Unit -1

Introduction to IoT

- Introduction to the Internet of Things (IoT)
- History and Evolution of IoT
- Key Concepts and Definitions
- Applications and Use Cases of IoT
- Challenges and Opportunities in IoT

Introduction to Internet of Things (IoT)

IoT stands for Internet of Things. It refers to the interconnectedness of physical devices, such as appliances and vehicles, that are embedded with software, sensors, and connectivity which enables these objects to connect and exchange data. This technology allows for the collection and sharing of data from a vast network of devices, creating opportunities for more efficient and automated systems.

Internet of Things (IoT) is the networking of physical objects that contain electronics embedded within their architecture in order to communicate and sense interactions amongst each other or with respect to the external environment. In the upcoming years, IoT-based technology will offer advanced levels of services and practically change the way people lead their daily lives. Advancements in medicine, power, gene therapies, agriculture, smart cities, and smart homes are just a few of the categorical examples where IoT is strongly established. IOT is a system of interrelated things, computing devices, mechanical and digital machines, objects, animals, or people that are provided with unique identifiers. And the ability to transfer the data over a network requiring human-to-human or human-to-computer interaction.

History and Evolution of IoT

The history and evolution of the Internet of Things (IoT) is a fascinating journey of technological advancement, where connected devices have become integral to various aspects of our daily lives. Here's a brief overview:

Early history

- **1960s**: Researchers began exploring ways to connect computers and systems
- **1970s**: The first wireless networks were developed
- **1980s**: The first commercial cellular networks were launched
- **1990s**: The first internet-connected devices appeared

The term "IoT" is coined

- In 1999, Kevin Ashton of Procter & Gamble coined the term "Internet of Things"

- Ashton proposed using radio-frequency identification (RFID) chips to track products through a supply chain

IoT becomes more widely used

- **2000s**: The number of connected devices increased rapidly
- **2010s**: IoT became a major force in the consumer market
- **2020s**: 5G networks began to be rolled out

Examples of IoT devices Smart refrigerators, Smart thermostats, Wearable computers, and Smart doorbells.

Organizations involved in IoT

- **Internet Protocol for Smart Objects Alliance (IPSO)**: A non-profit organization that works to discover and implement new ideas in the field of IoT

Key Concepts and Definitions

The Internet of Things (IoT) is a network of physical objects that are connected to the internet and can exchange data with each other. IoT devices are embedded with sensors, software, and other technologies.

Key concepts

- **Communication**: IoT devices can communicate with each other and with other internet-enabled devices.
- **Data collection**: IoT devices can collect data from their environment and share it with other devices.
- **Self-reporting**: IoT devices can report on themselves and respond to users in real time.
- **Network connectivity**: IoT devices can connect to a network and be individually addressable.

Examples of IoT devices

- Smart home devices like thermostats
- Wearables like smartwatches
- Industrial machinery
- Transportation systems
- Farm animals with biochip transponders
- Cars with built-in sensors

Applications of IoT

- **Smart cities**

  IoT sensors can provide citizens with services like environmental monitoring and parking applications.

- **Agriculture**

  IoT can monitor and collect agricultural data to improve product quality and minimize waste.

- **Transportation**
  IoT can help make transportation more efficient with smart traffic control systems, smart parking systems, and more.

Applications and Use Cases of IoT

The Internet of Things (IoT) has a wide range of applications across various industries, including healthcare through wearables monitoring vital signs, smart homes with automated controls, asset tracking in manufacturing, predictive maintenance for machinery, supply chain management for product tracking, smart agriculture monitoring soil conditions, autonomous driving with sensor data, and even waste management with tracking systems for efficient disposal; essentially, any scenario where connected devices can collect and analyze data to optimize processes and improve decision-making.

Key IoT application areas:

- **Healthcare:**

  Wearable devices like fitness trackers monitoring heart rate, blood pressure, and activity levels; remote patient monitoring for chronic conditions.

- **Smart Homes:**

  Automated lighting, temperature control, security systems, appliance management through connected devices.

- **Manufacturing:**

  Predictive maintenance by analyzing machine health data to predict potential failures, asset tracking to monitor equipment location and status.

- **Supply Chain Management:**

  Real-time tracking of goods throughout the supply chain, optimizing inventory and delivery times.

- **Logistics and Transportation:**

  Vehicle tracking, route optimization, fleet management, driver behavior monitoring.

- **Agriculture:**

  Soil moisture sensors, weather monitoring systems, irrigation control for optimized crop yield.

- **Retail:**

  Smart shelves monitoring inventory levels, customer behavior analysis through IoT-enabled devices.

- **Energy Management:**

  Smart grids with real-time monitoring of energy consumption, optimizing power distribution.

- **Environmental Monitoring:**

  Air quality sensors, water quality monitoring, wildlife tracking.

- **Public Safety:**
  Emergency response systems, crowd monitoring, surveillance cameras.

Important aspects of IoT applications:

- **Sensors:**

  Devices that collect data from the physical environment, like temperature, pressure, motion, and light.

- **Connectivity:**

  Networks like Wi-Fi, Bluetooth, cellular networks to transmit data from sensors to the cloud.

- **Data Analytics:**

  Processing and interpreting large volumes of sensor data to extract meaningful insights.

- **Cloud Computing:**
  Storage and processing of IoT data on remote servers.

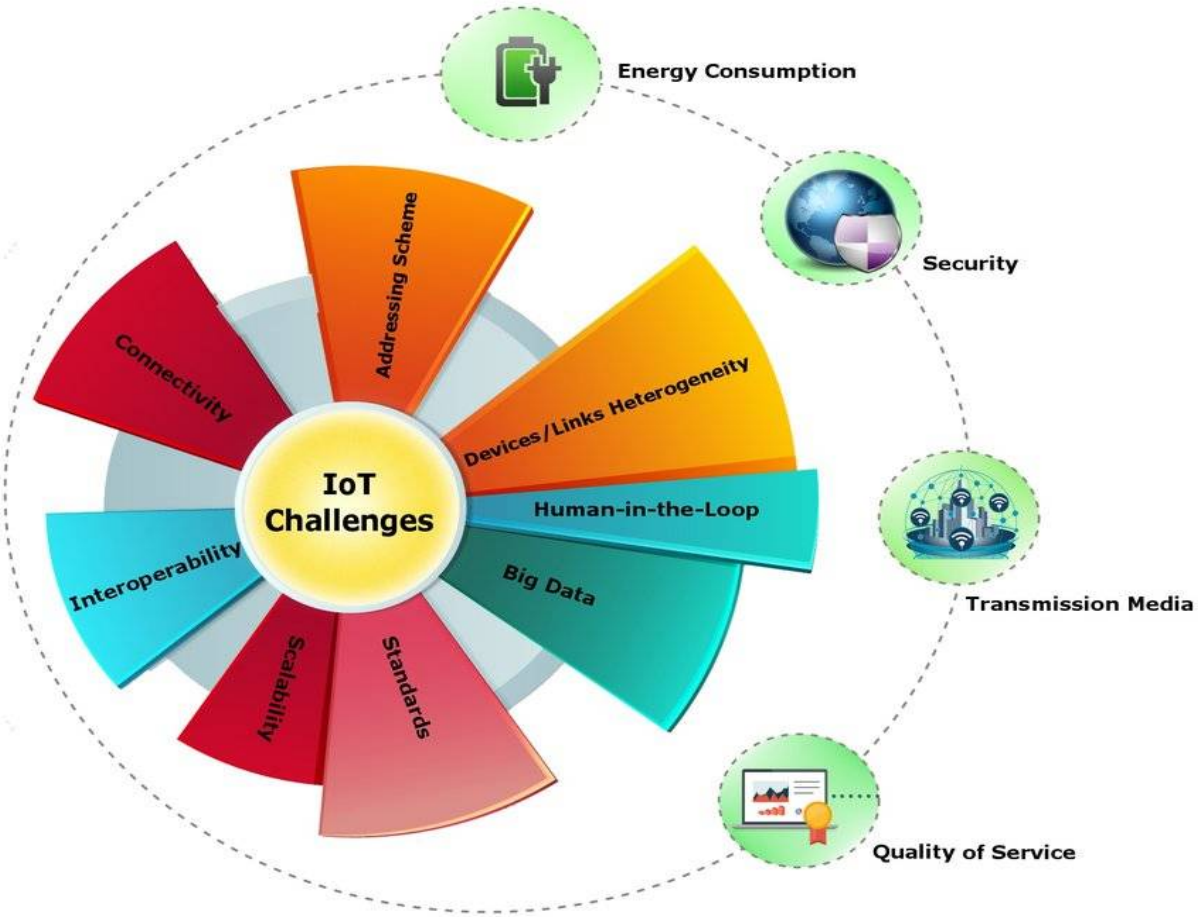
The 10 most adopted IoT use cases

- Process automation. Process automation definition (by IoT Analytics) ...
- Quality control and management. ...
- Energy monitoring. ...
- Real-time inventory management. ...
- Supply chain track and trace. ...
- Operations planning and scheduling. ...
- On-site facility track and trace. ...
- Asset performance optimization.

Challenges and Opportunities in IoT

Challenges in IoT

Key challenges in the Internet of Things (IoT) include: data privacy concerns, interoperability issues between diverse devices, scalability limitations as the number of connected devices grows, security vulnerabilities, managing large volumes of data, ensuring reliable connectivity in remote locations, and potential for unauthorized access to sensitive information collected by IoT devices; all of which require robust security measures and standardized protocols to address effectively.



Breakdown of major IoT challenges:

- **Data Privacy:**

  Protecting sensitive personal data collected by IoT devices from unauthorized access or breaches is a critical challenge, especially when dealing with health or location data.

- **Interoperability:**

  Different manufacturers produce IoT devices with varying communication protocols, making it difficult to integrate them into a unified system and share data seamlessly.

- **Scalability:**

  As the number of connected devices increases, managing and maintaining a large IoT network becomes complex, requiring robust infrastructure to handle data volume and processing power.

- **Security:**

IoT devices often have inherent vulnerabilities that can be exploited by hackers, necessitating strong security measures like encryption, authentication, and regular software updates.

- **Data Management:**

  The sheer volume of data generated by IoT devices presents challenges in storage, analysis, and extracting valuable insights.

- **Connectivity:**

  Maintaining reliable network connections, especially for devices situated in remote areas with poor signal strength, can be problematic.

- **Device Maintenance:**

  Ensuring ongoing functionality and timely updates for deployed IoT devices, particularly in hard-to-reach locations, can be difficult.

- **Bandwidth limitations:**

  As more devices connect to the network, the demand for bandwidth increases, potentially leading to network congestion.

- **Regulatory Compliance:**

  Adhering to various data privacy regulations across different regions can be complex for IoT deployments.

