैं, जैसे temperature ज़्यादा होने पर AC खुद चालू होना।

Dynamic Nature	IoT devices की state समय के साथ बदलती रहती है, जैसे किसी device का on/off होना या location बदलना। System को इन बदलावों को real-time में handle करना पड़ता है।
Scalability	IoT system को आसानी से बढ़ाया जा सकता है — नए devices बिना पूरे system को redesign किए add किए जा सकते हैं।
Heterogeneity	अलग-अलग manufacturers के, अलग-अलग hardware/software वाले devices एक साथ काम कर सकते हैं।
Safety और Security	चूंकि IoT devices sensitive data handle करते हैं, इसलिए data की safety और security बनाए रखना बहुत ज़रूरी होता है।

1.2 Applications of IoT

- **Smart Home:** Smart lights, locks, thermostats आपस में जुड़कर घर को automatically control करते हैं, जैसे कोई ना हो तो lights खुद बंद हो जाएँ।
- **Healthcare:** Wearable devices (जैसे fitness bands) patients के vitals (heart rate, sleep) को continuously monitor करते हैं और doctors को real-time data भेजते हैं।
- **Smart Agriculture:** Soil moisture और weather sensors किसानों को सही समय पर पानी देने और fertilizer इस्तेमाल करने में मदद करते हैं।
- **Smart City:** Traffic management, waste management, और street lighting जैसी civic services को IoT sensors के through optimize किया जाता है।

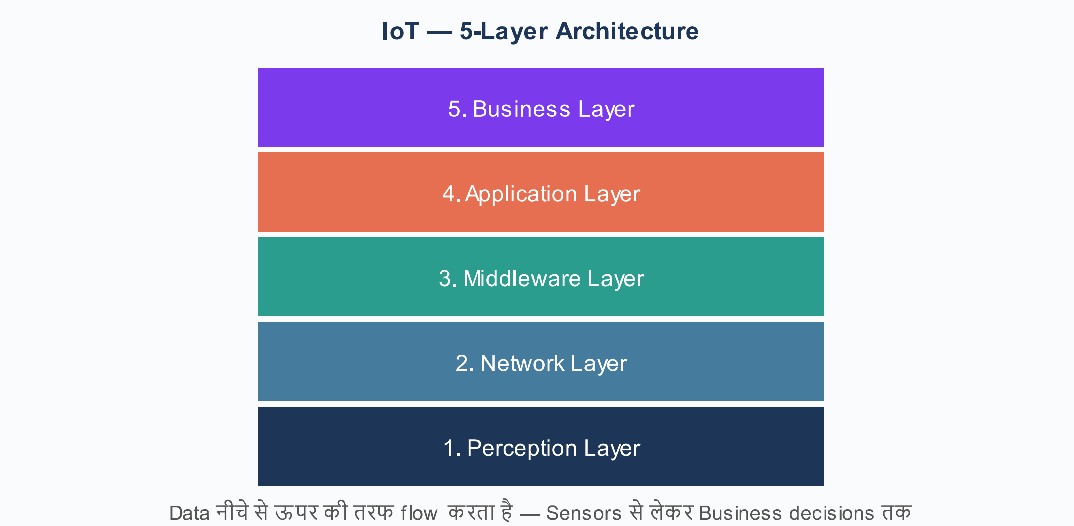
- **Industrial IoT (IIoT):** Factories में machines की health monitor करके, breakdown होने से पहले ही maintenance की जा सकती है (predictive maintenance)।



चित्र 1.1 — IoT का इस्तेमाल आज लगभग हर domain में हो रहा है — घर से लेकर healthcare, agriculture, cities, और industries तक।

1.3 5-Layer Architecture of IoT

> IoT system को design और समझने के लिए उसे **पाँच layers** में divide किया जाता है। हर layer का एक specific role होता है, जिससे data physical world से लेकर business-level decisions तक व्यवस्थित तरीके से पहुँचता है।



चित्र 1.2 — IoT की 5-Layer Architecture में data सबसे नीचे वाली Perception Layer से शुरू होकर, ऊपर की Business Layer तक पहुँचता है।

