Internet of Things

Introduction of Embedded System

- An embedded system is a combination of computer hardware and software designed for a specific function
- Embedded systems may also function within a larger system
- The systems can be programmable or have a fixed functionality
- The embedded systems can work without human intervention or with little human intervention
- The possible locations for an embedded system are : Industrial machines, consumer electronics, agricultural and processing industry devices, automobiles, medical equipment, cameras, digital watches, household appliances, airplanes, vending machines and toys, as well as mobile devices

Characteristics of embedded systems

- The main characteristic of embedded systems is that they are task-specific
- Additionally, embedded systems can include the following characteristics:
  - typically, consist of hardware, software and firmware
  - can be embedded in a larger system to perform a specific function, as they are built for specialized tasks within the system, not various tasks;
  - can be either microprocessor-based or microcontroller-based -- both are integrated circuits that give the system compute power;
  - are often used for sensing and real-time computing in internet of things (IoT) devices, which are devices that are internet-connected and do not require a user to operate;
  - can vary in complexity and in function, which affects the type of software, firmware and hardware they use; and
  - are often required to perform their function under a time constraint to keep the larger system functioning properly

Working of Embedded System

- Embedded systems always function as part of a complete device that's what's meant by the term embedded
- They are low-cost, low-power-consuming, small computers that are embedded in other mechanical or electrical systems
- Generally, they comprise a processor, power supply, and memory and communication ports
- Embedded systems use the communication ports to transmit data between the processor and peripheral devices -- often, other embedded systems -- using a communication protocol
- The processor interprets this data with the help of minimal software stored on the memory. The software is usually highly specific to the function that the embedded system serves

Structure of embedded systems

Accura Tequipment

Embedded systems vary in complexity but, generally, consist of three main elements:

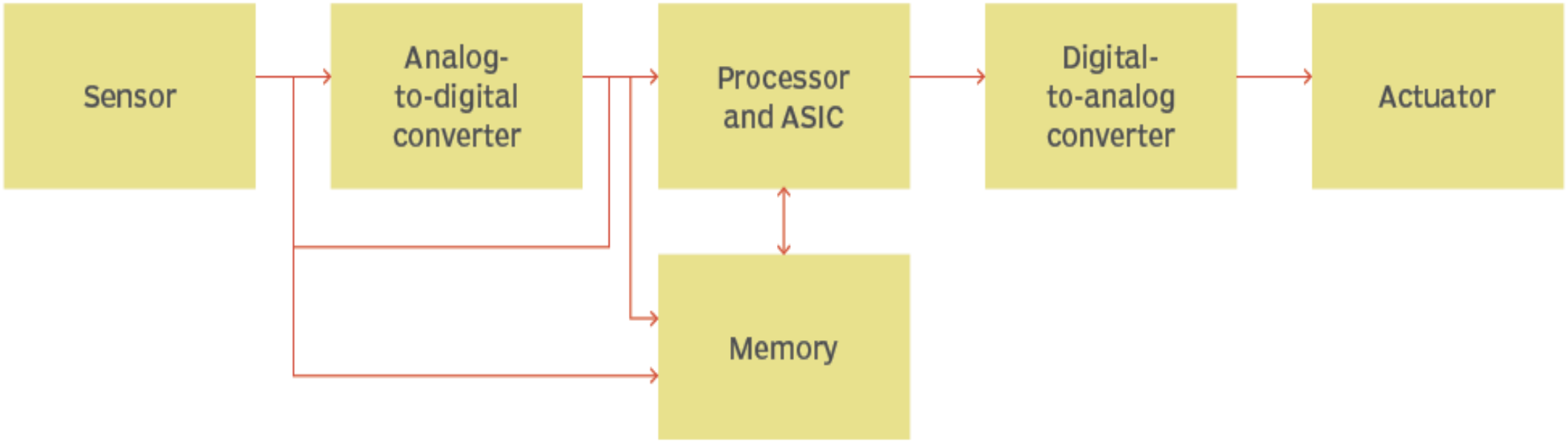
- **Hardware.** The hardware of embedded systems is based around microprocessors and microcontrollers. Microprocessors are very similar to microcontrollers and, typically, refer to a CPU (central processing unit) that is integrated with other basic computing components such as memory chips and digital signal processors (DSPs). Microcontrollers have those components built into one chip.

- **Software and firmware.** Software for embedded systems can vary in complexity. However, industrial-grade microcontrollers and embedded IoT systems usually run very simple software that requires little memory.

- **Real-time operating system.** These are not always included in embedded systems, especially smaller-scale systems. RTOSes define how the system works by supervising the software and setting rules during program execution

Basic Structure of an Embedded System

Accura Tequipment



Accura Tequipment

The basic structure of an embedded system includes the following components:

- **Sensors** convert physical sense data into an electrical signal.
- **Analog-to-digital (A-D) converters** change an analog electrical signal into a digital one.
- **Processors** process digital signals and store them in memory.
- **Digital-to-analog (D-A) converters** change the digital data from the processor into analog data.
- **Actuators** compare actual output to memory-stored output and choose the correct one.

The sensor reads external inputs, the converters make that input readable to the processor, and the processor turns that information into useful output for the embedded system.

Types of embedded systems

There are a few basic embedded system types, which differ in their functional requirements. They are:

- **Mobile embedded systems** are small-sized systems that are designed to be portable. Digital cameras are an example of this.

- **Networked embedded systems** are connected to a network to provide output to other systems. Examples include home security systems and point of sale (POS) systems.

- **Standalone embedded systems** are not reliant on a host system. Like any embedded system, they perform a specialized task. However, they do not necessarily belong to a host system, unlike other embedded systems. A calculator or MP3 player is an example of this.

- **Real-time embedded systems** give the required output in a defined time interval. They are often used in medical, industrial and military sectors because they are responsible for time-critical tasks. A traffic control system is an example of this.

Microcontroller

Accura Tequipment

- A microcontroller (MCU) is a small computer on a single integrated circuit that is designed to control specific tasks within electronic systems. It combines the functions of a central processing unit (CPU), memory, and input/output interfaces, all on a single chip

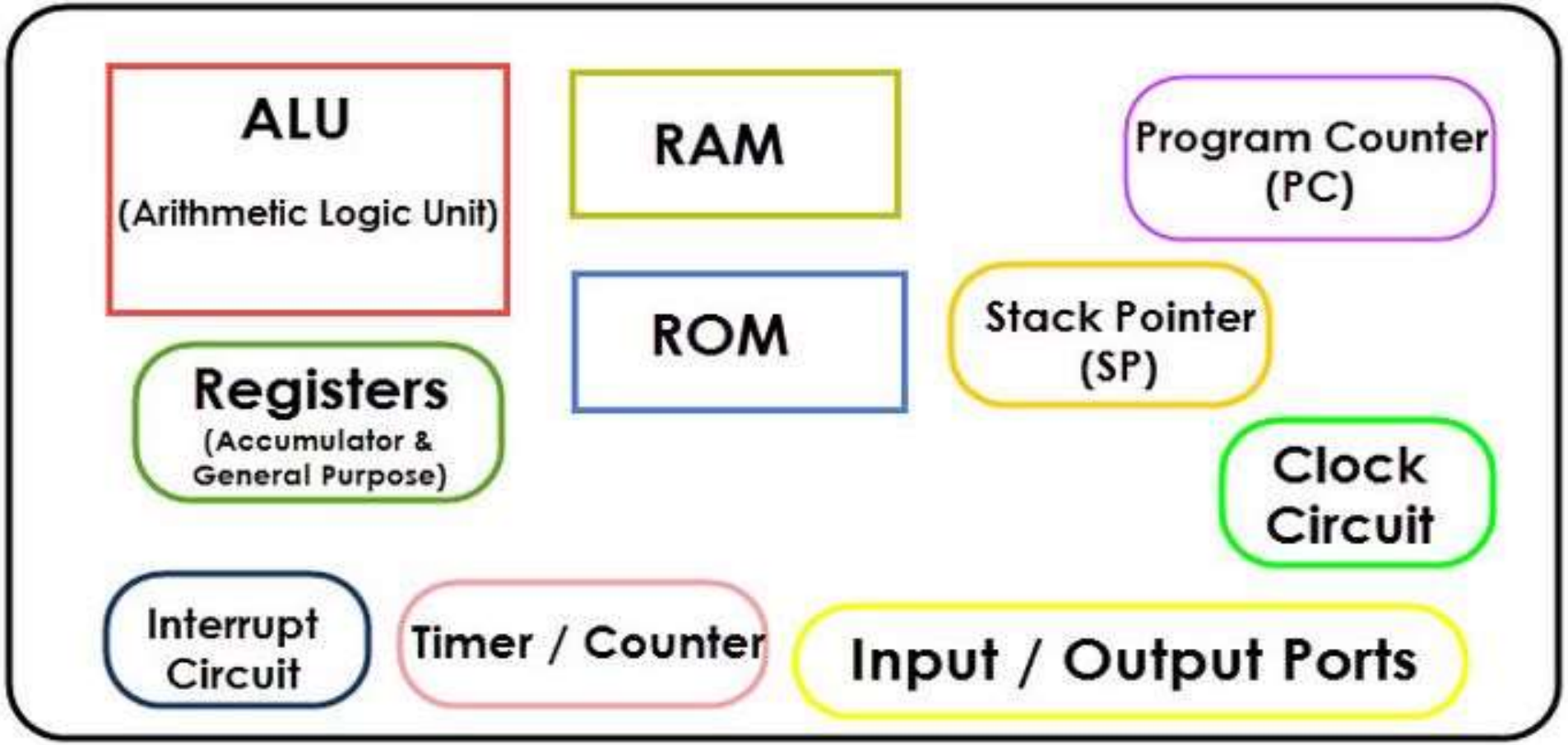
- Microcontrollers are widely used in embedded systems, such as home appliances, automotive systems, medical devices, and industrial control systems.

- Microcontrollers are programmable, which means that they can be customized to perform specific tasks.

- The programming languages used to write code for microcontrollers vary depending on the manufacturer and the type of microcontroller. Some of the commonly used programming languages include C, C++, and assembly language.

Accura Tequipment

Block Diagram of Microcontroller



Accura Tequipment

- The microcontroller has the processor, RAM, ROM, I/O ports & timer are all embedded together on one chip, therefore the designer cannot add any external memory, I/O ports or timer to it.

- The fixed amount of on chip ROM, RAM & number of I/O ports in microcontrollers makes them ideal for many applications in which cost & space are critical.

Selection of Microcontrollers

Accura Tequipment

1. First decide whether an 8-bit, 16-bit or 32-bit microcontroller can handle the task most effectively. Among other considerations in this category are:
   a. Speed
   b. Packaging
   c. Power consumption
   d. The amount of RAM and ROM on chip.
   e. The number of I/O pins & the timer on the chip.
   f. Upgradeability to higher-performance or lower power-consumption versions.

Selection of Microcontrollers

2. Availability of software development tools such as compiler, assemblers and debuggers

3. Wide availability in needed quantities and reliable sources of microcontroller

Which microcontroller is right for your IoT needs?

- The selection of a right IoT microcontroller will determine the success of your product

- The selection of the microcontroller heavily depends on the functional requirements of your IoT product, like does it require low power, or high-performance, wireless connectivity, or high-end security.

- IoT products in industries are more complex and have more computing power and energy constraints.

MICROPROCESSOR

- The microprocessor is an IC which is capable of performing arithmetic and logic operations. It is used as CPU(Central Processing Unit) in computers.

OR

- A microprocessor is a program controlled semiconductor device (IC), which fetches, decode & execute instructions.

OR

- Computer's Central Processing Unit (CPU) built on a single Integrated Circuit (IC) is called a microprocessor

Accura Tequipment

- A digital computer with one microprocessor which acts as a CPU is called microcomputer.