- **Lack of Skilled Professionals:**
  The growing IoT market demands expertise in device development, network management, and data analytics, which can be a challenge to find.

Addressing these challenges:

- **Standardized protocols:**

  Developing common communication protocols to enhance interoperability between devices from different manufacturers.

- **Robust security practices:**

  Implementing strong encryption, authentication mechanisms, and regular security updates for IoT devices.

- **Edge computing:**

  Processing data closer to the source on IoT devices to reduce network bandwidth usage and improve response time.

- **Data privacy frameworks:**

  Establishing clear guidelines for data collection, storage, and usage to protect user privacy.

- **Advanced analytics:**
  Utilizing data analytics techniques to extract meaningful insights from the vast amount of IoT data.

Opportunities in IoT

Opportunities in the Internet of Things (IoT) span across various sectors, including hardware development, software engineering, data analytics, network infrastructure, and cybersecurity, offering diverse career paths like IoT developers, platform engineers, data analysts, security specialists, and product managers, with potential to design and implement connected devices and systems for various industries, leveraging technologies like cloud computing, AI, and machine learning to optimize operations and create innovative solutions for smart homes, industrial automation, healthcare, and more; making it a rapidly growing field with significant demand for skilled professionals across different levels of expertise.

Key areas where IoT presents opportunities:

- **Hardware Development:**

  Designing and building physical IoT devices, including sensors, actuators, microcontrollers, and communication modules.

- **Software Development:**

  Creating embedded software for IoT devices, developing cloud-based platforms to manage and analyze data from connected devices, and building user interfaces for interacting with IoT systems.

- **Data Analytics:**

  Collecting, processing, and analyzing large volumes of data generated by IoT devices to extract valuable insights and drive decision-making.

- **Network Engineering:**

  Designing and managing network infrastructure for IoT devices, ensuring reliable connectivity and security.

- **Cybersecurity:**
  Developing security measures to protect IoT devices and networks from cyber threats.

Some potential career roles in IoT:

- **IoT Developer:** Develops software applications for IoT devices, including firmware and cloud-based services.
- **IoT Platform Engineer:** Designs and manages the platform that connects and manages various IoT devices.
- **Data Scientist/Analyst:** Analyzes data generated by IoT devices to identify patterns and generate actionable insights.
- **IoT Security Engineer:** Focuses on securing IoT systems and devices against cyber threats
- **IoT Product Manager:** Oversees the development and launch of new IoT products, including market analysis and strategy
- **Embedded Systems Engineer:** Develops software for microcontrollers within IoT devices

Industries utilizing IoT:

- **Manufacturing:** Predictive maintenance, asset tracking, production optimization
- **Healthcare:** Remote patient monitoring, wearable devices, smart healthcare systems
- **Smart Cities:** Traffic management, energy optimization, environmental monitoring
- **Retail:** Inventory management, customer analytics, personalized shopping experiences
- **Logistics:** Real-time tracking of shipments, delivery optimization

Key skills for an IoT career:

- Programming languages like C, C++, Java, Python
- Understanding of networking protocols (Wi-Fi, Bluetooth, LoRa)
- Data analysis and visualization skills
- Cloud computing knowledge (AWS, Azure, GCP)
- Cybersecurity concepts and practices

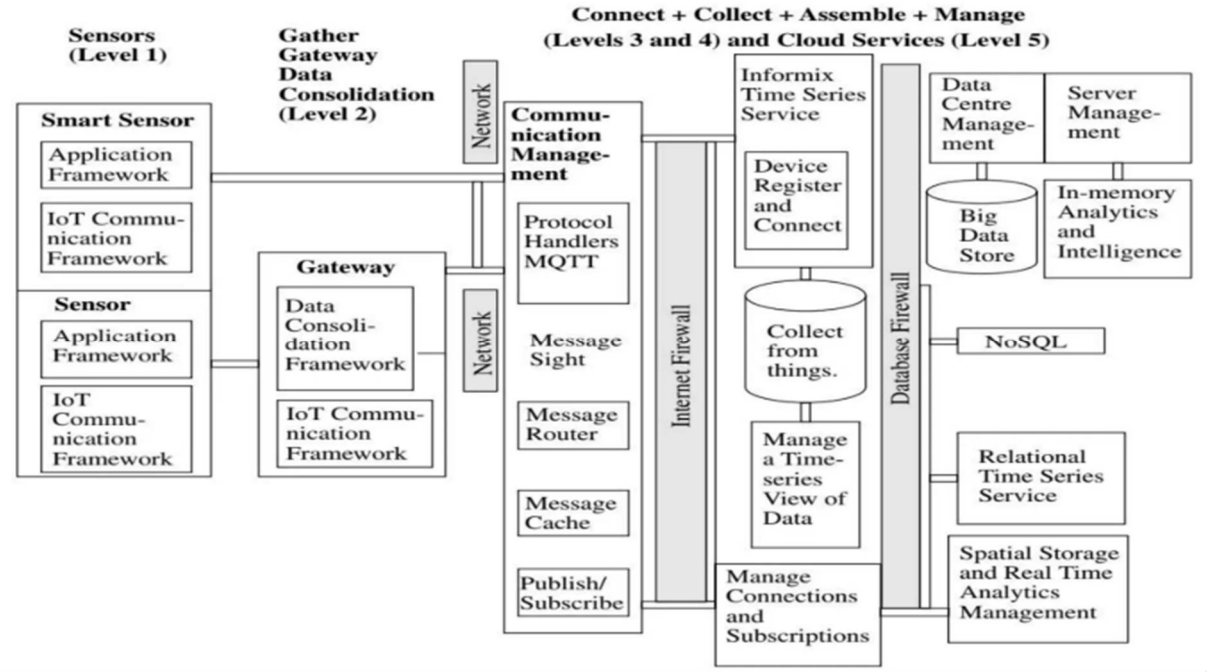
Unit – 2

IoT Architecture and Technologies

- Conceptual Framework
- IoT Architecture Overview
- Technology behind IoT
- Sources of the IoT
- M2M Communication
- IoT Examples

Conceptual Framework

A conceptual framework for the Internet of Things (IoT) outlines the key components and layers involved in connecting physical devices, collecting data from sensors, transmitting that data through a network, and ultimately utilizing it through applications, typically including a "perception layer" for data collection, a "connectivity layer" for network transmission, a "data processing layer" for analysis, and an "application layer" for user interaction; essentially providing a structured understanding of how an IoT system functions from device to application.



Key elements within an IoT conceptual framework:

- **Perception Layer (Device Layer):**

  This layer consists of physical devices like sensors, actuators, and RFID tags that collect raw data from the environment.

- **Connectivity Layer (Network Layer):**

  This layer manages the communication between devices, including protocols like Wi-Fi, Bluetooth, cellular networks, and gateways to connect to the broader internet.

- **Data Processing Layer (Application Layer):**

  This layer processes, analyzes, and aggregates the collected data from the devices, often utilizing cloud computing services for storage and computation.

- **Application Layer (User Interface Layer):**
  This layer presents the processed data to users through user interfaces like dashboards, mobile apps, or other applications that enable decision-making and control actions.

Important aspects of an IoT conceptual framework:

- **Data Security:**

  Mechanisms to protect data privacy and integrity throughout the IoT system, including encryption and authentication.

- **Device Management:**

  Techniques to manage and monitor connected devices, including updates, configuration, and troubleshooting.

- **Scalability:**

  The ability to add new devices and applications to the network without compromising performance.

- **Standards and Protocols:**
  Adherence to industry standards for interoperability between different devices and systems.

Example Applications of IoT Conceptual Framework:

- **Smart Homes:**

  Sensors monitoring temperature, lighting, and appliance usage, allowing automated adjustments based on user preferences.

- **Industrial Automation:**

  Sensors on machinery collecting data for predictive maintenance and optimized production processes.

- **Healthcare Monitoring:**
  Wearable devices tracking vital signs and providing real-time health insights.

IoT Architecture Overview

Internet of Things (IoT) is a system of interrelated, internet-connected objects which are able to collect and transfer data over a wireless network without human intervention.

For example, smart fitness bands or watches, driverless cars or drones, smart homes that can be unlocked through smartphones and smart cars, etc.

Architecture of IoT

There are different phases in the architecture of IoT but they can vary according to the situations but generally, there are these four phases in the architecture of IoT –

Networked Devices

These are the physical devices which include sensors, actuators, and transducers. These are the actual devices that collect and send the data for processing. They are capable of receiving real-time data and they can convert the physical quantities into electrical signals which can be sent through a network.

Data Aggregation

It is a very important stage as it includes converting the raw data collected by sensors into meaningful data which can be used to take actions. It also includes Data Acquisition Systems and Internet Gateways. It converts the Analog signals provided by sensors into digital signals.

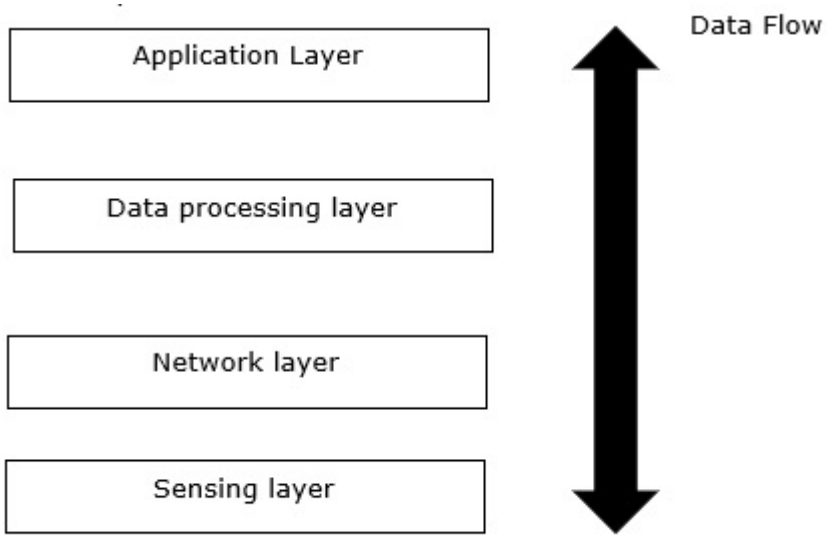
Final Analysis

This is a stage that includes edge IT analytics and the processing of data to make it more efficient and fully capable of execution. It also includes managing and locating all the devices correctly

Cloud Analysis

The final data is received here and analysed closely and precisely in data centres. They process and clean the data to make it free from any kind of errors and missing values. After this stage, data is ready to be sent back and executed to perform operations.