1.3.1 Perception Layer

> यह सबसे नीचे वाली, सबसे basic layer है, जिसमें actual physical devices — **sensors और actuators** — शामिल होते हैं। इसका काम physical world से data collect करना (sensing) और ज़रूरत पड़ने पर action perform करना (actuation) है।

- यह layer temperature, humidity, motion जैसी real-world जानकारी को digital signal में convert करती है, ताकि उसे आगे भेजा जा सके।
- इस layer में RFID tags, GPS modules, और camera जैसे identification/sensing components भी शामिल होते हैं, जो devices को uniquely पहचानने या उनकी location track करने में मदद करते हैं।
- यह पूरे IoT system की सबसे बुनियादी (foundational) layer है — अगर यहाँ का data ही गलत या अधूरा हो, तो ऊपर की सभी layers की performance प्रभावित होती है।

1.3.2 Network Layer

> Network Layer, Perception Layer से आए हुए data को **transmit** करने का काम करती है — यह data को internet, Wi-Fi, Bluetooth, या cellular network के through आगे भेजती है।

- यह layer devices को gateway या cloud से जोड़ती है, ताकि data आगे processing के लिए पहुँच सके।
- इस layer में Wi-Fi, Zigbee, Bluetooth जैसे short-range protocols और 4G/5G, LPWAN जैसी long-range technologies इस्तेमाल होती हैं, जो data की distance और speed की ज़रूरत के हिसाब से चुनी जाती हैं।
- Network Layer यह भी सुनिश्चित करती है कि data बिना किसी loss या corruption के अपने destination तक सही तरीके से पहुँचे।

1.3.3 Middleware Layer

> Middleware Layer वह layer है, जो आए हुए raw data को **process, store, और manage** करती है। यह अलग-अलग devices से आए data को एक जगह इकट्ठा करके, उसे meaningful information में बदलती है।

- यह layer database management, data aggregation, और device management जैसी services provide करती है।
- यहीं पर cloud platforms, APIs, और message-queuing systems (जैसे MQTT broker) काम करते हैं, जो अलग-अलग applications को एक-दूसरे के साथ data share करने में मदद करते हैं।
- Middleware Layer को कई बार IoT system की "backbone" कहा जाता है, क्योंकि यही layer raw data को actual, इस्तेमाल करने योग्य information में बदलती है।

1.3.4 Application Layer

> Application Layer वह layer है, जिससे end-user directly interact करता है — जैसे mobile app या web dashboard, जहाँ processed information दिखाई जाती है।

- यह layer domain-specific services (जैसे smart home app, health monitoring app) provide करती है।
- इसी layer में user को notifications, reports, और controls (जैसे light on/off का button) दिखाए जाते हैं, ताकि वह system के साथ आसानी से interact कर सके।

1.3.5 Business Layer

> सबसे ऊपर वाली Business Layer, collected data और application के इस्तेमाल का analysis करके business decisions, reports, और future strategies बनाने में मदद करती है।

- यह layer यह देखती है कि पूरा IoT system business goals (जैसे cost कम करना, efficiency बढ़ाना) को कितना पूरा कर रहा है।

- यहाँ पर Business Intelligence (BI) tools और analytics dashboards इस्तेमाल होते हैं, जो management को आगे के फैसले लेने में मदद करते हैं, जैसे किसी नए feature में invest करना है या नहीं।

1.4 IoT Challenges

- **Security और Privacy:** बहुत सारे devices जुड़े होने की वजह से hacking और data leakage का खतरा बढ़ जाता है।
- **Interoperability:** अलग-अलग manufacturers के devices और protocols आपस में हमेशा आसानी से काम नहीं कर पाते।
- **Scalability:** जैसे-जैसे devices की संख्या बढ़ती है, network और infrastructure को उतना ही scale करना पड़ता है, जो costly हो सकता है।
- **Power Consumption:** छोटे, battery से चलने वाले devices की battery life लंबी रखना एक बड़ी technical चुनौती है।
- **Data Management:** IoT devices से इतना ज़्यादा data generate होता है कि उसे store, process, और analyze करना मुश्किल हो जाता है।