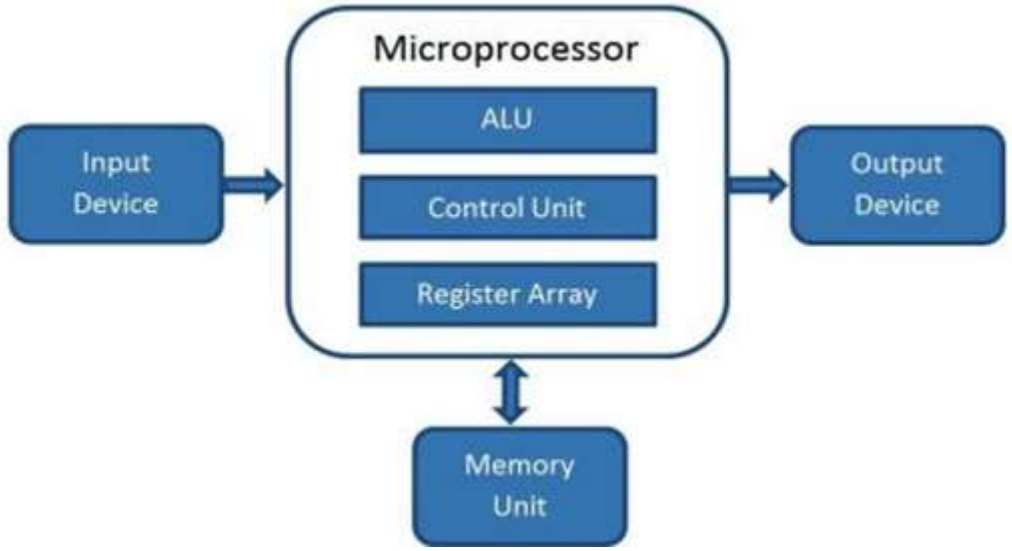
- It is a programmable, multipurpose, clock -driven, register-based electronic device that reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output

- The microprocessor contains millions of tiny components like transistors, registers, and diodes that work together.

Block Diagram of a Microprocessor

Accura Tequipment

- A microprocessor consists of an ALU, control unit and register array.
- **ALU** performs arithmetic and logical operations on the data received from an input device or memory.
- Control unit controls the instructions and flow of data within the computer.
- The **register array** consists of registers identified by letters like B, C, D, E, H, L, and accumulator.



Difference Between Microprocessor and Microcontroller

MICROPROCESSOR

- Microprocessors can be understood as the heart of a computer system
- The circuit is complex due to external connection
- The memory and I/O components are to be connected externally
- Many instructions to move data between memory & CPU
- Access time for memory & I/O device is more
- Single memory map for data & code
- Few pins are multifunctioning
- Microprocessor used for general applications

MICROCONTROLLER

- Microcontrollers can be understood as the heart of an embedded system.
- Microcontrollers are present on chip memory. The circuit is less complex.
- The memory and I/O components are available
- One or two instructions to move data between memory & CPU
- Less access time between I/O & memory device
- Separate memory map for data &code
- More pins are multifunctioning
- Microcontroller are used for specified tasks only.

Sensors

Accura Tequipment

- Sensors plays a key role when we use it in IoT
- Sensors or transducers represent physical devices that convert one form of energy into another
- Sensors convert a physical attribute into an electrical impulse to take the desired action
- The sensor is a device which detects changes in an environment or measures physical property, record, indicates or respond to it. so, different types of applications require different types of sensors to data from the environment

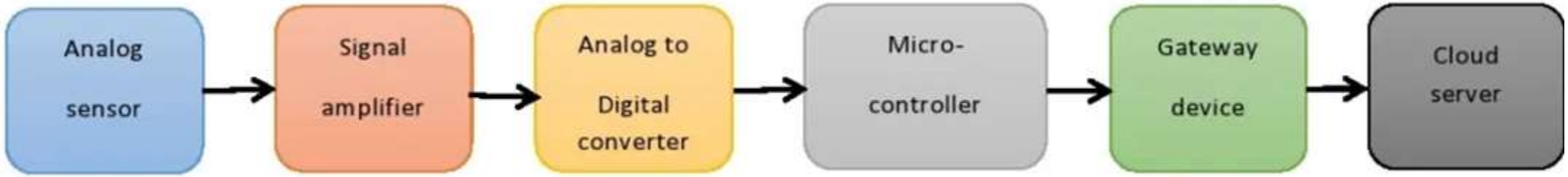
Role of sensors in IoT

Accura Tequipment

- In real-time data capture and analytics, using various types of sensors. It plays a crucial role in creating IoT devices
- Sensor technology in IoT makes it possible to collect data in almost any situation
- We use sensors in medical care, nursing care, industrial, logistics, transportation, agriculture, disaster prevention, tourism, regional businesses, and many more
- With the expansion of the fields in which sensors play an essential role, the market is still growing with various sensors.

Working of Sensor in IoT

Accura Tequipment



- IoT cloud servers and devices depend on sensors to collect real-time data.
- Sensors detect changes in their environment and then convert it to digital data. Because, the physical parameters are present in the analog signal like the temperature in Fahrenheit, speed in miles per hour, distance in feet, etc.
- Sensors need to connect to the cloud through some connectivity. Nowadays, we use wireless technologies such as Bluetooth, NFC, RF, Wi-Fi.

Sensor Classification

Accura Tequipment

- Passive & Active
- Analog & digital
- Scalar & vector

- **Passive Sensor :** Can not independently sense the input.

  Example- Accelerometer, soil moisture, water level and temperature sensors.

- **Active Sensor :** Independently sense the input.

  Example- Radar, sounder and laser altimeter sensors.

Accura Tequipment

- **Analog Sensor :**The response or output of the sensor is some continuous function of its input parameter.

  Example- Temperature sensor, LDR, analog pressure sensor and analog hall effect.

- **Digital sensor :**Response in binary nature. Design to overcome the disadvantages of analog sensors. Along with the analog sensor, it also comprises extra electronics for bit conversion.

  Example- Passive infrared (PIR) sensor and digital temperature sensor(DS1620).

- **Scalar sensor :** Detects the input parameter only based on its magnitude. The answer for the sensor is a function of magnitude of some input parameter. Not affected by the direction of input parameters.

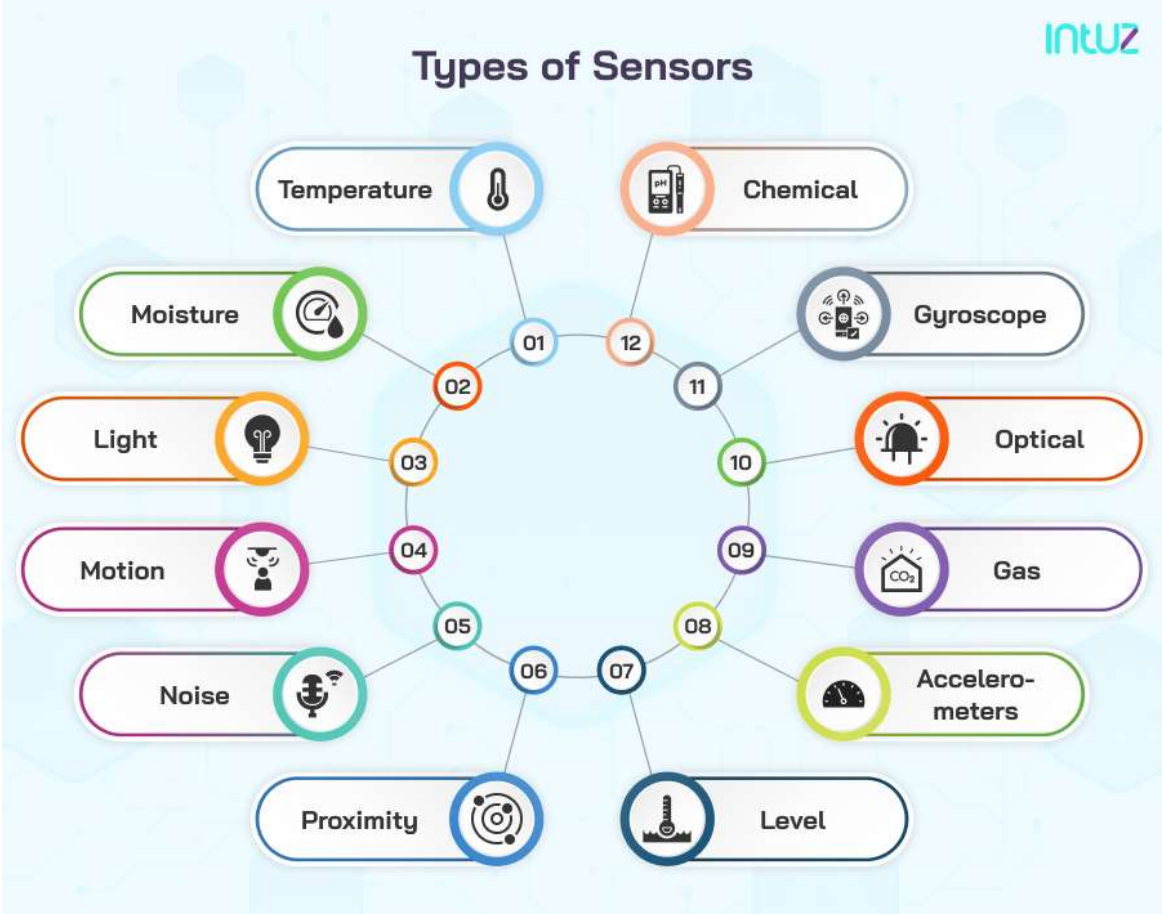
  Example – temperature, gas, strain, color and smoke sensor.

- **Vector sensor :**The response of the sensor depends on the magnitude of the direction and orientation of input parameter.

  Example – , gyroscope, magnetic field and motion detector sensors.

Types of Sensors

Accura Tequipment

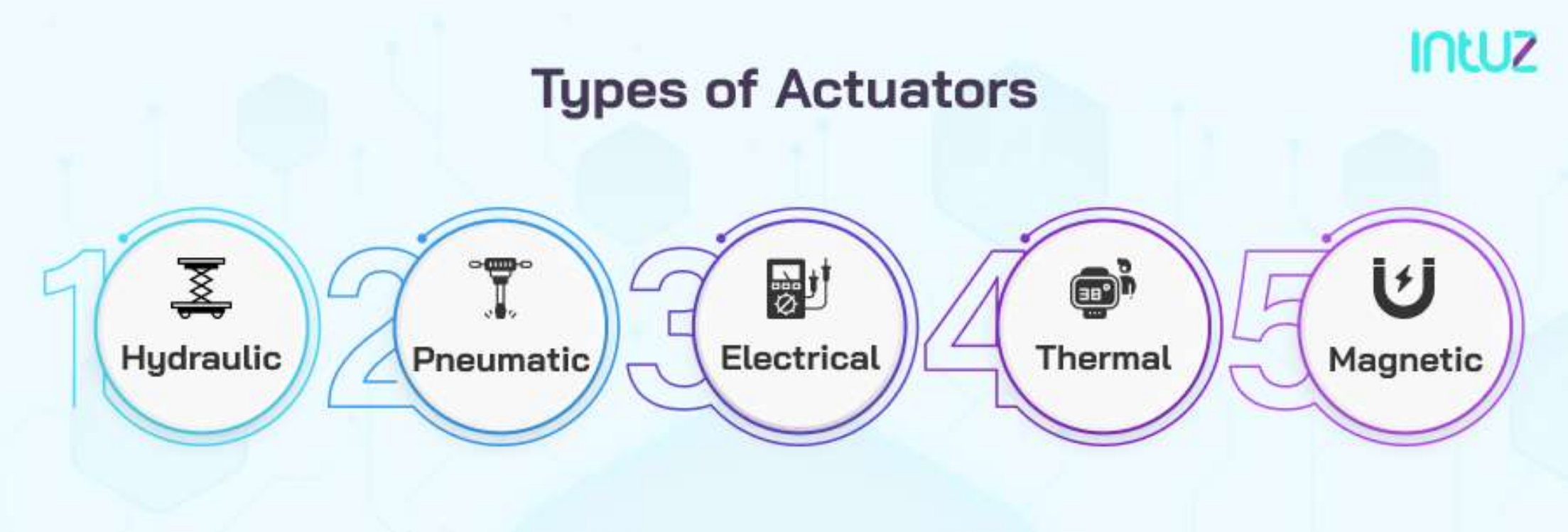


Actuator

- Actuators are simply devices that convert energy into action or motion.
- **Actuators set things in motion** Depending on "smart" decisions, actuators convert the control signals into actions – which is in most cases motion, but also light and heat.
- Actuators are the workhorses of the Internet of Things and enable the automation or remote control of a wide range of processes in sectors that include consumer, industrial, healthcare, and transportation.
- IoT or smart products that use actuators differ from the applications that solely rely on sensors for monitoring.
- For IoT products to be effective and function properly, the right actuator has to be selected.