Now let us see the basic fundamental architecture of IoT which consists of four stages as shown in the diagram given below –



- **Sensing Layer** – The first stage of IoT includes sensors, devices, actuators etc. which collect data from the physical environment, processes it and then sends it over the network.
- **Network Layer** – The second stage of the IoT consists of Network Gateways and Data Acquisition Systems. DAS converts the analogue data (collected from Sensors) into Digital Data. It also performs malware detection and data management.

- **Data Processing Layer** – The third stage of IoT is the most important stage. Here, data is pre-processed on its variety and separated accordingly. After this, it is sent to Data Centres. Here Edge IT comes into use.
- **Application Layer** – The fourth stage of IoT consists of Cloud/Data Centres where data is managed and used by applications like agriculture, defence, health care etc.

Advantages

The advantages of IoT are as follows –

- **Cost Reduction** – IOT devices catch any problem very fast as compared to traditional troubleshooting. It not only saves time but also saves costs of large repairs.
- **Efficiency and Productivity** – An automated PDF conversion and creation tool will remove the hustle of PDF editing and archiving. Hence, increase in Efficiency and Productivity.
- **Business Opportunities** – IOT provides advanced analytics, smart utility grids which help Small Management Businesses to provide more valuable content and things to their customers.
- **Customer Experience** – Nowadays customer's experience is the most valuable thing in running a business. IoT has drastically increased the customer's experience. An example of customer experience is Home Automation. Since everything is connected, customers need not have to worry about appliances. One can turn off the appliance through mobile.
- **Mobility and Agility** – With the help of IoT, employees can do their work from any geographical location, anytime without any restrictions.

Disadvantages

The disadvantages of IoT are as follows –

- **Security** – The data is travelling all over the Internet. So maintaining its privacy is still a Big Challenge. End-to-end Encryption is a must in IoT.
- **Compatibility** – There is no International Standard for the monitoring of the equipment.
- **Complexity** – Most of the devices still contain some software bugs. Each device must be able to seamlessly interact with other devices in the network.
- **Safety** – Suppose a patient is left unattended by a doctor. And some notorious guy changes the prescription or Health monitoring devices malfunctioned. Then it can result in the death of the patient.
- **Policies** – Government authorities must take some steps to make policies and standards related to IoT to stop the Black marketing of IoT devices.

Technology behind IoT

The technology behind the Internet of Things (IoT) involves embedding sensors and processors into everyday objects, allowing them to collect data, connect to the internet, and communicate with other devices, essentially creating a network of "smart things" that can be monitored and controlled remotely; key components include sensors to gather data, actuators to respond to data, low-power computing chips, wireless communication protocols

like Wi-Fi, Bluetooth, and cellular networks, cloud platforms for data storage and analysis, and often, machine learning algorithms for intelligent decision-making.

Key aspects of IoT technology:

- **Sensors:**

  These are the primary data collection tools, detecting environmental changes like temperature, pressure, motion, light, etc.

- **Actuators:**

  Devices that respond to sensor data by taking actions like turning on a motor, opening a valve, or adjusting a setting.

- **Microcontrollers:**

  Small, low-power computers embedded within IoT devices to process sensor data and execute basic functions.

- **Connectivity protocols:**

  Wireless standards like Wi-Fi, Bluetooth, Zigbee, LoRaWAN, and cellular networks enable communication between devices and the cloud.

- **Cloud platforms:**

  Secure servers where data from IoT devices is stored, analyzed, and managed.

- **Data analytics:**
  Techniques like machine learning and AI are used to extract insights from the vast amount of IoT data.

How IoT works:

1. 1. **Data collection:**

   Sensors on an IoT device detect changes in the environment and collect data.

2. 2. **Data processing:**

   The microcontroller within the device pre-processes the raw data.

3. 3. **Data transmission:**

   The processed data is sent wirelessly to a gateway or directly to the cloud via a chosen network protocol.

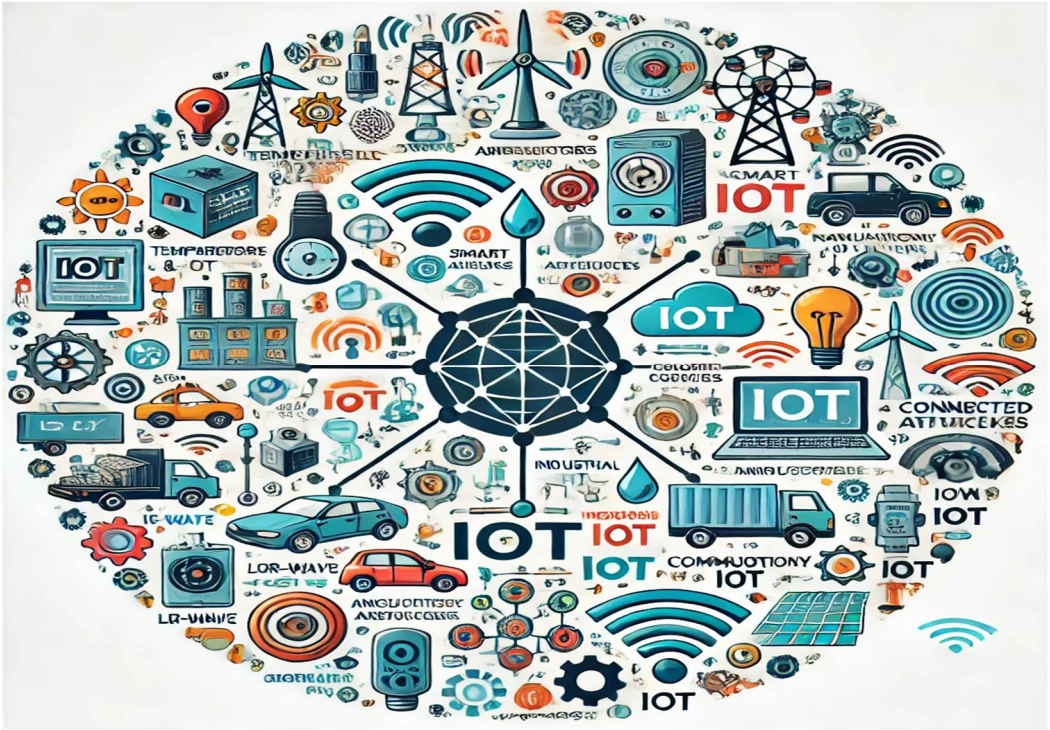
4. 4. **Data storage and analysis:**

   The cloud platform stores the data and uses analytics tools to extract meaningful insights.

5. 5. **Action execution:**
   Based on the analysis, the cloud may send commands to actuators on the IoT device to take appropriate actions.

Sources of the IoT

The Internet of Things (IoT) relies on various sources and technologies to enable connectivity, data exchange, and automation.



Wi-Fi

- Wi-Fi technology allows IoT devices to connect to local area networks (LANs) and the internet wirelessly,
- providing high-speed data transmission and connectivity within a certain range.
- **Example**: Smart home devices like cameras, and speakers connect to Wi-Fi networks for remote control and data sharing.

Bluetooth

- Bluetooth technology enables short-range wireless communication between IoT devices, s

- martphones, and accessories,facilitating seamless data transfer and device pairing.
- **Example:** Wearable fitness trackers use Bluetooth to sync data with mobile apps for activity tracking and analysis.

Zigbee and Z-Wave

- These low-power wireless protocols are commonly used in smart home automation for creating mesh networks,
- allowing devices to communicate efficiently with minimal energy consumption.
- **Example:** Smart lighting systems use Zigbee or Z-Wave to control and coordinate multiple light bulbs in a home network.

Cellular Networks 3G/4G/5G

- Cellular networks provide IoT devices with wide-area connectivity,
- allowing them to transmit data over long distances using cellular infrastructure and SIM cards.
- **Example:** GPS trackers for vehicles use cellular networks to send real-time location data to fleet management systems.

Satellite Communication

- In remote or rural areas with limited terrestrial connectivity, satellite internet servic**es enable IoT devices to access the internet and communicate globally.**
- **Example:** Environmental monitoring systems in remote regions use satellite communication to transmit weather data and alerts.

M2M Communication

WHAT IS M2M?

M2M, short for machine-to-machine, refers to the ability of networked devices to exchange information and perform actions without human intervention. It encompasses any technology that enables seamless communication between machines, allowing them to make autonomous decisions. M2M is facilitated by artificial intelligence (AI) and machine learning (ML) techniques, which enable systems to interpret data and make informed choices.

HOW DOES M2M WORK?

M2M technology taps into sensor data collected by devices and transmits it over a network. Unlike traditional remote monitoring tools, M2M systems commonly employ public networks such as cellular or Ethernet connections, which makes the technology more cost-effective.

Key components of an M2M system include sensors, radio frequency identification (RFID), a Wi-Fi or cellular communications link, and autonomic computing software that interprets data and triggers preprogrammed automated actions.

HOW DOES M2M COMPARE TO IOT?

While often used interchangeably, M2M and the Internet of Things (IoT) are not the same. M2M serves as a building block for IoT but can also function independently. It typically involves isolated, stand-alone networked equipment that communicates point-to-point over cellular or wired networks.

On the other hand, IoT takes M2M to the next level by integrating disparate systems into a connected ecosystem. It relies on IP-based networks to collect and transmit data from connected devices to gateways, the cloud, or middleware platforms. While M2M affects business operations, IoT impacts both business operations and end users.

WHERE IS M2M USED?

M2M has found applications in various industries, enabling significant advancements. Some of them include:

- **Manufacturing:** M2M helps in remotely managing and controlling data from equipment, leading to improved productivity and reduced maintenance costs.
- **Healthcare:** M2M devices enable real-time monitoring of patient vital stats, dispensing medicine when needed, tracking healthcare assets, and enhancing patient care.
- **Utilities:** M2M is utilized in harvesting energy, billing customers through smart meters, and monitoring factors such as pressure, temperature, and equipment status.
- **Telemedicine:** M2M plays a vital role in remote patient monitoring, allowing healthcare providers to deliver timely care and medications.
- **Transportation:** M2M contributes to logistics and fleet management by enabling asset tracking, optimizing routes, and improving safety and security measures.
- **Smart grid:** M2M facilitates the collection and analysis of data from smart meters, helping to manage energy distribution better and minimize waste.

HOW IS M2M USED?

M2M technology is employed in a wide range of use cases, including but not limited to the following:

- **Remote monitoring:** M2M enables remote monitoring of equipment, such as vending machines, that can communicate with distributors to request refills when running low on certain products.
- **Asset tracking:** M2M plays a crucial role in warehouse management systems and supply chain management, allowing the tracking and monitoring of assets in real time.
- **Telecommunications:** M2M is utilized in monitoring network performance, measuring signal quality, detecting faults or outages, and facilitating quicker response times.
- **Home automation:** M2M is integrated into smart home systems, allowing appliances and devices to be controlled remotely and communicate in real time.