IoT के Advantages

- Real-time monitoring और automation से मानव effort काफी कम हो जाता है।
- Data-driven decisions लेने से efficiency और accuracy दोनों बढ़ती हैं।
- Remote locations से भी devices को control और monitor किया जा सकता है।

Disadvantages

- ज़्यादा devices जुड़े होने की वजह से security risks बढ़ जाते हैं।
- Setup और maintenance cost काफी ज़्यादा हो सकता है।
- Internet connection चले जाने पर पूरा system प्रभावित हो सकता है।

Real-life Example: एक smart fitness band पूरे दिन heart rate, steps, और sleep pattern track करता है (Perception Layer), Bluetooth/Wi-Fi से data

भेजता है (Network Layer), cloud पर data process होता है (Middleware Layer), और mobile app पर आपको report दिखती है (Application Layer)।

2. IoT Devices, Sensor, Actuator और Controllers

> कोई भी IoT system मुख्य रूप से तीन तरह के hardware components से मिलकर बनता है — **Sensors** (जो data collect करते हैं), **Actuators** (जो physical action perform करते हैं), और **Controllers** (जो पूरे device को control और process करते हैं)।

2.1 IoT Devices — Introduction

> **IoT Device** वह कोई भी physical hardware है, जिसमें sensors, थोड़ी-बहुत processing capability, और एक communication module लगा होता है, जिससे वह internet या local network से जुड़कर data collect और exchange कर सके।

- IoT devices आकार और complexity में बहुत अलग-अलग हो सकते हैं — एक छोटे smart bulb से लेकर बड़ी industrial machine तक, दोनों को IoT device कहा जा सकता है, बशर्ते वे network से जुड़े हों।
- हर IoT device का एक basic structure होता है — पहले वह environment से data collect करता है (sensor की मदद से), फिर उस data को थोड़ा process करता है, और अंत में उसे network के through आगे भेज देता है।
- इन devices को समझने के लिए इन्हें तीन मुख्य components में divide किया जाता है — **Sensors** (जो data इकट्ठा करते हैं), **Actuators** (जो physical action perform करते हैं), और **Controllers** (जो पूरे device को manage करते हैं) — जिन्हें अब हम विस्तार से समझेंगे।

2.2 Sensors

> **Sensor** एक ऐसा electronic component है, जो अपने आस-पास के environment में मौजूद किसी physical quantity (जैसे तापमान, नमी, प्रकाश, गति) को precisely measure करता है और उसे एक electrical signal में convert कर देता है, ताकि उस जानकारी को आगे किसी controller या system द्वारा process किया जा सके।

- Sensors किसी भी IoT system की "आँखें और कान" कहलाए जा सकते हैं, क्योंकि यही वे component हैं जिनके through system को बाहरी दुनिया की जानकारी मिलती है।
- अलग-अलग applications की ज़रूरत के हिसाब से अलग-अलग तरह के sensors design किए गए हैं, जिनमें से कुछ सबसे common types नीचे विस्तार से समझाए गए हैं।

2.2.1 Temperature Sensor

यह sensor अपने आस-पास के तापमान (temperature) को precisely measure करता है और उसे एक readable, digital value में convert कर देता है। इसका इस्तेमाल वहाँ किया जाता है जहाँ heat level पर नज़र रखना ज़रूरी हो, जैसे Smart AC systems में room temperature detect करना, या weather monitoring stations में मौसम का तापमान track करना।

2.2.2 Humidity Sensor

यह sensor हवा में मौजूद नमी (moisture) की मात्रा को measure करता है। यह अक्सर temperature sensor के साथ मिलकर काम करता है, ताकि तापमान और नमी दोनों की सही जानकारी मिल सके। इसका मुख्य इस्तेमाल Smart Agriculture और Greenhouse control में होता है, जहाँ पौधों के लिए सही humidity level maintain करना बहुत ज़रूरी होता है।

2.2.3 Motion / PIR Sensor

Motion Sensor (जिसे PIR — Passive Infrared — Sensor भी कहा जाता है) अपने आस-पास किसी वस्तु या व्यक्ति की movement को detect करता है, यह body से निकलने वाली infrared heat को sense करके काम करता है। इसका सबसे common