Types of Actuators

Accura Tequipment



Hydraulic Actuators

Accura Tequipment

- These actuators harness hydraulic power to perform mechanical functions and operations.
- These types of actuators are powered by a cylinder or a fluid motor.
- According to the requirements and recommendations, the mechanical motion is converted into oscillatory, linear, or rotary.
- These industrial actuators are rugged and can exert a large amount of force at high speed. They are present in construction, manufacturing, agricultural and mining heavy machinery, using the movement of fluids under pressure to achieve movements.
- They are powerful, they often leak or bust and require a lot of maintenance.

Pneumatic Actuators

- Pneumatic actuators create two types of motions, rotary or linear.
- They are powered by a vacuum or compressed air at high pressure to implement the required type of motion.
- Compared to other types of actuators, pneumatic actuators are low-cost and low-maintenance actuators. They are durable and will withstand harsh conditions.
- They also can quickly activate and deactivate but should be monitored to ensure that their air pressure is maintained.

Accura Tequipment

Electrical Actuators:

- The electrical actuators, a motor converts electrical energy into mechanical motion.
- These actuators are powered by electricity and provide precision control.
- These actuators are heavily used in industrial settings to automate mechanical operations.
- They function quietly but can achieve a high level of precision in their movements they are highly programmable.
- Electrical actuators may not withstand the harshest operating environments.

Thermal actuators

- Thermal actuators have thermal-sensitive material fitted inside, which is used to produce linear motion.
- The word thermal implies that these actuators are used in response to temperature changes.
- The most popular use case includes shutting off valves and operating latches or switches.

Magnetic Actuators

- Magnetic actuators convert electromagnetic energy into mechanical output and operate in a linear or rotary direction.
- Magnetic actuators can provide continuous mechanical operation and are popularly used in the automotive and aerospace industries.

Accura Tequipment

Actuators can also be classified according to the type of movement they produce. Select them according to the work you want them to do.

1.Rotary :

- These are cylindrical actuators that complete angular, rotary, or torque movement.
- They can use electrical, pneumatic, or hydraulic energy to achieve their mechanical rotation.

2.Linear :

- The motion produced by a linear actuator is a straight movement in the horizontal or vertical plane.
- These are often pneumatic devices.

3. Oscillating :

- These actuators produce a forward and backward rotary or angular movement at a specific frequency.

APPLICATIONS OF ACTUATORS

Accura Tequipment

1. A refrigerator unit that has a thermostat may use temperature sensors to monitor the temperature. If the temperature exceeds the thermostat threshold, the sensor automatically sends data to a controller that instructs an actuator to activate the refrigeration.

2. In a smart home, strategically positioned light sensors monitor daylight levels on the property. As the evening unfolds, the sensor sends data about light levels to the smart home controller that signals an actuator that causes window shades to fall and the lights to come on automatically when it becomes dark.

3. On farms, humidity sensors that monitor soil moisture can send data to a controller if the soil becomes too dry. The controller can then signal an actuator that switches on irrigation to water the crops.

Introduction to IoT

- **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.

- IoT is network of interconnected computing devices which are embedded in everyday objects, enabling them to send and receive data.

- 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 very few of the categorical examples where IoT is strongly established.

Accura Tequipment

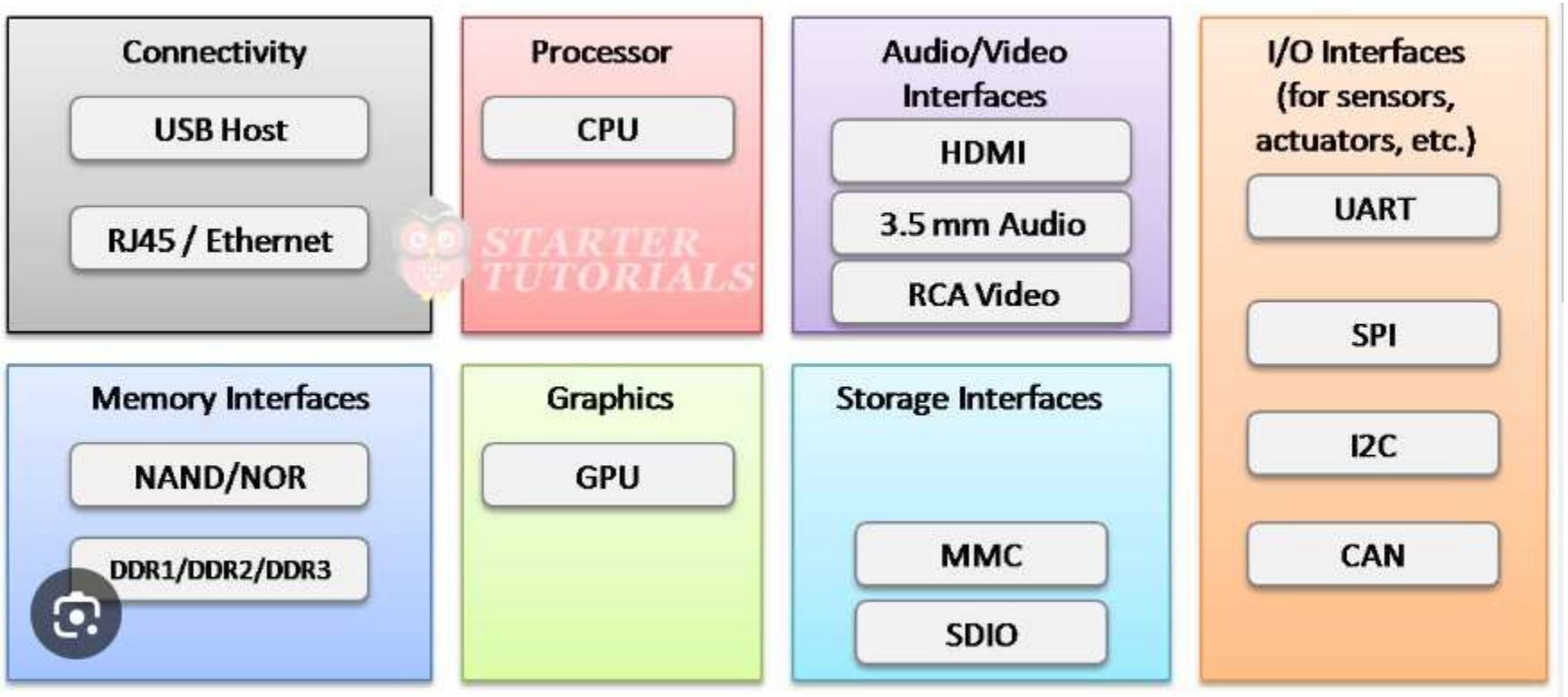
- Internet of Things (IoT) technology has a wide variety of applications and use of Internet of Things is growing so faster.

- Depending upon different application areas of Internet of Things, it works accordingly as per it has been designed/developed. But it has not a standard defined architecture of working which is strictly followed universally.

- The architecture of IoT depends upon its functionality and implementation in different sectors. Still, there is a basic process flow based on which IoT is built.

IoT Block Diagram

Accura Tequipment



Hardware Interfacing Protocol

- I2C Protocol
- UART Protocol
- SPI Protocol
- CAN Protocol
- MODBUS Protocol

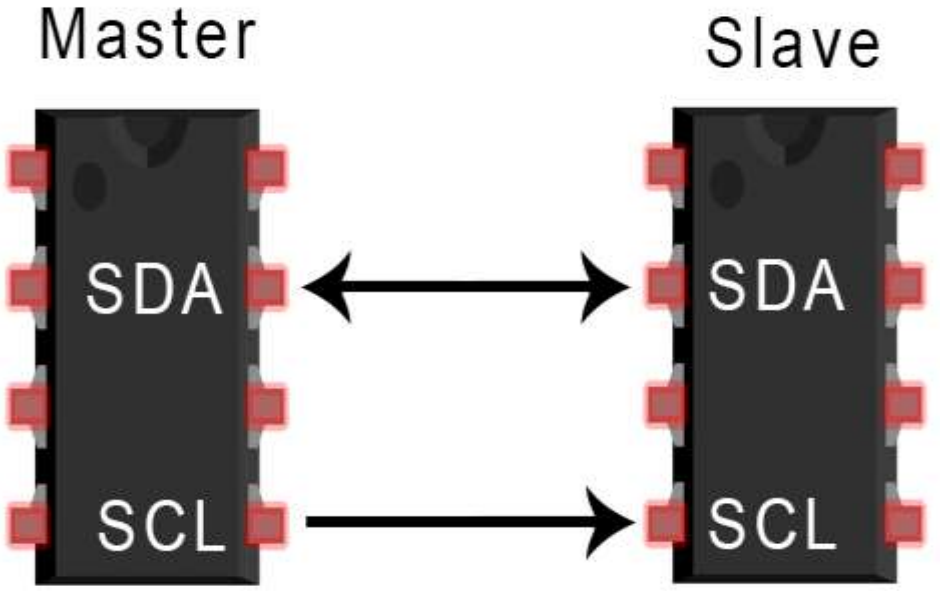
I2C Protocol

Accura Tequipment

- I2C stands for Inter-Integrated Circuit serial communication protocol. In other words data is sent bit by bit serially to the receiver.
- I2C only uses two wires to transmit data between devices:

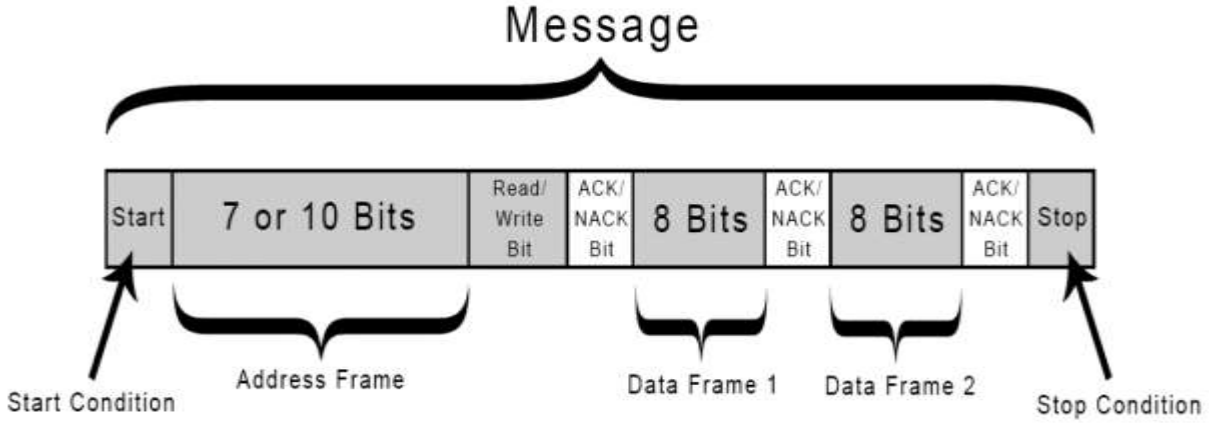
  SDA (Serial Data) – line for the master and slave to send and receive data.

  SCL (Serial Clock) – The line that carries the clock signal.
- I2C is a synchronous communication protocol. The master and slave both use the same clock to synchronize the bits and for sampling the data.
- It compares with other serial communication protocols like UART, SPI with I2C being the hybrid of the two.



Working of I2C communication

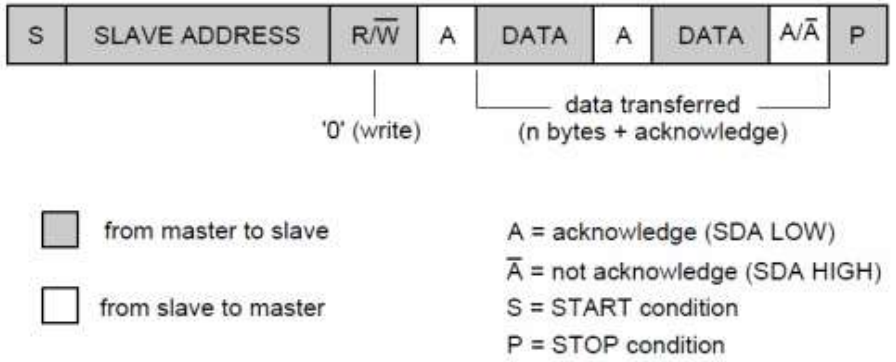
- In I2C, data is transferred in messages.
- Messages are broken up into frames of data.
- Each message has an address frame that contains the binary address of the slave, and one or more data frames that contain the data being transmitted. The message also includes start and stop conditions, read/write bits, and ACK/NACK bits between each data frame



Start Condition: SDA line switches from a high voltage level to a low voltage level before the SCL line switches from high to low.

Stop Condition: The SDA line switches from a low voltage level to a high voltage level after the SCL line switches from low to high.