HOW DOES M2M HELP?

One example is the use of M2M in predictive maintenance. By continuously monitoring equipment and analyzing data in real-time, maintenance can be scheduled proactively based on actual usage and potential faults. This approach eliminates unnecessary maintenance and reduces downtime, resulting in cost savings, improved operational efficiency, and enhanced customer satisfaction.

WHAT ARE THE TYPES OF M2M?

M2M can be categorized into various types based on the communication and applications involved. Some common types of M2M systems are:

- **Cellular-based M2M:** Utilizes cellular networks such as 4G or 5G for communication between devices.
- **Satellite-based M2M:** Relies on satellite networks for global coverage and remote locations where terrestrial networks are unavailable.
- **RFID-based M2M:** Uses radio frequency identification technology for tracking and monitoring assets.
- **Industrial M2M:** Specifically tailored for industrial applications, typically involving large-scale systems and equipment.

WHAT ARE M2M REQUIREMENTS?

To ensure the proper functioning of M2M systems, several demands need to be met. According to the European Telecommunications Standards Institute (ETSI), these requirements include:

- **Scalability:** The ability to handle an increasing number of connected devices while maintaining system efficiency.
- **Anonymity:** The system's capability to protect the identity of devices when required while adhering to regulatory requirements.
- **Logging:** Support for recording important events and system logs, which can be accessible upon request.
- **Communication principles:** Enabling communication between M2M applications and devices using techniques such as SMS and IP, including peer-to-peer communication.
- **Delivery methods:** Supporting various communication modes, such as unicast, anycast, multicast, or broadcast, while minimizing network load.
- **Message transmission scheduling:** Controlling network access and messaging schedules in line with the tolerance for scheduling delays.
- **Message communication path selection:** Optimizing message paths based on factors like transmission failures, delays, and network costs.

WHAT ARE M2M SECURITY STANDARDS?

M2M systems face security challenges, including unauthorized access, data breaches, and device hacking. To address these concerns, several security measures and standards have been developed. Typical M2M security measures include tamper-resistant devices, ensuring communication security through encryption, securing back-end servers, and managing device identity and data confidentiality. Standards such as OMA DM (Open Mobile Alliance Device Management), OMA LightweightM2M, MQTT, and TR-069 provide protocols and guidelines for secure M2M communication and data exchange.

WHAT ARE THE BENEFITS OF M2M?

M2M technology offers numerous benefits across various industries, including:

- **Reduced costs:** By minimizing equipment maintenance and downtime, M2M helps lower operational expenses.
- **Boosted revenue:** M2M can identify new business opportunities by providing insights into product servicing needs and customer preferences.
- **Improved customer service:** Proactive monitoring and servicing of equipment ensure optimal performance and minimize disruptions, leading to enhanced customer satisfaction.

WHAT ARE THE CHALLENGES OF M2M?

Despite its many benefits, M2M still faces challenges that need to be addressed for widespread adoption. Some key challenges include:

- **Security:** Protecting M2M systems from unauthorized access, data breaches, and device hacking.
- **Standardization:** The lack of standardized device platforms in M2M leading to fragmentation and interoperability issues.
- **Scalability:** Ensuring M2M systems can handle large-scale deployments and an increasing number of connected devices.
- **Device management:** Effectively managing and updating M2M devices remotely, especially when they are deployed in inaccessible locations.
- **Privacy:** Addressing concerns related to data privacy and ensuring compliance with relevant regulations.

WHAT IS THE FUTURE OF M2M?

The future of M2M looks promising, with its impact expected to grow rapidly. As technologies like AI, ML, and 5G continue to evolve, M2M systems will become even more integrated, intelligent, and efficient. With advancements in security measures and standards, M2M will be able to address current challenges and drive innovation in various industries.

The convergence of M2M with other emerging technologies will further enhance the capabilities and applications of connected devices, leading us towards a more connected and automated future.

IoT Example

Examples of IoT (Internet of Things)

The **Internet of Things (IoT)** is widely used across various industries and everyday life. Below are some key examples of IoT applications:

1. Smart Home Automation

- **Smart Thermostats** (e.g., Nest, Ecobee) adjust temperature based on user preferences.
- **Smart Lights** (e.g., Philips Hue) can be controlled via mobile apps or voice assistants.
- **Security Cameras & Smart Doorbells** (e.g., Ring, Arlo) provide remote monitoring and motion detection.

2. Industrial IoT (IIoT)

- **Predictive Maintenance**: Sensors monitor machines and alert when maintenance is required.
- **Smart Factories**: Automated robots and real-time tracking optimize manufacturing.
- **Supply Chain Management**: IoT-based GPS tracking enhances logistics and inventory management.

3. Healthcare & Wearables

- **Smart Wearables** (e.g., Fitbit, Apple Watch) monitor heart rate, oxygen levels, and sleep patterns.
- **Remote Patient Monitoring**: IoT-enabled devices track blood pressure, glucose levels, and ECG data.
- **Smart Pills**: Ingestible sensors track medication adherence inside the body.

4. Smart Cities & Infrastructure

- **Smart Traffic Lights**: AI-powered traffic control reduces congestion.
- **Smart Parking Systems**: Sensors detect available parking spaces in real time.
- **Air Quality Monitoring**: IoT sensors measure pollution levels in cities.

5. Agriculture & Smart Farming

- **Automated Irrigation Systems**: IoT sensors adjust water supply based on soil moisture.
- **Livestock Monitoring**: GPS collars track animal health and location.
- **Weather Sensors**: Provide real-time climate data for precision farming.

6. Connected Vehicles & Transportation

- **Self-Driving Cars** (e.g., Tesla Autopilot) use IoT sensors and AI for navigation.
- **Fleet Management**: GPS tracking optimizes delivery routes for logistics companies.
- **Smart Public Transport**: IoT-enabled buses provide real-time tracking for passengers.

Unit – 3

Hardware for IoT

- Sensors
- Digital Sensors
- Actuators
- Radio Frequency Identification (RFID) Technology
- Wireless sensor networks
- Overview of IoT supported Hardware platforms:
    - Arduino
    - Netduino

Sensors

What are sensors in IoT?

The Internet of Things (IoT) is a technology innovation that allows all our devices to be connected and communicate with each other. For example, if your television, fridge, and air conditioner are using Wi-Fi, you can control these devices using your smartphone or voice assistant. From changing the temperature to adjusting the volume of your television or any other activity, IoT creates new possibilities for all our electronic devices.
Sensors are electronic devices that continuously gather real-time information about their surroundings. This information can be anything from temperature and pressure to motion and light. Then, sensors convert this physical data into an electrical signal that devices can understand.

How do sensors work?

Imagine a sensor as a tiny translator constantly monitoring a specific aspect of its environment. Here's a simplified breakdown of the process:

1. **Physical Measurement:** The sensor detects a physical quantity like temperature, pressure, light, or motion.
2. **Signal Conversion:** This physical phenomenon is then converted into an electrical signal the device can understand.
3. **Data Processing:** The processed electrical signal becomes valuable data, often transmitted wirelessly.
4. **Action or Analysis:** This data is then used to trigger actions (like adjusting a thermostat) or analyzed for insights (like tracking fitness data).

Thus, sensors unlock new possibilities for all our devices, including its use for business or personal use. Some of the ways sensors are being used in key industries include:

- **Healthcare:** Wearable sensors in smartwatches and fitness trackers monitor heart rate, blood pressure, and activity levels, providing valuable data for health monitoring and disease prevention.
- **Automotive:** Cars equipped with temperature sensors can detect engine overheating and alert drivers, preventing potential breakdowns. Additionally, pressure sensors in tires ensure optimal performance and fuel efficiency.
- **Home Automation:** Smart thermostats utilize temperature sensors to adjust room temperature based on your preferences automatically, saving energy and enhancing comfort.

These are just a few examples, but the possibilities are truly endless! Sensors are continuously being integrated into new devices, paving the way for a more connected and intelligent world.

Characteristics of a sensor

Do you wonder what qualities make sensors effective tools for gathering data? Here are some key characteristics that define a good sensor:

- **Sensitivity:** This refers to the sensor's ability to detect even the slightest changes in the measured quantity. A highly sensitive sensor can pick up on subtle variations, leading to more accurate data.
- **Resolution:** This describes the smallest detectable change in the measured quantity. Imagine a ruler; a higher resolution sensor would be like having markings for millimeters instead of centimeters, allowing for more precise measurements.
- **Accuracy:** This reflects how closely the sensor's output reflects the actual physical quantity being measured. A sensor with high accuracy provides data that closely represents the real world.
- **Linearity:** Ideally, the output of a sensor should have a linear relationship with the measured quantity. This means a consistent change in output for a consistent change in the physical property.
- **Range:** This defines the minimum and maximum values the sensor can effectively measure. Choosing a sensor with an appropriate range ensures it can capture the data relevant to your application.
- **Selectivity:** A good sensor should be selective to the specific quantity it's designed to measure. Minimal interference from other environmental factors leads to cleaner and more reliable data.
- **Response Time:** This refers to the time it takes for a sensor to respond to a change in the measured quantity. Sensors with faster response times are crucial for applications requiring real-time data processing.

Classification of sensors

The world of sensors is vast and diverse, catering to a multitude of applications. To bring some order to this variety, sensors can be broadly classified based on two main aspects:

1. By Operating Principle:

This classification focuses on the physical phenomenon the sensor uses to detect and convert the measured quantity into an electrical signal. Here are some common types:

- **Piezoelectric Sensors:** These sensors convert pressure or mechanical stress into an electrical voltage. They are widely used in pressure sensors, microphones, and accelerometers (used in airbags and motion tracking).
- **Thermocouple Sensors:** These sensors rely on the principle that the junction of two dissimilar metals generates a voltage proportional to the temperature difference between the junction and a reference point. They are commonly used for industrial temperature measurement.
- **Photoelectric Sensors:** These sensors detect light or changes in light intensity. They are used in applications like automatic doors, security systems, and object detection.
- **Electrochemical Sensors:** These sensors utilize chemical reactions to generate an electrical signal. They are used in gas sensors, smoke detectors, and pH meters.

2. By Application:

This classification categorizes sensors based on the specific function they perform in an IoT application. Some examples include:

- **Temperature Sensors:** They measure temperature and are ubiquitous in various devices, from thermostats to smartwatches.
- **Pressure Sensors:** They detect changes in pressure and are used in applications like tire pressure monitoring systems and industrial process control.
- **Proximity Sensors:** They discover the presence or absence of nearby objects without physical contact. You'll find them in automatic faucets, self-driving cars, and even robotic vacuum cleaners.
- **Motion Sensors:** These detect movement or changes in position. They are used in security systems, activity trackers, and video game controllers.