इस्तेमाल Security Alarm Systems में होता है, जहाँ कोई अनजान movement पकड़ में आने पर alert जनरेट हो जाता है, और Automatic Lights में, जहाँ किसी के आने पर light अपने-आप जल जाती है।

2.2.4 Light Sensor (LDR)

Light Dependent Resistor (LDR) surrounding environment में मौजूद light की intensity को measure करता है। जैसे-जैसे रोशनी कम या ज़्यादा होती है, इस sensor का electrical resistance भी उसी हिसाब से बदलता है। इसका सबसे practical उदाहरण Street Lights हैं, जो शाम को अंधेरा होते ही automatically जल जाती हैं और सुबह रोशनी होते ही खुद बंद हो जाती हैं, जिससे बिजली की बचत होती है।

2.2.5 Gas Sensor

Gas Sensor हवा में मौजूद harmful या hazardous gases (जैसे LPG, Carbon Monoxide, Methane) की मौजूदगी को detect करता है। यह Industrial Safety Systems में बहुत महत्वपूर्ण component माना जाता है, जहाँ किसी gas leakage को समय रहते पकड़कर बड़ी दुर्घटना को टाला जा सकता है, जैसे घरों और factories में LPG leakage detection systems।

2.2.6 Pressure Sensor

यह sensor किसी liquid या gas के pressure को measure करता है। इसका इस्तेमाल Tyre Pressure Monitoring Systems (TPMS) में होता है, जो गाड़ी के tyres में हवा का pressure लगातार track करते हैं, और Water Tank Systems में, जहाँ यह टैंक में पानी के level को पानी के pressure के आधार पर measure करता है।

Type of Sensor	क्या Measure करता है	Use-case Example
Temperature Sensor	तापमान (Heat level)	Smart AC, weather monitoring
Humidity Sensor	हवा में नमी की मात्रा	Smart agriculture, greenhouse control

Motion/PIR Sensor	किसी वस्तु या व्यक्ति की movement	Security alarm systems, automatic lights
Light Sensor (LDR)	Surrounding light की intensity	Street lights जो अंधेरा होने पर automatically जल जाती हैं
Gas Sensor	हवा में harmful gases की मौजूदगी	Industrial safety, LPG leakage detection
Pressure Sensor	Liquid/gas का pressure	Tyre pressure monitoring, water tanks

2.3 Actuators

> **Actuator** एक ऐसा electromechanical component है, जो controller से प्राप्त एक electrical control signal को किसी वास्तविक, physical movement या action में convert कर देता है — जैसे किसी motor को घुमाना, किसी valve को खोलना, या किसी light को जलाना। संक्षेप में कहें तो, actuator किसी भी IoT system का वह हिस्सा है, जो निर्णय (decision) को असली दुनिया में लागू करता है।

- अगर sensor को system की "आँखें" कहा जाता है, तो actuator को system के "हाथ-पैर" कहा जा सकता है, क्योंकि यही component actual physical बदलाव लाता है।
- नीचे कुछ सबसे प्रचलित (widely used) actuator types को विस्तार से समझाया गया है।

2.3.1 Motor

Motor एक electromechanical actuator है, जो electrical energy को rotational (घूर्णन) movement में convert करता है। इसका इस्तेमाल वहाँ होता है जहाँ किसी चीज़ को घुमाना या move कराना हो, जैसे Fan Speed Control में पंखे की गति को control करना, या Automatic Doors में दरवाज़ों को खोलने-बंद करने के लिए।

2.3.2 Relay

Relay एक electrically operated switch है, जो एक छोटे control signal (जैसे microcontroller से आने वाले low-voltage signal) का इस्तेमाल करके एक बड़े electrical circuit को on या off करता है। इसका मुख्य फायदा यह है कि यह low-power controller को high-power appliances (जैसे घर की lights या heavy machines) को safely control करने में सक्षम बनाता है, जैसे किसी smart plug के through appliances को remotely switch करना।