Address Frame: The address is used to locate and send or retrieve the data from the slave. It is generally 7-10 bits long. Hence it can identify 2^N slaves uniquely. It identifies the slave when the master wants to talk to it.



Read/Write Bit: single bit specifying whether the master is sending data to the slave (low voltage level) or requesting data from it (high voltage level). Generally, 0 indicates write operation and 1 indicates read operation.

ACK/NACK Bit: frame in a message is followed by an acknowledge/no-acknowledge bit. If an address frame or data frame was successfully received, an ACK bit is returned to the sender from the receiving device.

The Data Frame: is the 8 bits long part of the message which is the actual data that we are sending or have requested from a slave. It is always sandwiched between the ACK/NACK bits for verification. After getting the acknowledgement this 8 bits are sent serially with MSB first.

Working

- The master first sends the start bit followed by the address of the slave it wants to communicate with.
- To do so the master pulls the SDA line from HIGH to LOW before SCL goes from HIGH to LOW.
- Each slave compares the address with its own.
- If the address matches with that of the slave it sends and acknowledgement by pulling the SDA line low for one bit.
- The address is followed by read or write bit, after acknowledgement is received the data is transferred from master to slave or vice versa.
- The communication is terminated after the master pulls SDA HIGH before SCL goes HIGH.

ADVANTAGES of I2C

- Only uses two wires
- Supports multiple masters and multiple slaves
- ACK/NACK bit gives confirmation that each frame is transferred successfully
- Hardware is less complicated than with UARTs
- Well known and widely used protocol

DISADVANTAGES of I2C

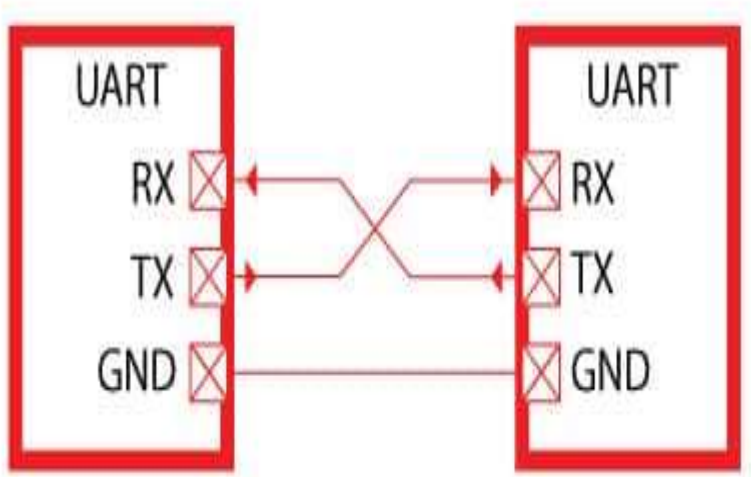
- Slower data transfer rate than SPI
- The size of the data frame is limited to 8 bits
- More complicated hardware needed to implement than SPI

UART Protocol

Accura Tequipment

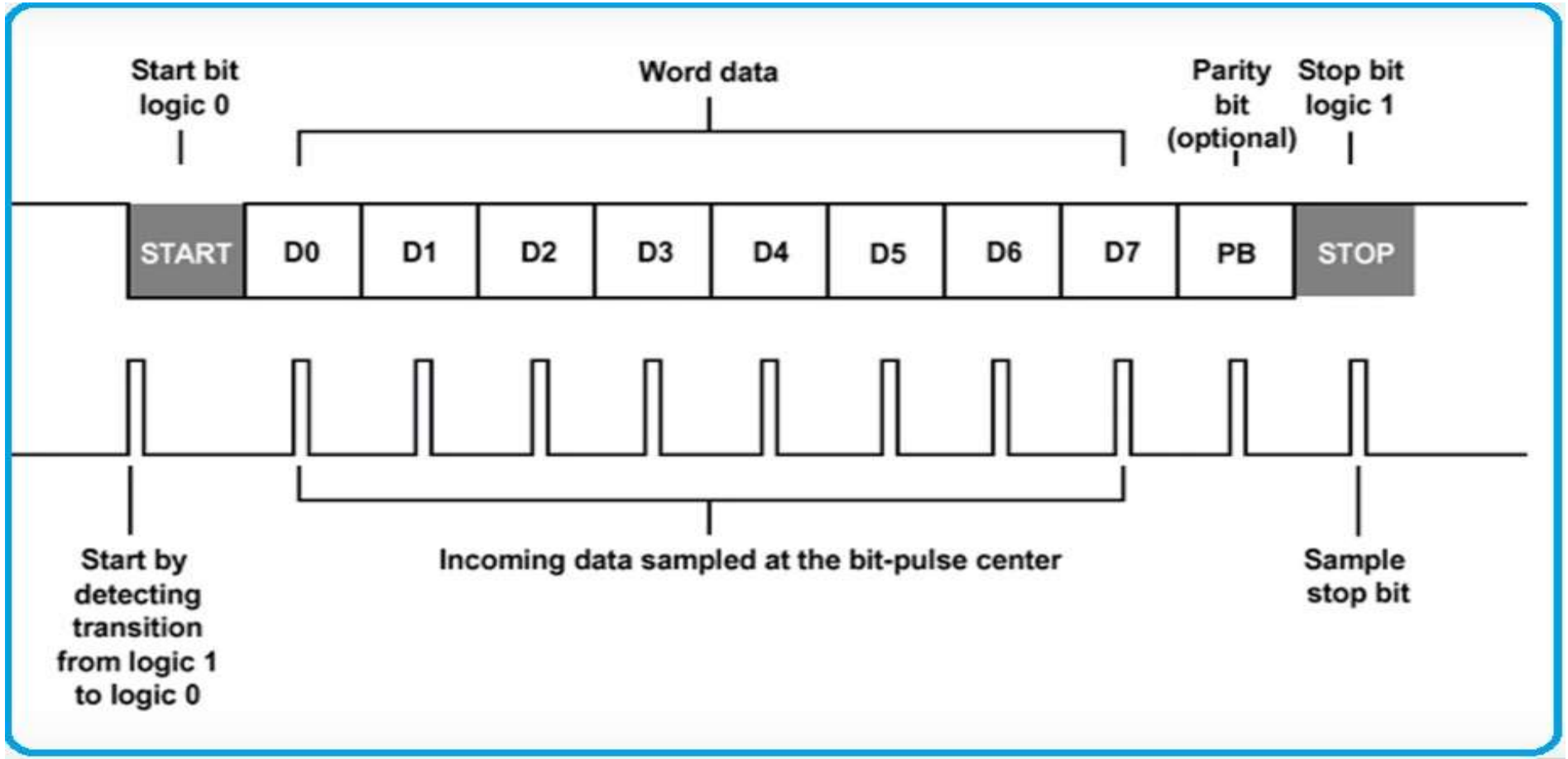
- UART stands for Universal Asynchronous Receiver/Transmitter.
- A UART's main purpose is to transmit and receive serial data.
- One of the best things about UART is that it only uses two wires to transmit data between devices.

- UART is a character-oriented protocol that means data is sent byte by byte.
- In UART communication, the transmitter wire of the first device is connected with the receiver wire of the second device, and the transmitter wire of the second device is connected with the receiver wire of the first device.



Working of UART communication

Accura Tequipment



- The Data frame of UART consists of the starting bit, main data then comes the parity bit, this parity bit is optional. some devices need this, some don’t.
- It is useful only if the devices need to check the error present in the data stream.
- Initially, the transmission line and reception line of the UART is high which indicates that the line is idle and there is no data transmission.
- When a Transmitting device wants to start the communication, it pulls the transmitter line low which means it goes to zero and due to this the receiver will understand that yes the transmitter wants to send the data.
- When the transmitting line goes low, it stays low for one clock pulse. After that there is a frame of 8 bits data that needs to be delivered at the receiver side, if there is 1 in the data it is shown as 5V in electronics hardware and 0 in the data is shown as 0V.
- After that parity bit comes, and finally, there is a stop bit to notify the receiver about the end of the communication.
- The stop bit is actually logic high, it stays high for one clock pulse and furthermore to notify the receiver.
- So basically UART data frame is a 10-bit data frame including start bit and stop bit if we use parity bit then it becomes 11-bit data.
- UART supports a wide range of RS-32, RS-422, RS-485 serial protocols. Examples of serial interfaces include Universal Serial Bus (USB), Recommended Standard No. 232 (RS-232).
- Internet of Things (IoT) devices may support UART to send and transmit signals wirelessly. Manufacturers install UART interfaces on IoT boards to review serial console logs and complete any debug activity required.

SPI Protocol

Accura Tequipment

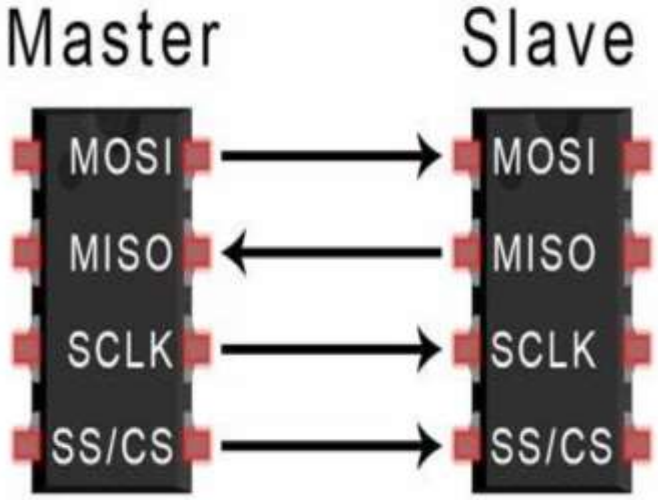
- SPI stands for the Serial Peripheral Interface. It is a serial communication protocol that is used to connect low-speed devices.
- It was developed by Motorola in the mid-1980 for inter-chip communication.
- It is commonly used for communication with flash memory, sensors, real-time clock (RTC), analog-to-digital converters, and more.
- It is a full-duplex synchronous serial communication, which means that data can be simultaneously transmitted from both directions.
- The main advantage of the SPI is to transfer the data without any interruption. Many bits can be sent or received at a time in this protocol.

Accura Tequipment

SPI Interface

The SPI protocol uses the four wires for the communication.

- **MOSI:** MOSI stands for Master Output Slave Input. It is used to send data from the master to the slave.
- **MISO:** MISO stands for Master Input Slave Output. It is used to send data from the slave to the master.
- **SCK or SCLK (Serial Clock):** is used to the clock signal.
- **SS/CS (Slave Select / Chip Select):** is used by the master to send data by selecting a slave.



Accura Tequipment

Advantages of SPI

1. The main advantage of the SPI is to transfer the data without any interruption.
2. It is simple hardware.
3. It provides full-duplex communication.
4. There is no need for a unique address of the slave in this protocol.
5. This protocol does not require precise oscillation of slave devices because it uses the master's clock.
6. In this, software implementation is very simple.
7. It provides high transfer speed.
8. Signals are unidirectional.
9. It has separate lines of MISO and MOSI, so the data can be sent and received at the same time.

Accura Tequipment

Disadvantages of SPI

1. Usually, it supports only one master.
2. It does not check the error like the UART.
3. It uses more pins than the other protocol.
4. It can be used only from a short distance.
5. It does not give any acknowledgment that the data is received or not.

Applications of SPI

- Memory: SD Card, MMC, EEPROM, and Flash.
- Sensors: Temperature and Pressure.
- Control Devices: ADC, DAC, digital POTS, and Audio Codec.
- Others: Camera Lens Mount, Touchscreen, LCD, RTC, video game controller, etc.