Role of sensors in the architecture of IoT

Sensors play a crucial role in the overall architecture of an IoT system, acting as the foundation for data collection and analysis. Here's a breakdown of their key functions:

- **Data acquisition:** Sensors are the primary source of data for any IoT system. They continuously gather real-time information about the surrounding environment, transforming physical quantities into electrical signals.
- **Pre-processing:** In some cases, sensors perform basic pre-processing on the collected data. This might involve filtering out noise or converting the data into a format suitable for further processing or transmission.
- **Communication:** Modern sensors often have built-in communication capabilities, allowing them to transmit data wirelessly to other devices within the IoT network. This data can then be sent to gateways or cloud platforms for further processing and analysis.
- **Enabling automation:** Sensor data can be used to automate various tasks and processes within an IoT system. For instance, smart lights can be programmed to turn on when they detect you entering the house and off when you exit.

- **Real-time Monitoring:** Sensors enable real-time monitoring of various parameters in an IoT system. This allows for immediate responses to changes in the environment, leading to improved efficiency and control.
- . **Data-driven Insights:** The vast amount of data collected by sensors can be analyzed to gain valuable insights. This data can be used for predictive maintenance, optimizing processes, and even identifying potential problems before they occur

Digital Sensors

A **digital sensor** is an essential hardware component in the **Internet of Things (IoT)** ecosystem. It is responsible for **detecting physical parameters**, converting them into **digital signals**, and transmitting the data for processing, analysis, and automation. These sensors enable real-time monitoring and control in various IoT applications such as **smart homes, healthcare, industrial automation, agriculture, and environmental monitoring**.

Key Characteristics of Digital Sensors for IoT

1. **Digital Output** – Unlike analog sensors, digital sensors provide data in binary format (0s and 1s), ensuring **high accuracy** and ease of integration.
2. **Low Power Consumption** – Many IoT applications require energy-efficient sensors for battery-powered devices.
3. **Wireless Connectivity** – Digital sensors can transmit data using **Wi-Fi, Bluetooth, LoRa, Zigbee, NB-IoT**, or other communication protocols.
4. **Compact & Reliable** – Modern digital sensors are designed to be small, durable, and reliable for long-term operations.
5. **Edge Processing** – Some digital sensors feature onboard microcontrollers or AI capabilities to process data before transmission, reducing **latency** and **network load**.

Types of Digital Sensors in IoT

1. **Environmental Sensors** – Measure **temperature, humidity, air quality, and light** (e.g., **DHT22, BME680, BH1750**).
2. **Motion & Position Sensors** – Include **accelerometers, gyroscopes, and proximity sensors** (e.g., **MPU6050, VL53L0X**).
3. **Industrial Sensors** – Used for **vibration, gas detection, and predictive maintenance** (e.g., **MQ135, MEMS vibration sensors**).
4. **Healthcare Sensors** – Monitor **heart rate, oxygen levels, and ECG** (e.g., **MAX30102, AD8232**).
5. **Smart Agriculture Sensors** – Detect **soil moisture, weather conditions, and CO_2 levels** (e.g., **Capacitive Soil Moisture Sensor v1.2**).

Common Communication Interfaces for Digital Sensors

- **I²C (Inter-Integrated Circuit)** – Used for sensors like **BMP280** (barometric pressure).
- **SPI (Serial Peripheral Interface)** – Suitable for high-speed data transfer.
- **UART (Universal Asynchronous Receiver-Transmitter)** – Common in GPS and gas sensors.
- **Wireless Protocols** – **LoRa, Zigbee, Bluetooth, Wi-Fi** enable remote monitoring.

Integration with IoT Platforms

- **Microcontrollers** – ESP8266, ESP32, and Arduino support digital sensors.
- **Single-Board Computers** – Raspberry Pi enables advanced IoT applications.
- **Cloud Platforms** – AWS IoT, Google Cloud IoT, and ThingsBoard process and visualize sensor data.

Actuators

Actuators are physical devices in the Internet of Things (IoT) that convert energy into motion. They are used to move objects, open doors, close windows, and more. Actuators are powered by various sources, including batteries, electricity, and manual energy.

How do actuators work?

- Actuators receive an electrical signal
- They combine the signal with an energy source
- The actuator converts the energy into motion

Types of actuators

- **Hydraulic actuators**

  Convert hydraulic energy into mechanical energy. They can produce linear or rotary output.

- **Pneumatic actuators**

  Convert energy into compressed gas, then into mechanical energy. They are used to open and close valves.

- **Electric actuators**

  Convert electric signals into kinetic energy. They can produce linear, rotary, or single motion.

- **Thermal actuators**

  Use thermal-sensitive material to produce linear motion when the temperature changes. They do not require additional power to create motion.

Applications of actuators

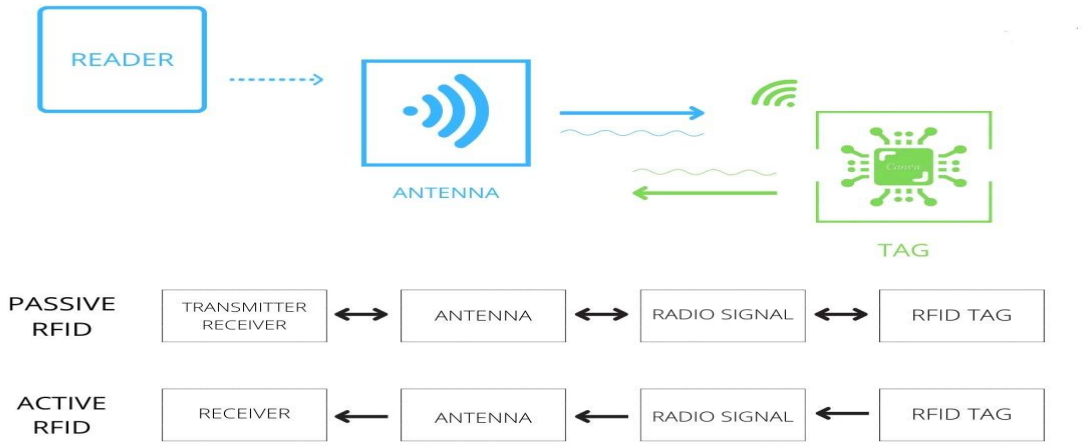
- **Robotics**: Actuators are used in robots to control their movement.
- **Industrial automation**: Actuators are used to automate industrial valves.
- **Smart home systems**: Actuators are used to control devices like blinds, doors, and windows.
- **Irrigation systems**: Actuators are used to control sprinklers and valves to optimize water usage.

Radio Frequency Identification (RFID) Technology

RFID (Radio Frequency Identification) is a type of wireless communication that uses electromagnetic or electrostatic coupling in the radio frequency spectrum to uniquely identify an object, animal, or human.
It is a technology used for automatically identifying and recording data about an object via a tiny, uniquely identifiable microchip tag connected to the object. A built-in antenna on the RFID tag interacts with a scanning device that can remotely read the tag's data.
The scanning device scans the tag when it comes in range. After that, the data is sent from the scanning equipment to an application program. With the help of the application, the user will store and send it wherever he desires.

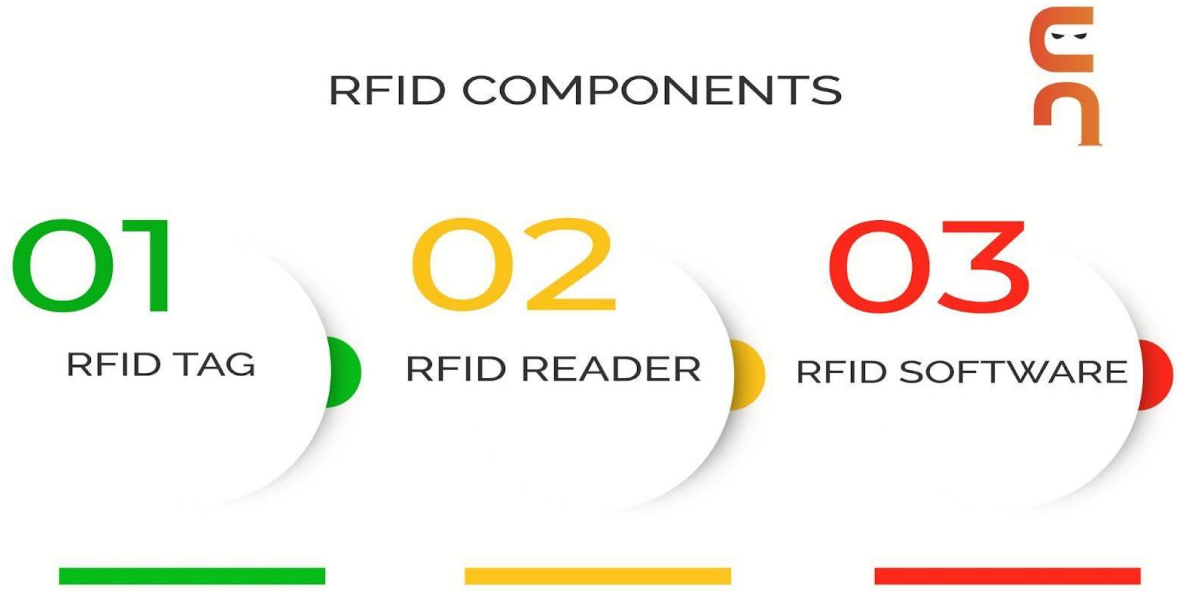
Working of RFID

RFID, or radio frequency identification, is a technique for automatically identifying and capturing data about an object that has been stored in a small microchip tag attached to the object. An antenna built into the RFID tag communicates with a scanning device that reads the data remotely.
This data is then transferred from the scanning device to the data-housing enterprise application software. Each RFID tag has a unique identification number.
RFID can be used to track and control asset and personnel movement. RFID tags can be found on the back of library books and even in the new biometric passports. It simplifies the management of assets contained in boxes or pallets.



Components of RFID

Radio Frequency Identification technology consists of three main components:



1. **The RFID tag**: The RFID tag comprises an integrated circuit, a substrate, and an antenna. If the tag has an active power source and thus can support a sensor, it is called an active RFID tag. If the tag doesn't have an active power source, it is called a passive RFID tag.

2. **The RFID reader:** It is a device that reads RFID tags and gathers data about the connected object. It can be both wired and wireless. It can use many technologies to communicate with the software, including USBs and Bluetooth connections.

3. **The RFID software**: The software monitors and tracks the object connected to the RFID tags. It can be called data exchange and management software.

Applying RFID to IoT Devices

RFID tags are helpful in cameras, GPS, and other smart sensors when used in IoT. They can aid in the identification and location of objects. It is a low-cost way to make household objects "smart," similar to the popular Google Nest products. RFID tags are being used by some healthcare systems to track patients and their medical records.
RFID is used in transportation systems to read passenger data, control traffic, and update transportation systems.