2.3.3 Solenoid Valve

Solenoid Valve एक ऐसा actuator है, जो electrical signal मिलने पर किसी liquid या gas के flow के रास्ते को खोलता या बंद करता है। यह उन systems में बहुत उपयोगी है जहाँ पानी या gas की supply को automatically control करना हो, जैसे Automatic Irrigation Systems में, जहाँ मिट्टी में नमी कम होने पर valve अपने-आप खुल जाता है और खेत को पानी मिलना शुरू हो जाता है।

2.3.4 LED / Buzzer

LED (Light Emitting Diode) और Buzzer साधारण, लेकिन बहुत उपयोगी actuators हैं, जो किसी event के होने पर visual (रोशनी) या sound (आवाज़) के रूप में alert देते हैं। इनका इस्तेमाल Notifications और Alarms में होता है, जैसे किसी security system में unauthorized entry होने पर buzzer का बज उठना, या किसी device की status दिखाने के लिए LED का जलना।

Type of Actuator	क्या काम करता है	Use-case Example
Motor	Rotational movement पैदा करता है	Fan speed control, automatic doors

Relay	Electrical circuit को on/off करता है	Lights, appliances को remotely switch करना
Solenoid Valve	Liquid/gas के flow को control करता है	Automatic irrigation systems
LED/Buzzer	Visual या sound alert देता है	Notifications और alarms

याद रखें: Sensor हमेशा data "input" करता है (physical world से digital signal की तरफ), जबकि Actuator हमेशा "output" perform करता है (digital signal से physical action की तरफ)।

2.4 Controllers (Arduino, Raspberry Pi)

Controller वह component है जो sensors से data पढ़ता है, उसे process करता है, और actuators को control signal भेजता है — यह पूरे IoT device का "brain" होता है। दो सबसे popular controllers हैं — **Arduino** और **Raspberry Pi**।

2.4.1 Arduino

Arduino एक **microcontroller-based** open-source board है, जो simple, low-power tasks (जैसे sensor पढ़ना, actuator control करना) perform करने के लिए design किया गया है।

- यह beginners के लिए बहुत आसान है और simple, single-task projects के लिए perfect है।
- इसमें कोई operating system नहीं होता — यह सीधे लिखा गया program (sketch) execute करता है।

2.4.2 Raspberry Pi

> Raspberry Pi एक **microprocessor-based mini-computer** है, जो एक पूरे operating system (जैसे Linux) को run कर सकता है और ज़्यादा complex, resource-heavy tasks perform कर सकता है।

- यह multiple applications एक साथ चला सकता है, जैसे web server होस्ट करना या image processing करना।
- इसमें built-in networking, USB ports, और ज़्यादा processing power होती है, जिससे complex IoT projects बनाए जा सकते हैं।

Basis	Arduino	Raspberry Pi
Type	Microcontroller board	Microprocessor-based mini-computer
Operating System	कोई नहीं (direct program run होता है)	Linux-based OS चलाता है
Complexity	Simple, single-task projects	Complex, multi-tasking projects
Power Consumption	बहुत कम	Comparatively ज़्यादा
Cost	सस्ता	थोड़ा महंगा

Arduino के Advantages

- सीखना और इस्तेमाल करना बहुत आसान है, beginners के लिए best choice है।
- Power consumption बहुत कम होती है, इसलिए battery-operated projects के लिए उपयुक्त है।

Raspberry Pi के Advantages

- ज़्यादा processing power की वजह से complex, multi-tasking applications चला सकता है।
- Built-in networking और multiple ports की वजह से ज़्यादा flexible है।

Real-life Example: एक automatic plant-watering system में Arduino board soil moisture sensor से data पढ़ता है और पानी की motor (actuator) को चालू कर देता है — यह एक simple, single-task project है, जबकि एक home-security system जिसमें camera, face-recognition, और cloud storage तीनों साथ चलने चाहिए, वहाँ Raspberry Pi ज़्यादा उपयुक्त रहेगा।

3. Evolution of IoT, Enabling Technologies, M2M Communication, IoT vs M2M vs WSN

3.1 Evolution of Internet of Things

> IoT आज जो रूप है, वह एक दिन में नहीं बना — यह कई दशकों में धीरे-धीरे विकसित (evolve) हुआ है, जैसे-जैसे networking, sensors, और computing technologies बेहतर होती गईं।