CAN Protocol

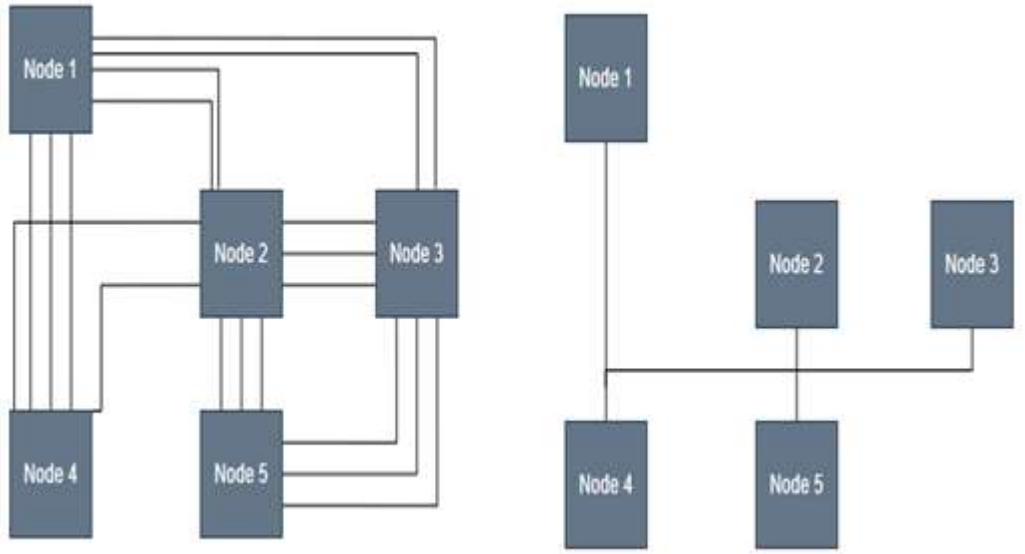
Accura Tequipment

- CAN stands for Controller Area Network protocol.
- It is a protocol that was developed by Robert Bosch in around 1986.
- The CAN protocol is a standard designed to allow the microcontroller and other devices to communicate with each other without any host computer.
- **The feature that makes the CAN protocol unique among other communication protocols is the broadcast type of bus.**
- Here, broadcast means that the information is transmitted to all the nodes. The node can be a sensor, microcontroller, or a gateway that allows the computer to communicate over the network through the USB cable or ethernet port.
- The CAN is a message-based protocol, which means that message carries the message identifier, and based on the identifier, priority is decided.
- There is no need for node identification in the CAN network, so it becomes very easy to insert or delete it from the network.
- It is a serial half-duplex and asynchronous type of communication protocol.
- The CAN is a two-wired communication protocol as the CAN network is connected through the two-wired bus.

Need of CAN Protocol

Accura Tequipment

- Increasing number of electronic devices in vehicles
- Example: Over 7 TCU (Transmission Control Unit)s for various subsystems (dashboard, transmission control, engine control unit)
- One-to-one connection leads to high communication speed but also high wiring complexity and cost
- Example: A single dashboard requiring 8 connectors
- Introduction of CAN Protocol as a centralized solution
- Requires only two wires (CAN high and CAN low)
- Reduces wiring complexity and cost
- Ensures efficient communication among electronic control units

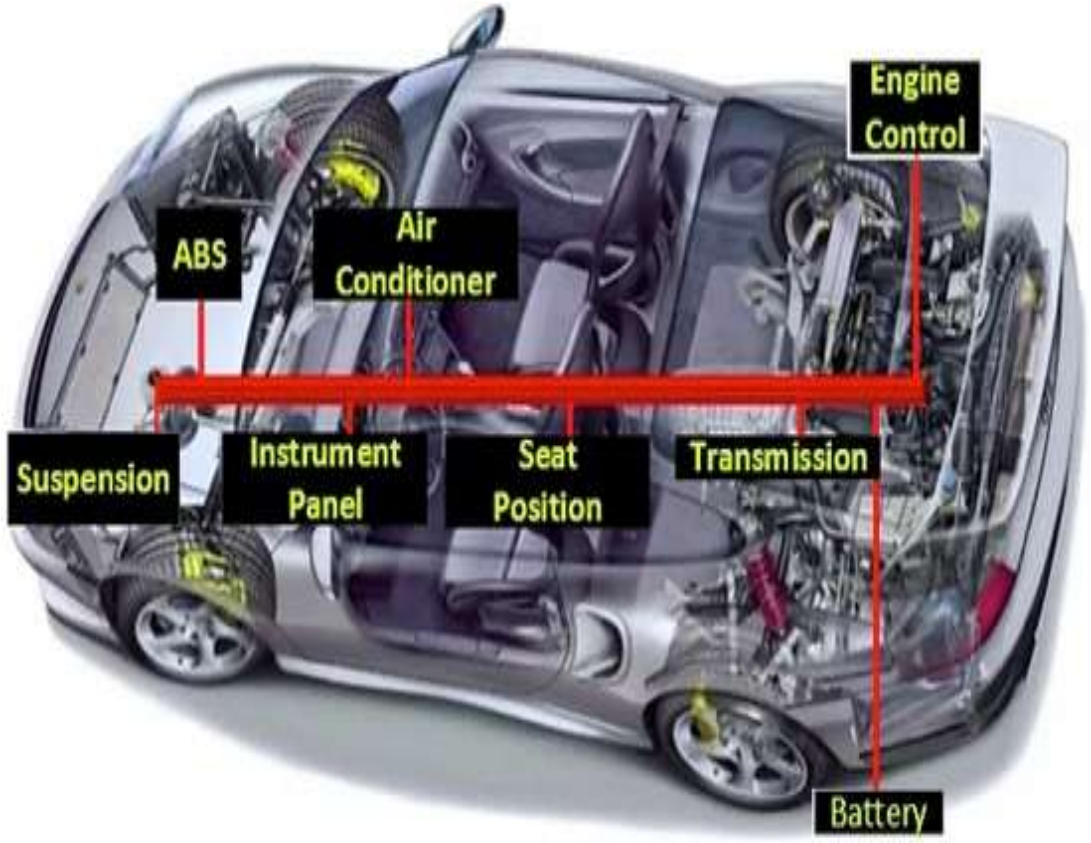


Benefits of CAN Protocol

- **Low cost:** Since a CAN serial bus uses two wires (with high-volume and low-cost production), it offers a good price-to-performance ratio.
- **Reliable:** CAN offers excellent error-detection and error-handling mechanisms, which provides highly reliable transmission. It's also largely immune to electromagnetic interference.
- **Flexible:** CAN nodes are not limited by the protocol and can be easily connected or disconnected.
- **Fast:** CAN supports **a** data rate of 1 MBit/s @ 40m bus length.
- **Multi-master communication:** Any node can access the bus
- **Fault confinement:** Faulty nodes do not disturb the communication.
- **Broadcast capabilities:** Messages can be sent to one /many/all nodes.
- **Standardized:** ISO has standardized the CAN protocol via ISO-DIS 11898 (for high-speed applications) and ISO-DIS 11519-2 (for low-speed applications). The CAN protocol is also standardized by industry organizations, such as the SAE-Society of Automotive Engineers.

Importance of Onboard Diagnostics (OBD) and CAN Protocol

- OBD: Vital system for vehicle diagnostics and reporting.
- Diagnostic Trouble Codes (DTCs): Enable troubleshooting of vehicle problems.
- "Check Engine" Light: Triggers indication of potential issues.
- Handheld Diagnostic Devices: Used to read engine codes for diagnosis.
- Signaling Protocol: Facilitates data transmission, commonly utilizing CAN.
- Efficiency and Reliability: CAN ensures efficient and reliable communication.
- Seamless Troubleshooting: Integration of OBD and CAN enhances diagnostic accuracy.



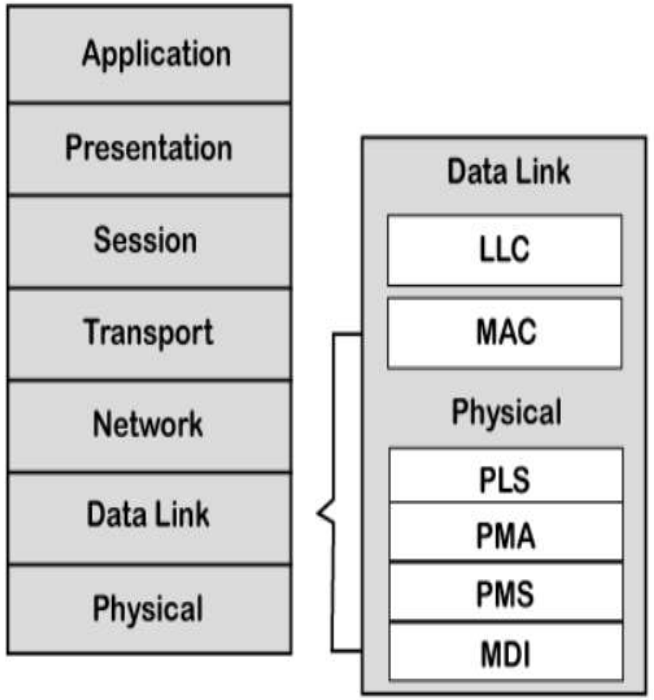
Applications of CAN protocol

1. Automotive (passenger vehicles, trucks, buses)
2. Electronic equipment for aviation and navigation
3. Industrial automation and mechanical control
4. Elevator and escalators
5. Building automation
6. Medical instruments and equipment
7. Marine, medical, industrial, medical

CAN layered architecture

Accura Tequipment

The OSI model partitions the communication system into 7 different layers. But the CAN layered architecture consists of two layers, i.e., data-link layer and physical layer.

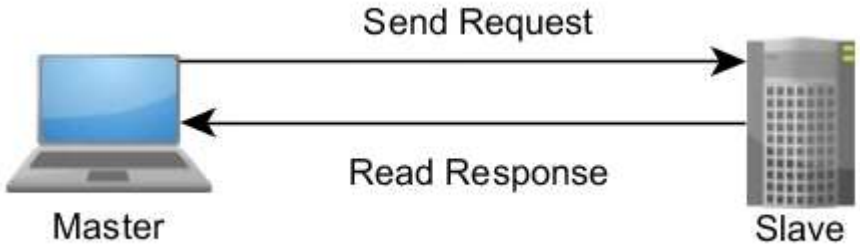


- **Data-link layer:** This layer is responsible for node to node data transfer. It allows you to establish and terminate the connection. It is also responsible for detecting and correcting the errors that may occur at the physical layer. Data-link layer is subdivided into two sub-layers:
  - **MAC:** stands for Media Access Control. It defines how devices in a network gain access to the medium. It provides Encapsulation and Decapsulation of data, Error detection, and signaling.
  - **LLC:** LLC stands for Logical link control. It is responsible for frame acceptance filtering, overload notification, and recovery management.
- **Physical layer:** The physical layer is responsible for the transmission of raw data. It defines the specifications for the parameters such as voltage level, timing, data rates, and connector.

MODBUS Protocol

Accura Tequipment

- Modbus is a serial communication protocol developed by Modicon published by Modicon® in 1979 for use with its programmable logic controllers (PLCs) and to make communication possible between automation devices.
- Originally implemented as an application-level protocol intended to transfer data over a serial layer, Modbus has expanded to include implementations over serial, TCP/IP, and the user datagram protocol (UDP).
- Modbus is a request-response protocol implemented using a master-slave relationship. In a master-slave relationship, communication always occurs in pairs—one device must initiate a request and then wait for a response—and the initiating device (the master) is responsible for initiating every interaction.



- Typically, the master is a human machine interface (HMI) or Supervisory Control and Data Acquisition (SCADA) system and the slave is a sensor, programmable logic controller (PLC), or programmable automation controller (PAC). The content of these requests and responses, and the network layers across which these messages are sent, are defined by the different layers of the protocol.

Accura Tequipment

- The Modbus protocol is used for transmitting information over serial lines between electronic devices. The device requesting the information is called the Modbus Client and the devices supplying information are Modbus Servers.
- In a standard Modbus network, there is one Client and up to 247 Servers, each with a unique Server Address from 1 to 247. The Client can also write information to the Servers.
- Modbus protocol utilizes a simple message structure, which makes it easier to deploy.
- Modbus is an open protocol, meaning that it's free for manufacturers to build into their equipment without having to pay royalties. It has become a standard communications protocol in industry, and is now the most commonly available means of connecting industrial electronic devices.
- It is used widely by many manufacturers throughout many industries. Modbus is typically used to transmit signals from instrumentation and control devices back to a main controller or data gathering system, for example a system that measures temperature and humidity and communicates the results to a computer. Modbus is often used to connect a supervisory computer with a remote terminal unit (RTU) in supervisory control and data acquisition (SCADA) systems.
- Versions of the Modbus protocol exist for serial lines (Modbus RTU and Modbus ASCII) and for Ethernet (Modbus TCP).

Application areas of Modbus Protocol

The main application of Modbus is in multi master-slave applications to communicate between smart devices and sensors and instruments to monitor field devices using Desktop PCs and Human machine interfaces. Modbus is a perfect protocol for RTU related applications where in which wireless communication is required. That is why it is used in endless gas and oil substation utilities. Modbus is an industrial protocol, moreover building, infrastructure, transportation and energy applications can also utilize the benefits of Modbus. The common factor is the messaging structure that all devices support.