Role of RFID in IoT

Radio Frequency Identification technology is one of the three main components of IoT, along with the Savant system and the Internet. Thus, it has had wide-ranging implications for IoT development as a whole.
RFID technology has a wide range of applications in the Internet of Things. RFID tags are generally used to enable ordinary things to interact with one another and with the central hub and report their status. These features serve as the building blocks for an IoT system. To put it another way, RFID technology allows IoT to connect items to a network and will enable them to produce and deliver data.

Applications of RFID in IoT

- RFID has seen applications since the 1940s when they were first introduced. Its use rapidly increased to mainstream levels during the 70s. With the rise of IoT, it has threatened barcodes and NFCs as the most efficient technology to identify and track objects, livestock and humans uniquely.

- RFID tags are useful in cameras, GPS, and other smart sensors when utilised in the IoT. They can help with identifying and locating items. It's a low-cost approach to make household items seem "smart", as many companies are now entering the smart home market.

- Healthcare institutions also use RFID tags to track patients and their medical information. They are being used in transportation systems to read passenger data, regulate traffic, and update transportation systems.

Wireless sensor networks

What Is Wireless Sensor Network?

WSNs stands for Wireless Sensor Networks can be defined as a self-configured and infrastructure-less wireless network to observe physical or environmental conditions, like temperature, pressure, motion, sound, vibration, or pollutants, and to directly pass their data or information through the network to a sink which is also called the main location where the information is often observed and analyzed.

A base station or sink seems like an interface between the users and the network. It can convert back some required information from the network by injecting some queries and gathering results from the sink. Typically a wireless sensor network contains many thousands of sensor nodes.

The sensory nodes can communicate with each other by using radio signals. The wireless sensor nodes are equipped with sensing and radio transceivers, computing devices, and power components.

A sensor node in a wireless sensor network is inherently resource-constrained, also it has limited processing speed, storage capacity, and communication bandwidth. After the sensor nodes are installed, they're responsible for self-organizing an appropriate network infrastructure often with multi-hop communication with them.

Then the onboard sensors begin to collect information of their interest. And then the specifically designed devices of wireless sensor networks reply to those queries sent from a "control site" to perform specific instructions or provide sensing samples.

The working mode of the sensor nodes could also be either continuous or event-driven. GPS or Global Positioning System and LPA or local positioning algorithms can be used to obtain location and positioning information.

Wireless sensor devices are often equipped with actuators to "act" upon certain conditions. These networks are sometimes or normally called Wireless Sensor Network and Actuator Network.

Types of Wireless Sensor Networks

There are five types of Wireless Sensor Networks depending on the environment. Different Types of WSNs are:

1. Terrestrial Wireless Sensor Networks: Terrestrial WSNs are used for communicating base stations efficiently, and comprise thousands of wireless sensor nodes deployed either in an unstructured (ad hoc) or structured (Pre-planned) manner.

In an unstructured mode (ad hoc), the sensor nodes are randomly distributed within the target area that's dropped from a set plane.

In WSNs, the battery power is limited, however, the battery is provided with solar cells as a secondary power source. The conservation of energy of the WSNs gets by using low duty cycle operations, optimal routing, minimizing delays, and so on.

2. Underground Wireless Sensor Networks: In terms of deployment, maintenance, equipment cost considerations, and careful planning, underground wireless sensor networks are more expensive than terrestrial WSNs.

The Underground Wireless sensor networks UWSNs comprises several sensory nodes that are hidden in the ground to observe underground conditions.

Additional sink nodes are located above the bottom to transfer information from the sensor nodes to the base station, These underground WSNs deployed into the ground are difficult to recharge.

The sensor battery nodes equipped with limited battery power are also difficult to recharge. Additionally, the underground environment makes wireless communication a challenge because of the high attenuation and signal loss level.

3. Underwater Wireless Sensor Networks: About more than 70% of the earth's planet is occupied with water. These networks contain several sensor nodes and vehicles deployed underwater. Autonomous underwater devices and vehicles are used to collect data from these sensor nodes.

A challenge of underwater communication may be a long propagation delay, and bandwidth and sensor failures. Underwater, WSNs are equipped with a limited battery that can't be recharged or replaced.

The difficulty of energy conservation for underwater WSNs involves the development of underwater communication and networking techniques.

4. Multimedia Wireless Sensor Networks: Multimedia wireless sensor networks are proposed to enable tracking and monitoring of events in the sort of multimedia, like video, imaging, and audio.

These networks contain low-cost sensor nodes equipped with cameras and microphones. These sensory nodes of Multimedia WSNs are interconnected together over a wireless connection for data retrieval, data compression, and correlation.

The challenges with the Multimedia WSNs include high bandwidth requirements, high energy consumption, processing, and compressing techniques. Additionally, multimedia contents need high bandwidth for the content to be delivered properly and easily.

5. Mobile Wireless Sensor Networks MWSNs: Mobile WSNs networks comprise a group of sensor nodes that can be moved on their own and can be interacted with the physical environment. The mobile nodes can also compute sense and communicate respectively.

Mobile wireless sensor networks are way more versatile than static sensor networks. The benefits of Mobile WSNs over Static WSNs include better and improved coverage, superior channel capacity, better energy efficiency, and so on.

Classification of Wireless Sensor Networks

1. Static and Mobile WSN: All the sensor nodes are connected without movement and these are static networks in many applications. Some applications especially in biological systems-mobile sensor nodes are needed. These are called mobile networks. An example of a mobile network is animal monitoring.

2. Deterministic and Nondeterministic WSN: In deterministic wireless sensor networks, the sensor node position is calculated and fixed.

The deployment of sensor nodes is possible in a limited number of applications. The position of sensor nodes determination isn't possible because of several factors like harsh environments or hostile operating conditions. Such kinds of networks are non-deterministic and need a complex system.

3. Single Base Station and Multi Base Station WSN: In single base station WSNs, only one base station is used that is found close to the sensor node region.

All the nodes communicate with this base station, in the case of a multi-base station WSNs, more than one base station is used and a sensor node can transfer data to the closest base station.

4. Static Base Station and Mobile Base Station WSN: It is similar to sensor nodes, even base stations of the WSN are often either static or mobile. A static base station contains a fixed position usually close to the sensing region.

A mobile base station WSN moves around the sensing region because a load of sensor nodes is balanced.

5. Single-hop and Multi-hop WSN: In single-hop WSNs, the sensor nodes are directly connected to the base station. And in the case of multi-hop WSNs, peer nodes and cluster heads are used to relay the information to reduce energy consumption.

6. Self Reconfigurable and Non- Self Configurable WSN: In non-Self Configurable WSNs, the sensor networks cannot organize themselves in a network and consider a control unit to gather data.

In many WSNs, the sensor nodes can be able to organize and maintain the connection and work collaboratively with other sensor nodes to accomplish the task.

7.Homogeneous and Heterogeneous WSN: In the case of homogeneous WSNs, all the sensor nodes have the same energy consumption, storage capabilities, and computational power.

And in the case of heterogeneous WSNs, some sensor nodes have higher computational power and energy requirements than others and also the processing and communication tasks are divided accordingly.

Structure of Wireless Sensor Network

The structure of WSNs includes different types of topologies for radio communications networks.

1. Star Network: A star network is also called a single point-to-multipoint is a communications topology where one base station can send and receive a message to a variety of remote nodes. The remote nodes aren't permitted to send messages.

The benefit of these kinds of networks for wireless sensor networks includes simplicity, ability to keep the remote node's power consumption to a minimum. It allows low-power communications between the remote node and the base station.

The disadvantage of such a network is that the base station must be within the radio transmission range of all the individual nodes and isn't as robust as other networks because of its dependency on a single node to manage the whole network.

2. Mesh Network: A mesh network allows transmitting data from one node to another in the network that's within its radio transmission range.

This enables what is called multi-hop communications, i.e. if a node wants to send a message to a different node that's out of radio communications range, it can use an intermediate node to forward the message to the particular node.

This topology has the power of redundancy and scalability. When an individual node fails to work, a remote node still can communicate to the other node in its range, which successively, can forward the message to the specified location.

Additionally, the range of the network isn't necessarily limited by the range in between single nodes, it can simply be extended by adding more nodes to the system.

3. Hybrid Star: A hybrid Star is a combination between the star network and a mesh network that provides a strong and versatile communications network while maintaining the ability to keep the wireless sensor node's power consumption to a minimum.

In network topology, the sensor nodes with the lowest power aren't enabled with the ability to forward messages. This permits for minimal power consumption to be maintained.

Similarly, the various other nodes on the network are having multi-hop capability, allowing them to forward messages from the low power nodes to another on the network.

The nodes whose having the multi-hop capability are of a higher power, and if possible, are often plugged into the electrical mains line.

Applications of wireless sensor network

Wireless sensor networks have been used widely over the world. The applications of wireless sensor networks are:

- **Military applications:** The military domain isn't only the primary field of human activity that is used by WSNs but it's also considered to have motivated the initiation of sensor network research.

Tracking and environment monitoring surveillance applications use these kinds of networks. The sensor nodes from sensor networks are dropped to the sector of interest and are remotely controlled by a user.

Security detections and enemy tracking are also performed by using these networks.

- **Health applications:** These networks are generally used by doctors to track and monitor patients.
- **Transport systems:** Most frequently used wireless sensor networks are in the transport systems like dynamic routing management, monitoring of traffic, and monitoring of parking lots, etc., use these networks.
- **Environmental trackings:** Wireless Sensor Networks have been used widely in the field of environment changes and their tracking.

Forest detection, animal tracking, weather prediction, flood detection, forecasting, and also commercial applications like seismic activity prediction and also monitoring are using these networks like Air pollution monitoring, water quality monitoring, etc.

- **Threat detection**: The Wide Area Tracking System (WATS) is a device and a prototype network for detecting a ground-based nuclear device such as a nuclear bomb, and many other WSNs are also used for threat detection.
- Industrial process monitoring, rapid emergency response, automated building climate control, area monitoring, civil structural health monitoring, ecosystem, and habitat monitoring, etc., use these networks to monitor things.

Characteristics of Wireless Sensor Network

Some basic characteristics of Wireless Sensor Networks are as follows:

- Power consumption constraints for nodes using energy harvesting or mainly batteries are used.
- Examples of suppliers are ReVibe Energy and Perpetuum
- Having the ability to deal with node failures (resilience)
- Having some mobility of nodes (for highly mobile nodes see Mobile Wireless Sensor Networks)
- Scalability to the large scale of deployment
- Ability to resist harsh environmental conditions

- Heterogeneity of nodes
- Homogeneity of nodes
- Easy to use
- Cross-layer optimization

Issues in Wireless Sensor Networks

Various issues are occurring in wireless sensor networks WSNs such as design issues, topology issues, and other issues.