चित्र 3.1 — IoT धीरे-धीरे connected machines के idea से लेकर, आज billions of smart devices तक विकसित हुआ है।

- **शुरुआती दौर (1980s-90s):** Machines को network से जोड़ने के early experiments हुए, जैसे एक coke मशीन जिसे इंटरनेट से जोड़कर उसका stock check किया जाता था।
- **"IoT" term की शुरुआत (1999):** Kevin Ashton ने पहली बार "Internet of Things" शब्द का इस्तेमाल किया, RFID technology के context में।
- **Growth Phase (2000s-2010s):** Smartphones, Wi-Fi, और सस्ते sensors की वजह से IoT devices बनाना आसान और सस्ता हो गया।
- **वर्तमान (Present):** आज billions of devices — smart homes, wearables, industrial machines — आपस में जुड़े हुए हैं, और AI/Cloud जैसी technologies के साथ मिलकर काम कर रहे हैं।

3.2 Enabling Technologies

> IoT को actual रूप में possible बनाने के लिए केवल sensors लगा देना काफी नहीं है — इसके पीछे कई अलग-अलग technologies मिलकर काम करती हैं, जो data collection से लेकर उसे meaningful information में बदलने तक की पूरी journey को संभव बनाती हैं।

3.2.1 Wireless Sensor Networks (WSN)

WSN बहुत सारे छोटे-छोटे sensors का एक network होता है, जो किसी विशेष area में फैले होते हैं और आपस में wirelessly communicate करके environment से data collect करते हैं। यह IoT की नींव मानी जा सकती है, क्योंकि ज़्यादातर IoT deployments (जैसे smart agriculture, forest monitoring) इसी WSN structure पर based होते हैं।

3.2.2 Cloud Computing

Cloud Computing IoT devices से आने वाले बड़ी मात्रा में data को store, manage, और process करने के लिए एक scalable, remote computing infrastructure provide करता है। इसका सबसे बड़ा फायदा यह है कि organizations को अपना खुद का महंगा hardware खरीदने की ज़रूरत नहीं पड़ती, और ज़रूरत बढ़ने पर storage/processing power को आसानी से बढ़ाया जा सकता है।

3.2.3 Big Data Analytics

IoT devices हर सेकंड बहुत बड़ी मात्रा में data generate करते हैं, जिसे साधारण tools से analyze करना संभव नहीं होता। Big Data Analytics इसी huge, complex data में से useful patterns, trends, और insights निकालने का काम करता है, जिससे organizations को बेहतर, data-driven decisions लेने में मदद मिलती है।

3.2.4 Communication Protocols

Communication Protocols वे rules और standards होते हैं, जो यह तय करते हैं कि devices आपस में और internet से किस तरीके से data exchange करेंगे। MQTT और CoAP जैसे lightweight protocols खासतौर पर उन IoT devices के लिए बनाए गए हैं, जिनकी power और bandwidth सीमित होती है, ताकि communication efficient बना रहे।

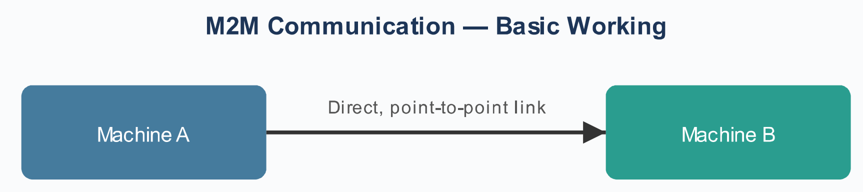
3.2.5 Embedded Systems

Embedded System एक छोटा, dedicated computing system होता है, जो किसी specific task को perform करने के लिए design किया जाता है — जैसे किसी sensor से data पढ़ना और उसके अनुसार actuator को control करना। यही embedded systems हर IoT device के अंदर "छिपा हुआ brain" होते हैं, जो पूरे device के operation को manage करते हैं।