- **Healthcare: For automated temperature monitoring**

  Modbus can be used by hospital's IT department to monitor the temperature in single interface. Data from different floors can be directly taken via RS485 Modbus ADC devices.

- **Transportation: Traffic behavior detection**

  The abnormal behavior of traffic can be detected by the cross referring with normal traffic patterns obtained through the Modbus TCP transactions.

- **Home automation: Easy transfer of data**

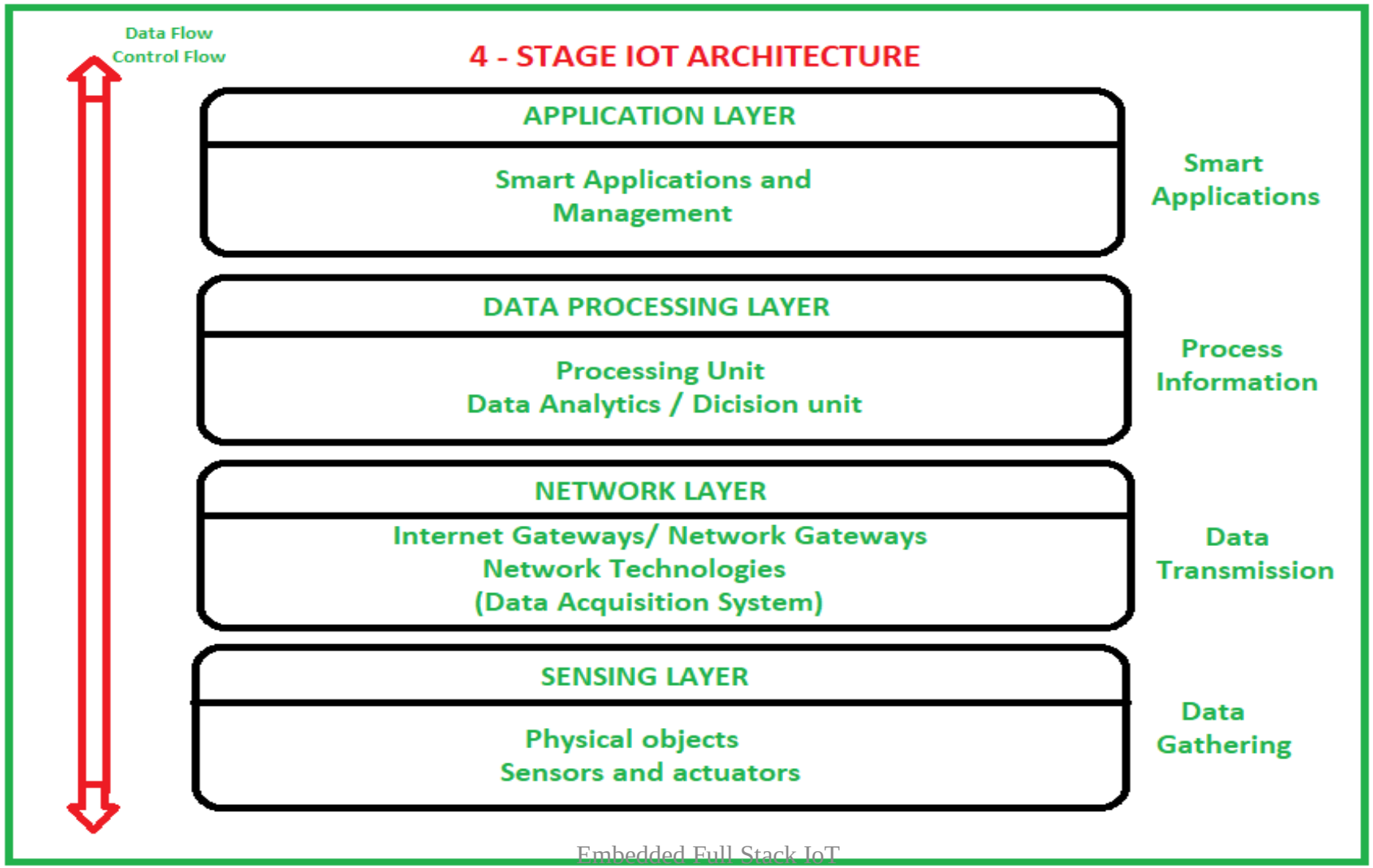
  For transferring data from different sensors used in home automation devices can be done through Modbus protocol. Since the data can be transferred via single layer it will be much easier when we compare other protocols

- **Other Industries**

  Another main application of Modbus is while connecting industrial devices that need to communicate with other automation equipment. Other major industries include Gas and oil, Renewable energy sources like Wind, Solar, Geothermal and Hydel etc.

IoT Architecture

Accura Tequipment



Accura Tequipment

The fundamental architecture of IoT i.e., 4 Stage IoT architecture. It can be divided as :

- Sensing Layer
- Network Layer
- Data processing Layer
- Application Layer

1. Sensing Layer

- The sensing layer is the first layer of the IoT architecture and is responsible for collecting data from different sources.
- This layer includes sensors and actuators that are placed in the environment to gather information about temperature, humidity, light, sound, and other physical parameters.
- These devices are connected to the network layer through wired or wireless communication protocols.

2. Network Layer

- The network layer of an IoT architecture is responsible for providing communication and connectivity between devices in the IoT system.
- It includes protocols and technologies that enable devices to connect and communicate with each other and with the wider internet.
- Examples of network technologies that are commonly used in IoT include WiFi, Bluetooth, Zigbee, and cellular networks such as 4G and 5G.
- Additionally, the network layer may include gateways and routers that act as intermediaries between devices and the wider internet, and may also include security features such as encryption and authentication to protect against unauthorized access

Accura Tequipment

3. Data processing Layer

- The data processing layer of IoT architecture refers to the software and hardware components that are responsible for collecting, analyzing, and interpreting data from IoT devices.
- This layer is responsible for receiving raw data from the devices, processing it, and making it available for further analysis or action.
- The data processing layer includes a variety of technologies and tools, such as data management systems, analytics platforms, and machine learning algorithms.
- These tools are used to extract meaningful insights from the data and make decisions based on that data.
- **Example** of a technology used in the data processing layer is a data lake, which is a centralized repository for storing raw data from IoT devices

Accura Tequipment

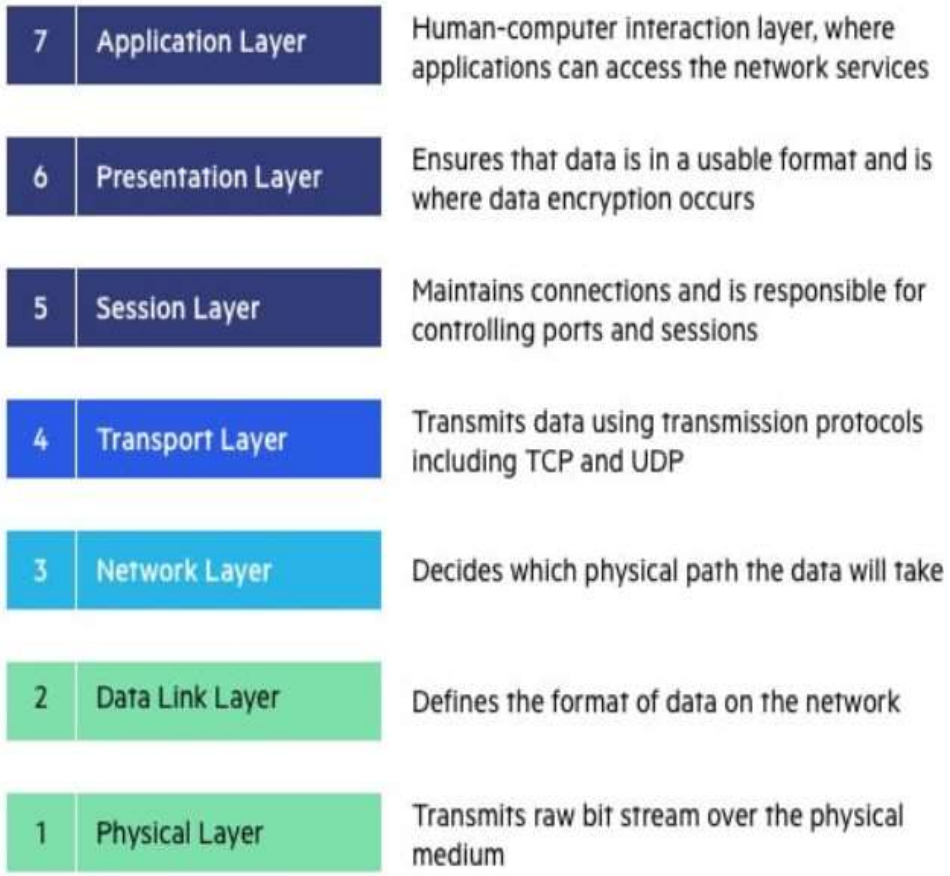
4. Application Layer

- The application layer of IoT architecture is the topmost layer that interacts directly with the end-user.
- It is responsible for providing user-friendly interfaces and functionalities that enable users to access and control IoT devices.
- This layer includes various software and applications such as mobile apps, web portals, and other user interfaces that are designed to interact with the underlying IoT infrastructure.
- It also includes middleware services that allow different IoT devices and systems to communicate and share data seamlessly.
- The application layer also includes analytics and processing capabilities that allow data to be analyzed and transformed into meaningful insights.
- This can include machine learning algorithms, data visualization tools, and other advanced analytics capabilities.

OSI Model

Accura Tequipment

- OSI stands for Open Systems Interconnection.
- The OSI model, created in 1984 by ISO, is a reference framework that explains the process of transmitting data between computers.
- The OSI Model can be seen as a universal language for computer networking.
- OSI consists of seven layers, and each layer performs a particular network function.
- OSI model divides the whole task into seven smaller and manageable tasks. Each layer is assigned a particular task.
- Each layer is self-contained, so that task assigned to each layer can be performed independently.



Physical Layer

- The main functionality of the physical layer is to transmit the individual bits from one node to another node.
- It is the lowest layer of the OSI model.
- It establishes, maintains and deactivates the physical connection.
- It specifies the mechanical, electrical and procedural network interface specifications.

Functions of the Physical Layer

- **Line Configuration:** It defines the way how two or more devices can be connected physically.
- **Data Transmission:** It defines the transmission mode whether it is simplex, half-duplex or full-duplex mode between the two devices on the network.
- **Topology:** It defines the way how network devices are arranged.
- **Signals:** It determines the type of the signal used for transmitting the information.
- **Bit synchronization:** The physical layer provides the synchronization of the bits by providing a clock. This clock controls both sender and receiver thus providing synchronization at the bit level.

Data Link Layer (DLL)

- The data link layer is responsible for the node-to-node delivery of the message.
- The main function of this layer is to make sure data transfer is error-free from one node to another, over the physical layer
- It defines the format of the data on the network.
- It provides a reliable and efficient communication between two or more devices.
- It is mainly responsible for the unique identification of each device that resides on a local network.
- It contains two sub-layers:
    1. **Logical Link Control Layer**
        - It is responsible for transferring the packets to the Network layer of the receiver that is receiving.
        - It identifies the address of the network layer protocol from the header.
        - It also provides flow control.
    2. **Media Access Control Layer**
        - A Media access control layer is a link between the Logical Link Control layer and the network's physical layer.
        - It is used for transferring the packets over the network.

Functions of the Data-link layer

- **Framing:** The data link layer translates the physical's raw bit stream into packets known as Frames. The Data link layer adds the header and trailer to the frame. The header which is added to the frame contains the hardware destination and source address.

- **Physical Addressing:** The Data link layer adds a header to the frame that contains a destination address. The frame is transmitted to the destination address mentioned in the header.
- **Flow Control:** Flow control is the main functionality of the Data-link layer. It is the technique through which the constant data rate is maintained on both the sides so that no data get corrupted. It ensures that the transmitting station such as a server with higher processing speed does not exceed the receiving station, with lower processing speed.
- **Error Control:** Error control is achieved by adding a calculated value CRC (Cyclic Redundancy Check) that is placed to the Data link layer's trailer which is added to the message frame before it is sent to the physical layer. If any error seems to occurr, then the receiver sends the acknowledgment for the retransmission of the corrupted frames.
- **Access Control:** When two or more devices are connected to the same communication channel, then the data link layer protocols are used to determine which device has control over the link at a given time.