The complications in design in different types of wireless sensor networks include:

- Low latency
- Transmission Media
- Fault
- Coverage Problems
- Scalability

The complications in the topology of wireless sensor networks include the following.

- Sensor Holes
- Coverage Topology
- Geographic Routing

The big issues of a wireless sensor network WSNs include the following. These issues mainly affect the design and performance of wireless sensor networks.

- Operating System & Hardware for WSN
- Schemes for Medium Access
- Deployment
- Middleware
- Characteristics of Wireless Radio Communication
- Architecture
- Calibration
- Database Centric and Querying
- Network Layer
- Localization
- Sensor Networks Programming Models
- Synchronization
- Transport Layer
- Data Dissemination & Data Aggregation

The Advantages and Disadvantages of wireless sensor networks:

The advantages of wireless sensor networks WSNs are as follows:

- It is suitable for non-reachable places like over the sea, mountains, rural areas, or deep forests.

- It avoids lots of wiring.
- It might accommodate new devices at any time.
- It can also be accessed by using a centralized monitor.
- Flexible if there's a random situation when the additional workstation is required.
- Implementation pricing is affordable.
- It's flexible to undergo physical partitions.

The disadvantages of Wireless sensor networks are as follows:

- Less secure because hackers can enter the access point and obtain all the data.
- Lower speed as compared to a wired network.
- It's easy for hackers to hack it we couldn't control the propagation of waves.
- It is even more complicated compared to a wired network.
- Easily troubled by surroundings (walls, microwave, large distances because of signal attenuation, etc).
- Comparatively low speed of communication.
- Gets distracted by various elements like Blue-tooth.
- Still Costly (most importantly).

Overview of IoT supported Hardware platforms:

Arduino

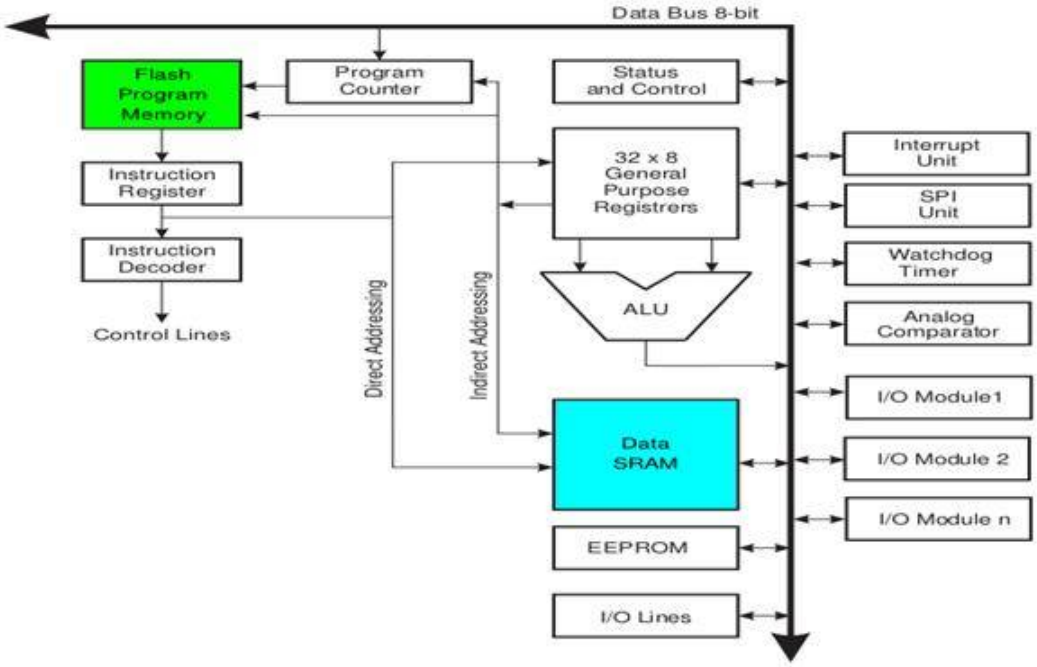
Defining Arduino

An Arduino is actually a microcontroller based kit which can be either used directly by purchasing from the vendor or can be made at home using the components, owing to its open source hardware feature. It is basically used in communications and in controlling or operating many devices. It was founded by Massimo Banzi and David Cuartielles in 2005.

Arduino Architecture:

Arduino's processor basically uses the Harvard architecture where the program code and program data have separate memory. It consists of two memories- Program memory and the data memory.The code is stored in the flash program memory, whereas the data is stored in

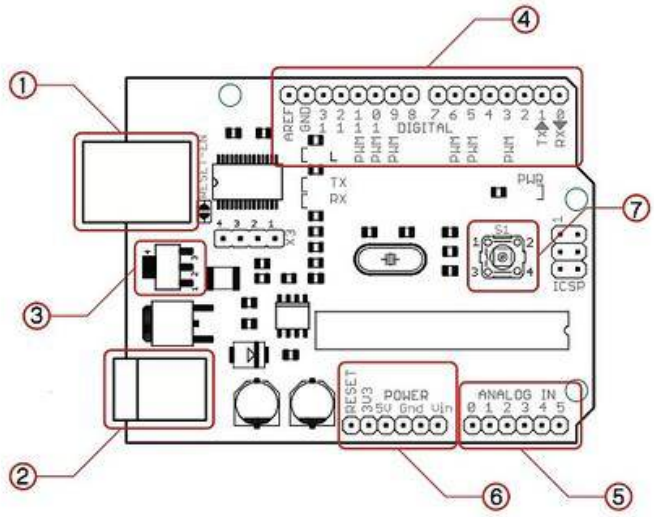
the data memory. The Atmega328 has 32 KB of flash memory for storing code (of which 0.5 KB is used for the bootloader), 2 KB of SRAM and 1 KB of EEPROM and operates with a clock speed of 16MHz.



Arduino Architecture

Arduino Pin Diagram

A typical example of Arduino board is Arduino Uno. It consists of ATmega328- a 28 pin microcontroller.



The most important parts on the Arduino board high lighted in red:

1: USB connector
2: Power connector
3: Automatic power switch
4: Digital pins
5: Analog pins
6: Power pins
7: Reset switch

Arduino Pin Diagram

Arduino Uno consists of 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button

Power Jack: Arduino can be power either from the pc through a USB or through external source like adaptor or a battery. It can operate on a external supply of 7 to 12V. Power can be applied externally through the pin Vin or by giving voltage reference through the IORef pin.

Digital Inputs: It consists of 14 digital inputs/output pins, each of which provide or take up 40mA current. Some of them have special functions like pins 0 and 1, which act as Rx and Tx respectively , for serial communication, pins 2 and 3-which are external interrupts, pins 3,5,6,9,11 which provides pwm output and pin 13 where LED is connected.

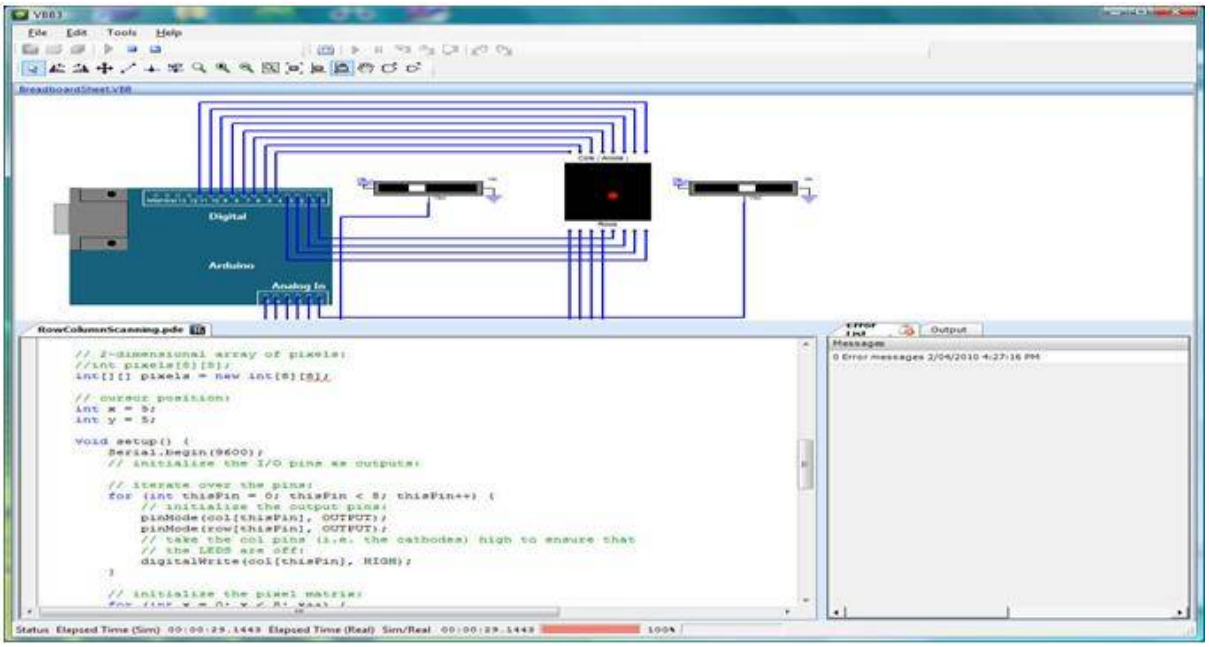
Analog inputs: It has 6 analog input/output pins, each providing a resolution of 10 bits.

ARef: It provides reference to the analog inputs

Reset: It resets the microcontroller when low.

How to program an Arduino?

The most important advantage with Arduino is the programs can be directly loaded to the device without requiring any hardware programmer to burn the program. This is done because of the presence of the 0.5KB of Bootloader which allows the program to be burned into the circuit. All we have to do is to download the Arduino software and writing the code.



The Arduino tool window consists of the toolbar with the buttons like verify, upload, new, open, save, serial monitor. It also consists of a text editor to write the code, a message area which displays the feedback like showing the errors, the text console which displays the output and a series of menus like the File, Edit, Tools menu.

Steps to program an Arduino

Programs written in Arduino are known as sketches. A basic sketch consists of 3 parts

1. Declaration of Variables
2. Initialization: It is written in the setup () function.
3. Control code: It is written in the loop () function.

- The sketch is saved with .ino extension. Any operations like verifying, opening a sketch, saving a sketch can be done using the buttons on the toolbar or using the tool menu.
- The sketch should be stored in the sketchbook directory.
- Chose the proper board from the tools menu and the serial port numbers.
- Click on the upload button or chose upload from the tools menu. Thus the code is uploaded by the bootloader onto the microcontroller.