Technology	भूमिका (Role)
Wireless Sensor Networks (WSN)	बहुत सारे छोटे sensors का network, जो मिलकर environment से data collect करते हैं और wirelessly communicate करते हैं।
Cloud Computing	IoT devices से आने वाले बड़ी मात्रा में data को store और process करने के लिए scalable storage और computing power provide करता है।
Big Data Analytics	IoT devices से collect हुए huge data को analyze करके useful patterns और insights निकालने में मदद करता है।
Communication Protocols	Devices के बीच data transfer के rules तय करते हैं (जैसे MQTT, CoAP), ताकि reliable communication हो सके।
Embedded Systems	Sensors और actuators को control करने वाले छोटे, dedicated computing systems, जो हर IoT device के अंदर लगे होते हैं।

3.3 M2M Communication

M2M (Machine-to-Machine) Communication वह technology है, जिसमें दो या ज़्यादा machines बिना किसी human intervention के आपस में सीधे data exchange करती हैं — आमतौर पर एक fixed, point-to-point connection के through।



चित्र 3.2 — M2M Communication में दो machines सीधे, बिना human/internet के बीच में आए, आपस में data exchange करती हैं।

- M2M ज़्यादातर wired या cellular connections इस्तेमाल करता है, और आमतौर पर एक ही तरह के machines के बीच होता है।
- यह IoT से पहले की technology मानी जाती है, जो आगे चलकर IoT के concept में विकसित हुई।

3.3.1 M2M कैसे काम करता है (Working Process)

1. पहली machine (जैसे कोई sensor-equipped device) अपने environment से data collect करती है, जैसे किसी vending machine में stock का level।
2. यह data एक pre-defined communication link (wired, cellular, या satellite) के through सीधे दूसरी machine या central server तक भेजा जाता है।
3. Receiving machine उस data को process करती है और अगर ज़रूरत हो तो एक automatic action ले लेती है — जैसे supplier को "stock खत्म हो रहा है" का alert भेजना।
4. यह पूरा process बिना किसी human के बीच में आए, machines के बीच सीधे होता है — इसीलिए इसे "Machine-to-Machine" कहा जाता है।

M2M Communication के Advantages

- Direct connection होने की वजह से communication fast और reliable होता है।
- Industrial applications (जैसे vending machines, ATMs) में बहुत उपयोगी है।

Disadvantages

- यह ज़्यादातर point-to-point होता है, इसलिए बड़े, flexible networks बनाना मुश्किल होता है।
- यह internet-based नहीं होता, इसलिए cloud/analytics जैसी advanced services इस्तेमाल करना सीमित होता है।

3.4 IoT vs M2M vs WSN

Basis	IoT	M2M	WSN
Full Form	Internet of Things	Machine-to-Machine	Wireless Sensor Network
Connectivity	Internet-based, बड़ा network	ज़्यादातर point-to-point (fixed link)	सिर्फ sensors का local wireless network
Scale	बहुत बड़ा (billions of devices)	सीमित (एक-दूसरे से जुड़ी specific machines)	सीमित area में सैकड़ों sensors
Human Interaction	Data cloud तक जाकर user तक पहुँचता है	पूरी तरह automated, बहुत कम human involvement	Data आमतौर पर एक central node तक जाता है

Use Case	Smart homes, smart cities, healthcare	Vending machines, ATMs, industrial equipment	Environmental monitoring, agriculture

Exam Tip: WSN को IoT का एक हिस्सा माना जा सकता है (sensors का network), M2M को IoT से पहले की, ज़्यादा limited technology माना जाता है, जबकि IoT इन दोनों को मिलाकर, internet और cloud के साथ जोड़कर एक बहुत बड़ा, flexible ecosystem बनाता है।

Real-life Example: एक vending machine जो सीधे supplier के server को "stock खत्म हो गया" का signal भेजती है, वह M2M का उदाहरण है; खेत में लगे कई sensors जो आपस में मिलकर मिट्टी की नमी track करते हैं, वह WSN का उदाहरण है; और जब यही data internet के through एक mobile app तक पहुँचकर किसान को दिखता है, तो वह पूरा system IoT कहलाता है।