Network Layer

- It is a layer 3 that manages device addressing, tracks the location of devices on the network.
- It determines the best path to move data from source to the destination based on the network conditions, the priority of service, and other factors.
- The Data link layer is responsible for routing and forwarding the packets.
- Routers are the layer 3 devices, they are specified in this layer and used to provide the routing services within an internetwork.
- The protocols used to route the network traffic are known as Network layer protocols. Examples of protocols are IP and Ipv6.

Functions of Network Layer:

- **Internetworking:** An internetworking is the main responsibility of the network layer. It provides a logical connection between different devices.
- **Addressing:** A Network layer adds the source and destination address to the header of the frame. Addressing is used to identify the device on the internet.
- **Routing:** Routing is the major component of the network layer, and it determines the best optimal path out of the multiple paths from source to the destination.
- **Packetizing:** A Network Layer receives the packets from the upper layer and converts them into packets. This process is known as Packetizing. It is achieved by internet protocol (IP).

Transport Layer

- The Transport layer is a Layer 4 ensures that messages are transmitted in the order in which they are sent and there is no duplication of data.
- The main responsibility of the transport layer is to transfer the data completely.
- It receives the data from the upper layer and converts them into smaller units known as segments.
- This layer can be termed as an end-to-end layer as it provides a point-to-point connection between source and destination to deliver the data reliably.

The two protocols used in this layer are:

1. **Transmission Control Protocol**
   - It is a standard protocol that allows the systems to communicate over the internet.
   - It establishes and maintains a connection between hosts.
   - When data is sent over the TCP connection, then the TCP protocol divides the data into smaller units known as segments. Each segment travels over the internet using multiple routes, and they arrive in different orders at the destination. The transmission control protocol reorders the packets in the correct order at the receiving end.
2. **User Datagram Protocol**
   - User Datagram Protocol is a transport layer protocol.
   - It is an unreliable transport protocol as in this case receiver does not send any acknowledgment when the packet is received, the sender does not wait for any acknowledgment. Therefore, this makes a protocol unreliable.

Functions of the Transport Layer

- **Service-point addressing:** Computers run several programs simultaneously due to this reason, the transmission of data from source to the destination not only from one computer to another computer but also from one process to another process. The transport layer adds the header that contains the address known as a service-point address or port address. The responsibility of the network layer is to transmit the data from one computer to another computer and the responsibility of the transport layer is to transmit the message to the correct process.
- **Segmentation and reassembly:** When the transport layer receives the message from the upper layer, it divides the message into multiple segments, and each segment is assigned with a sequence number that uniquely identifies each segment. When the message has arrived at the destination, then the transport layer reassembles the message based on their sequence numbers.
- **Connection control:** Transport layer provides two services Connection-oriented service and connectionless service. A connectionless service treats each segment as an individual packet, and they all travel in different routes to reach the destination. A connection-oriented service makes a connection with the transport layer at the destination machine before delivering the packets. In connection-oriented service, all the packets travel in the single route.
- **Flow control:** The transport layer also responsible for flow control but it is performed end-to-end rather than across a single link.
- **Error control:** The transport layer is also responsible for Error control. Error control is performed end-to-end rather than across the single link. The sender transport layer ensures that message reach at the destination without any error.

Session Layer

This layer is responsible for the establishment of connection, maintenance of sessions, and authentication, and also ensures security.

Functions of the Session Layer

- **Session establishment, maintenance, and termination:** The layer allows the two processes to establish, use and terminate a connection.
- **Synchronization:** This layer allows a process to add checkpoints that are considered synchronization points in the data. These synchronization points help to identify the error so that the data is re-synchronized properly, and ends of the messages are not cut prematurely and data loss is avoided.
- **Dialog Controller:** The session layer allows two systems to start communication with each other in half-duplex or full-duplex.

Presentation Layer

- A Presentation layer is mainly concerned with the syntax and semantics of the information exchanged between the two systems.
- It acts as a data translator for a network.
- This layer is a part of the operating system that converts the data from one presentation format to another format.
- The Presentation layer is also known as the syntax layer.

Functions of Presentation layer:

- **Translation:** The processes in two systems exchange the information in the form of character strings, numbers and so on. Different computers use different encoding methods, the presentation layer handles the interoperability between the different encoding methods. It converts the data from sender-dependent format into a common format and changes the common format into receiver-dependent format at the receiving end.
- **Encryption:** Encryption is needed to maintain privacy. Encryption is a process of converting the sender-transmitted information into another form and sends the resulting message over the network.
- **Compression:** Data compression is a process of compressing the data, i.e., it reduces the number of bits to be transmitted. Data compression is very important in multimedia such as text, audio, video.

Application Layer

- An application layer serves as a window for users and application processes to access network service.
- It handles issues such as network transparency, resource allocation, etc.
- An application layer is not an application, but it performs the application layer functions.
- This layer provides the network services to the end-users.

Functions of Application layer:

- **File transfer, access, and management (FTAM):** An application layer allows a user to access the files in a remote computer, to retrieve the files from a computer and to manage the files in a remote computer.
- **Mail services:** An application layer provides the facility for email forwarding and storage.
- **Directory services:** An application provides the distributed database sources and is used to provide that global information about various objects.

Example: Application – Browsers, Skype Messenger, etc.

OSI Model in a Nutshell

Layer No	Layer Name	Responsibility	Information Form(Data Unit)	Device or Protocol
7	Application Layer	Helps in identifying the client and synchronizing communication.	Message	SMTP
6	Presentation Layer	Data from the application layer is extracted and manipulated in the required format for transmission.	Message	JPEG, MPEG, GIF
5	Session Layer	Establishes Connection, Maintenance, Ensures Authentication, and Ensures security.	Message	Gateway
4	Transport Layer	Take Service from Network Layer and provide it to the Application Layer.	Segment	Firewall
3	Network Layer	Transmission of data from one host to another, located in different networks.	Packet	Router
2	Data Link Layer	Node to Node Delivery of Message.	Frame	Switch, Bridge
1	Physical Layer	Establishing Physical Connections between Devices.	Bits	Hub, Repeater, Modem, Cables

Gateways

Accura Tequipment

- IoT Gateways are intelligent central hubs for Internet of Things (IoT) devices.
- Gateway provides a bridge between different communication technologies which means we can say that a Gateway acts as a medium to open up connections between the cloud and controller (sensors/devices) in IoT. With the help of gateways, it is possible to establish device-to-device or device-to-cloud communication.
- A gateway can be a typical hardware device or software program.
- It enables a connection between the sensor network and the Internet along with enabling IoT communication, it also performs many other tasks such as this IoT gateway performs protocol translation, aggregating all data, local processing, and filtering of data before sending it to the cloud, locally storing data and autonomously controlling devices based on some inputted data, providing additional device security.

The Need For IoT Gateway

Accura Tequipment

- Organizations commonly deploy an array of different IoT devices, which can make it difficult to monitor and manage these devices. An IoT gateway is a centralized hub that connects IoT devices and sensors to cloud-based computing and data processing.

- Modern IoT gateways often allow bidirectional data flow between the cloud and IoT devices. This allows IoT sensor data to be uploaded for processing and commands to be sent from cloud-based applications to IoT devices.

Working of IoT Gateway

IoT gateways are designed to simplify and streamline IoT device communications and management. Some common functions of IoT devices include:

- **Inter-Device Communications:** If a company has deployed different types of devices from different vendors, they may not be able to communicate directly. An IoT gateway can act as a central hub and perform the necessary translations to allow inter-device communications.

- **Device-to-Cloud Communications:** IoT devices commonly send data to cloud-based infrastructure for processing and use in applications. An IoT gateway aggregates data from multiple devices, providing a single point of contact for cloud infrastructure.

- **Data Preprocessing:** IoT devices can collect a massive amount of data, which requires a significant amount of bandwidth to send to cloud-based applications for processing. An IoT gateway performs data filtering and pre-processing to reduce the volume of data sent to the cloud.

- **IoT Device Security:** IoT devices have notoriously poor security, making it inadvisable to make them accessible directly from the public Internet. IoT gateways sit between IoT devices and the Internet and can include integrated security functionality to help protect these devices against attack.

- **Intelligent Edge:** IoT gateways can natively understand and process the data produced by IoT devices. This may allow them to support basic device management directly.

Key functionalities of IoT Gateway

- Establishing communication bridge
- Provides additional security.
- Performs data aggregation.
- Pre processing and filtering of data.
- Provides local storage as a cache/ buffer.
- Data computing at edge level.
- Ability to manage entire device.
- Device diagnostics.
- Adding more functional capability.
- Verifying protocols.

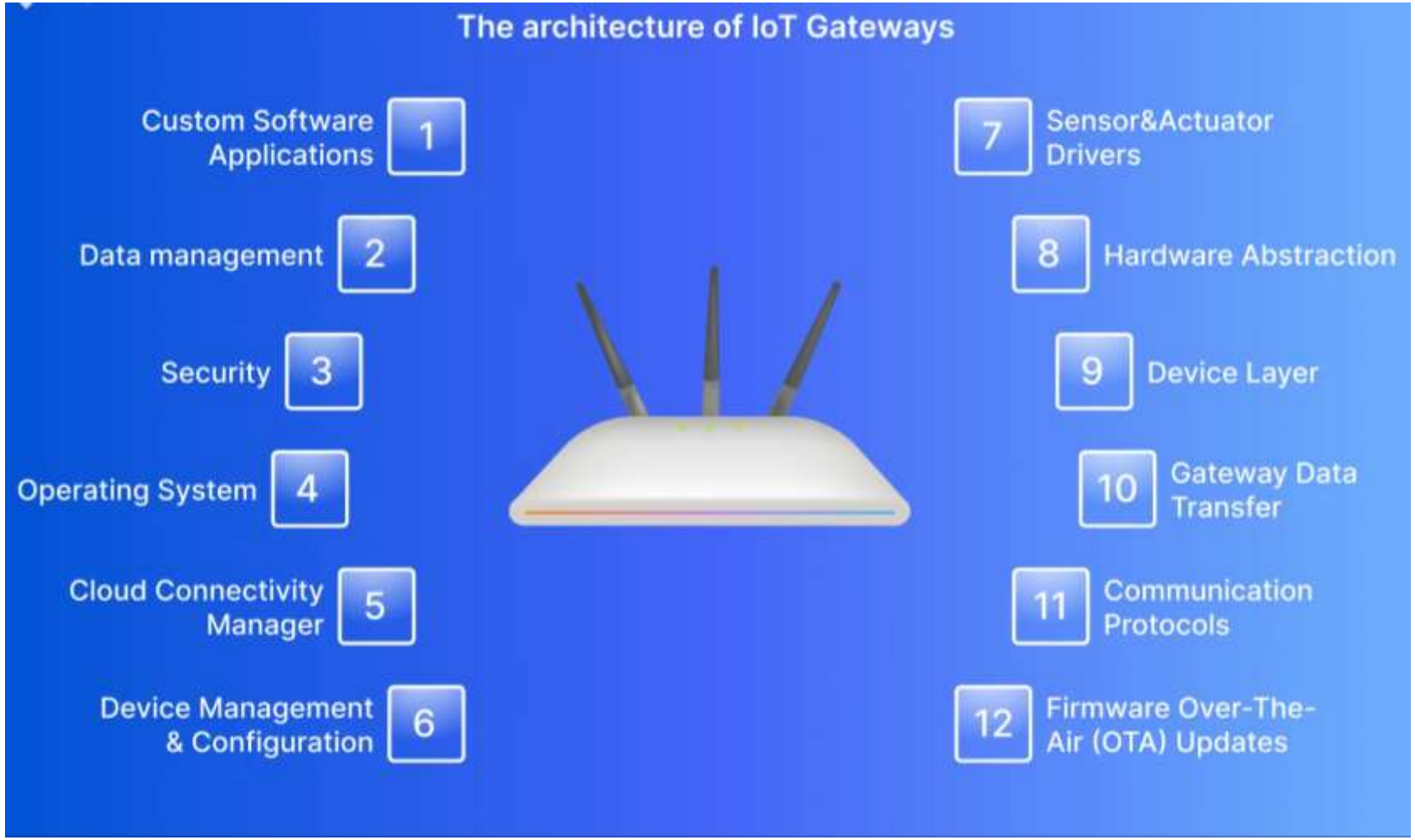
Advantages of Gateway

There are several advantages of using a gateway in the Internet of Things (IoT), including:

- **Protocol translation:** IoT devices typically use different communication protocols, and a gateway can translate between these protocols to enable communication between different types of devices.
- **Data aggregation:** A gateway can collect data from multiple IoT devices and aggregate it into a single stream for easier analysis and management.
- **Edge computing:** Gateways can perform edge computing tasks such as data processing, analytics, and machine learning, enabling faster and more efficient decision-making.
- **Security**: Gateways can act as a secure access point for IoT devices, providing a layer of protection against cyber threats.
- **Scalability:** Gateways can support a large number of IoT devices and can be easily scaled up or down to meet changing needs.
- **Improved reliability:** Gateways can help to improve the reliability of IoT devices by managing network connectivity and providing a backup mechanism in case of network failure.
- **Cost-effective**: Gateways can be a cost-effective way to manage and control a large number of IoT devices, reducing the need for expensive infrastructure and IT resources.