Few of basic Adruino functions are:

- **digitalRead**(pin): Reads the digital value at the given pin.
- **digitalWrite**(pin, value): Writes the digital value to the given pin.
- **pinMode**(pin, mode): Sets the pin to input or output mode.
- **analogRead**(pin): Reads and returns the value.
- **analogWrite**(pin, value): Writes the value to that pin.
- **serial.begin**(baud rate): Sets the beginning of serial communication by setting the bit rate.

How to Design your own Arduino?

We can also design our own Arduino by following the schematic given by the Arduino vendor and also available at the websites. All we need are the following components- A breadboard, a led, a power jack, a IC socket, a microcontroller, few resistors, 2 regulators, 2 capacitors.

- The IC socket and the power jack are mounted on the board.
- Add the 5v and 3.3v regulator circuits using the combinations of regulators and capacitors.
- Add proper power connections to the microcontroller pins.
- Connect the reset pin of the IC socket to a 10K resistor.
- Connect the crystal oscillators to pins 9 and 10
- Connect the led to the appropriate pin.
- Mount the female headers onto the board and connect them to the respective pins on the chip.
- Mount the row of 6 male headers, which can be used as an alternative to upload programs.
- Upload the program on the Microcontroller of the readymade Adruino and then pry it off and place back on the user kit.

Reasons why Arduino is being preferred these days

1. It is inexpensive
2. It comes with an open source hardware feature which enables users to develop their own kit using already available one as a reference source.
3. The Arduino software is compatible with all types of operating systems like Windows, Linux, and Macintosh etc.
4. It also comes with open source software feature which enables experienced software developers to use the Arduino code to merge with the existing programming language libraries and can be extended and modified.
5. It is easy to use for beginners.
6. We can develop an Arduino based project which can be completely stand alone or projects which involve direct communication with the software loaded in the computer.
7. It comes with an easy provision of connecting with the CPU of the computer using serial communication over USB as it contains built in power and reset circuitry.

So this is some basic idea regarding an Arduino. You can use it for many types of applications. For instance in applications involving controlling some actuators like motors, generators, based on the input from sensors.

Netduino

What is Netduino?

Netduino is an open source electronics platform based on the .NET Micro Framework. It's Featuring a 32-bit Micro-Controller and a rich development environment, Netduino is suitable for engineers and hobbyists alike. This combines the ease of high-level coding and the raw features of Micro-Controllers.

Through this broad developer can use event-based programming, line-by-line debugging, multi-threading, breakpoints and more. Netduino board supported most of the Arduino shields. It's composed with ethernet port and microSD card reader. Most interesting thing on this board is, developer can write code in C# languages but Arduino supported by C++. Chris Walker invented this device and he is also founder of secret lab.

Features of Netduino

The Netduino is based on the Cortex-M Micro Processor running on .NET Micro Framework v4.3. For development, Developer can use Visual Studio, Windows or with Xamarin Studio on Mac OS X. Netduino board is packed with IO; including 22 General Purpose Input/Output (GPIO) ports, 6 of which support hardware Pulse Width Modulation (PWM) generation, 4 UARTs (serial communication), I2C, and SPI (Serial Peripheral Interface Bus).

Coding in Netduino on C#

One of the major part of the Netduino is the robust framework (.NET Micro Framework) that Netduino employs. If talking about Arduino uses the wiring language, and the Arduino IDE allows for a high level of control and visibility over the "bare metal" of the Micro-Controller. Netduino on the other hand, uses the familiar .NET framework, allowing programers to work in C# using Microsoft Visual Studio.

Netduino is More Powerful, But More Expensive

In general the computing power of the Netduino range is higher than that of Arduino. With some Netduino models working with a 32-bit processor running at up to 120 MHz, and plenty of RAM and FLASH memory to spare, the Netduino is appreciably faster than many of its Arduino counterparts.

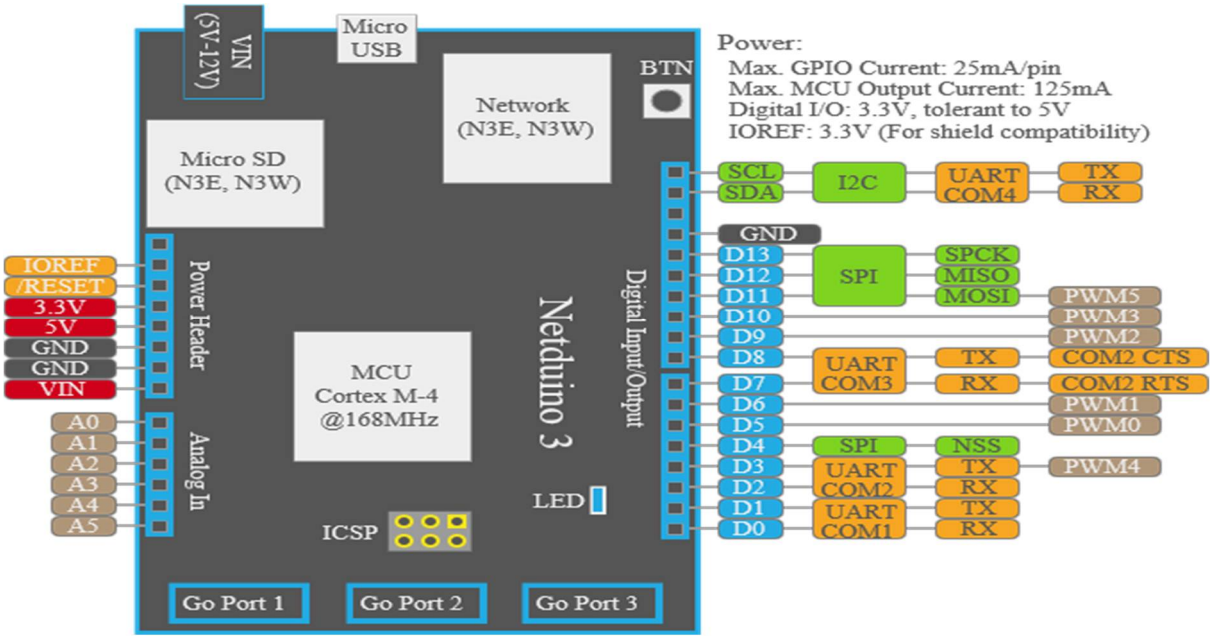
This additional power does come with a larger price tag, although Netduino costs per unit are not prohibitively more expensive. These costs can mount however, if Netduino units are needed at scale.

Netduino 3

Netduino 3 is offered in 3 different models, the N3 base model, N3 Ethernet model, and the N3 WiFi model; which vary by their internet connectivity mode and their code/flash storage size.

Features of Netduino 3

All Netduino3 models are support storage with SD cards up to 2GB. Both the Ethernet and WiFi models have a Micro SD slot built in to the board so that developer can use it for storage purpose.



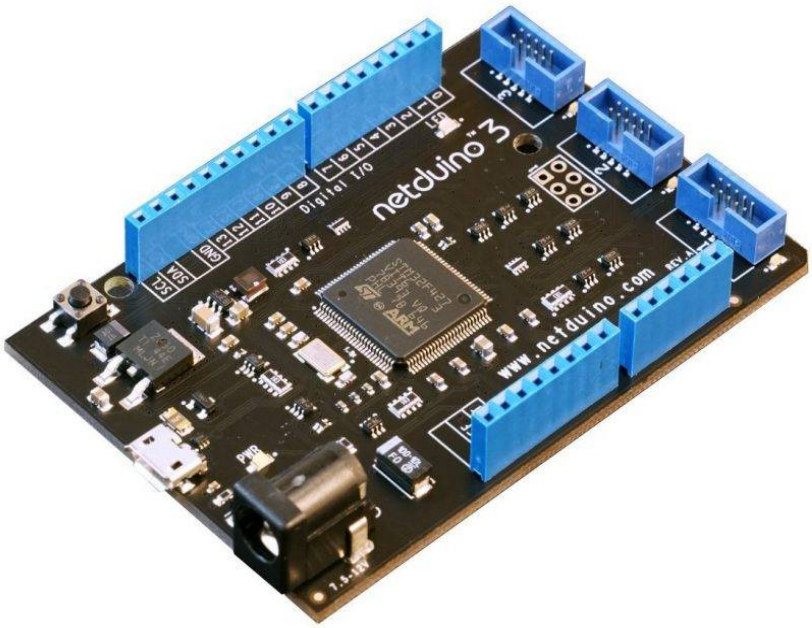
Technical Specification

Model	MCU	Flash	RAM	Network
N3	Cortex-M4 @ 168MHz	384KB	164+ KB	n/a
N3 Ethernet	Cortex-M4 @ 168MHz	1408KB	164+ KB	10/100Mbs Ethernet
N3 WiFi	Cortex-M4 @ 168MHz	1408KB	164+ KB	802.11b/g/n with SSL/TLS 1.2 Support

Netduino 3 Diagram

Netduino3 Diagram

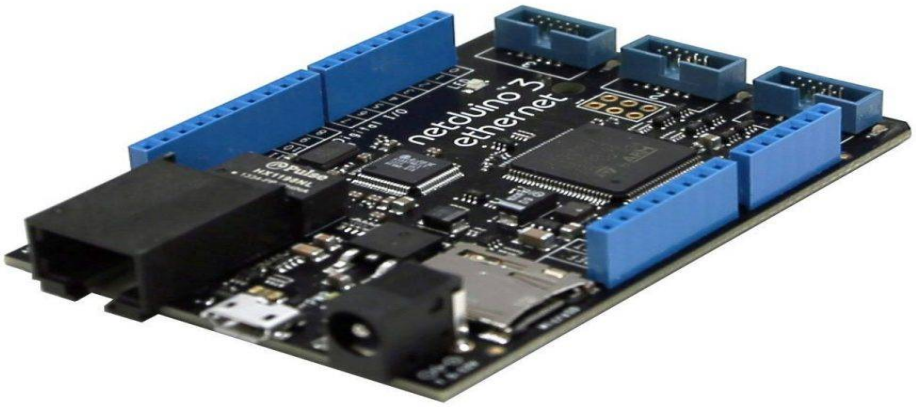
(i) Netduino – 3



Netduino 3 features an STM32F427VI or STM32F427VG ARM Cortex-M4 _MCU, 1 or 2MB of flash, and 256KB of RAM. A WiFi enabled version includes the TI CC3100 WiFi chip with 802.11b/g/n connectivity and supports built in SSL, WEP and WPA2. The Netduino 3 includes 3 GoBus 2.0 ports for plug and play components.

(ii) Netduino 3 Ethernet

(iii)



Netduino – 3 WiFi