IoT Gateway Architecture

Accura Tequipment



- **Security layer.** It's one of the most crucial components of an IoT cloud gateway. Security features usually include encryption for data protection during transmission and storage, user authentication to prevent unauthorized access, and secure boot to ensure the gateway cannot operate with illegal firmware.

- **Device layer.** It's responsible for device-to-gateway communication and comprises IoT sensors, networking modules, circuits, and a microcontroller/processor.

- **Data management layer.** It handles IoT data collection, storage, filtering, and streaming. It ensures that data flows smoothly and securely between the sensor nodes, the gateway, and the cloud without any loss or alteration.

- **Operating system.** It's a foundation for managing hardware resources and running other gateway applications and components. Common OSs are a real-time operating system, or RTOS, for simple use cases and Linux for handling complex tasks.

Accura Tequipment

- **Hardware abstraction layer.** It allows developers to write code that interacts with hardware components without needing to know all the hardware details.

- **Gateway data transfer layer.** It transmits data from the gateway to the cloud, connecting to the internet via Ethernet, WiFi, or cellular networks.

- **Communication protocols.** It's a component that defines the rules for data exchange. Some standard protocols include MQTT, HTTP, and CoAP.

- **Cloud connectivity manager.** It establishes and handles secure connections to cloud platforms, covering device status monitoring, authentication, and reconnection.

Accura Tequipment

- **Sensor and actuator drivers.** It's a layer that allows the IoT custom gateway to communicate with sensors and modules through software interfaces.

- **Custom software apps.** These are tailored to the specific needs of your IoT ecosystem and organization. For example, they can handle data analysis and process automation.

- **Firmware OTA updates.** It's a critical component that ensures the IoT device is up to date. Over-the-air updates protect device memory and power and introduce functionality improvements, security patches, and bug fixes.

- **Device management and configuration.** It's a layer that helps the gateway monitor all connected IoT devices and change settings whenever necessary.

Communication Protocols in IoT Gateways

- In the context of IoT (Internet of Things) gateways, communication protocols play a crucial role in facilitating the exchange of data between IoT devices and the cloud or other backend systems. IoT gateways act as intermediaries between the devices in the field and the central data processing and storage infrastructure.
- Here are some key communication protocols commonly used in IoT gateways:

  - **MQTT (Message Queuing Telemetry Transport):** MQTT is a lightweight and publish-subscribe messaging protocol designed for low-bandwidth, high-latency, or unreliable networks. It is widely used in IoT scenarios where devices need to send and receive small messages with minimal overhead. MQTT is commonly used for communication between IoT devices and gateways, as well as between gateways and cloud platforms. Its lightweight nature makes it suitable for resource-constrained devices.

  - **CoAP (Constrained Application Protocol):** CoAP is a specialized web transfer protocol for constrained devices and networks. It is designed to be simple, lightweight, and suitable for IoT applications. CoAP is built on top of the UDP protocol. CoAP is often used in scenarios where HTTP is too heavy for IoT devices. It is suitable for resource-constrained environments and can be used in both device-to-gateway and gateway-to-cloud communications.

  - **HTTP/HTTPS (Hypertext Transfer Protocol/Secure):** HTTP is a well-established protocol used for data communication on the World Wide Web. In IoT, HTTPS (HTTP over TLS/SSL) is employed for secure communication. HTTP/HTTPS is commonly used in scenarios where secure communication is a priority. It is often used for communication between IoT gateways and cloud services.

- **AMQP (Advanced Message Queuing Protocol):** AMQP is an open standard application layer protocol for message-oriented middleware. It enables the communication between systems supporting different programming languages. AMQP is used in scenarios where a reliable and efficient message queuing system is required. It allows for the exchange of messages between IoT devices and backend systems.

- **DDS (Data Distribution Service):** DDS is a middleware protocol and API standard for data-centric communication, particularly in real-time and embedded systems. DDS can be used for high-performance, real-time communication in IoT gateways where low-latency and reliable data distribution are critical.

- **Modbus:** Modbus is a serial communication protocol widely used in industrial automation for connecting electronic devices. Modbus is often used in industrial IoT (IIoT) settings where legacy equipment communicates with modern IoT systems through gateways.

- **Advanced Message Queuing Protocol (AMQP).** It's a secure messaging protocol. AMQP is best for scenarios where message queuing and reliability are crucial, like industrial IoT applications.

- **WebSocket.** It enables bidirectional communication between IoT devices and the gateway over a single TCP connection.

When designing IoT solutions, it's common to use a combination of these protocols based on the specific requirements of the application, the nature of the devices involved, and the characteristics of the network. The choice of protocol depends on factors such as data size, latency requirements, power consumption, and security considerations

IoT Protocols

Accura Tequipment

- An IoT (Internet of Things) protocol refers to a set of rules and standards that enable devices and systems to communicate and exchange data within an IoT network.
- These protocols define the methods and formats for data transmission, device discovery, connectivity, and security in an IoT environment.
- IoT protocols are designed to accommodate the unique requirements of IoT devices, which often have limited resources such as processing power, memory, and energy. They aim to provide efficient and reliable communication between devices, while minimizing bandwidth usage and power consumption.
- Protocols aren't all created equal. Not all protocols work, or work well, in every circumstance, some protocols work well for IoT use in buildings, some are well suited for IoT deployments spread among buildings and others work well for national or global IoT use cases.

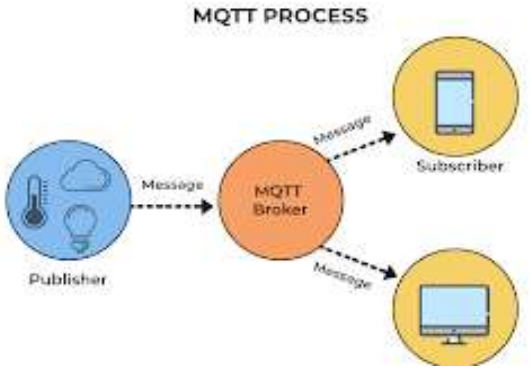
MQTT

Accura Tequipment

- MQTT is a lightweight publish-subscribe protocol designed for IoT devices with limited resources such as low bandwidth, memory, and processing power. It works on top of the TCP/IP protocol and uses a simple and efficient messaging model to enable bi-directional communication between devices.
- MQTT is widely used in IoT applications such as smart homes, industrial automation, and asset tracking.

Advantages:

- Low overhead and efficient use of network bandwidth
- Can handle unreliable and intermittent network connections
- Supports QoS (Quality of Service) levels for message delivery assurance
- Scalable and easy to implement



Disadvantages:

- Limited security features, which may require additional encryption and authentication mechanisms
- May not be suitable for applications that require real-time communication

CoAP (Constrained Application Protocol)

Accura Tequipment

- CoAP is a lightweight protocol designed for IoT devices with limited resources and low-power networks such as 6LoWPAN and Zigbee. It's similar to HTTP in terms of its request/response model, but it's optimized for constrained devices and networks.
- CoAP is widely used in IoT applications such as smart homes, healthcare monitoring, and industrial automation.

Advantages:

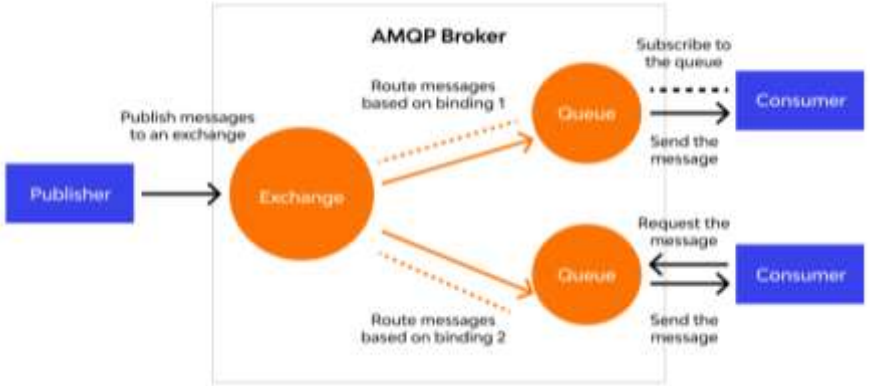
- Low overhead and efficient use of network bandwidth
- Can handle unreliable and intermittent network connections
- Supports resource discovery and manipulation
- Scalable and easy to implement

Disadvantages:

- Limited security features, which may require additional encryption and authentication mechanisms
- May not be suitable for applications that require real-time communication

AMQP

- AMQP refers to Advanced Message Queuing Protocol.
- Globally recognized standard that works on the application layer essentially, it is mainly used for developing unmatched communication operability between client and broker parties.
- The publisher bears the responsibility of message generation while clients collect and administer them.



- The role of brokers, in this whole process is to make a sure message, part of the exchange, goes directly from the publisher to the client.
- Speaking of its key functionality, routing, message orientation, and queuing are the top ones. Use of AMQP makes interoperability an achieved goal, featuring different configurations and infrastructure. It allows developers to bring all the protocol-approved compliant client library and broker offered into action.
- AMQP is a more complex protocol that is designed for high-reliability applications. It is often used for financial trading and other mission-critical applications.

HTTP (Hypertext Transfer Protocol)

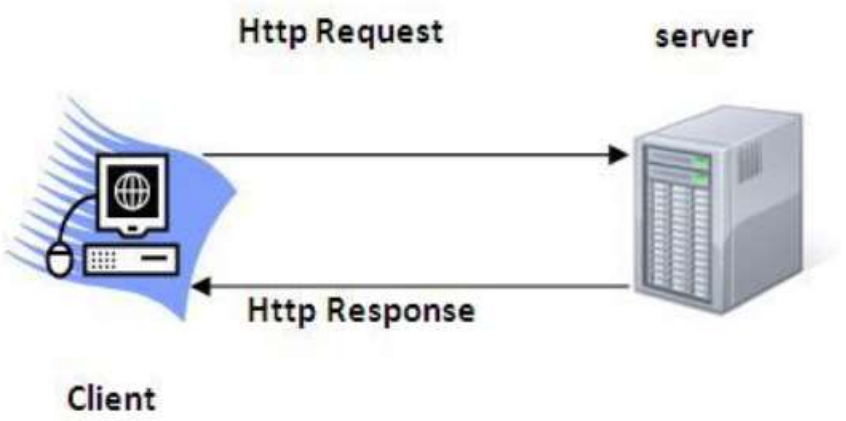
- HTTP is a widely used protocol for web communication and is also used in IoT applications such as smart homes and smart cities. It uses a request/response model and supports various data formats such as JSON and XML.
- HTTP is suitable for IoT applications that require real-time communication and high-speed data transfer.
- The HTTP protocol is also a stateless protocol meaning that the server isn't required to store session information, and each request is independent of the other

Advantages:

- Widely used and supported
- Supports various data formats
- Suitable for real-time communication and high-speed data transfer
- Robust security features

Disadvantages:

- High overhead and inefficient use of network bandwidth
- Not suitable for low-power and resource-constrained devices
- Requires a reliable and stable network connection



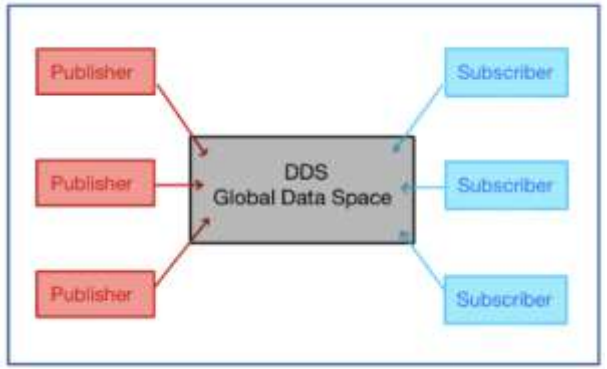
DDS (Data Distribution Service)

Accura Tequipment

- DDS is a protocol designed for real-time data distribution and messaging in mission-critical systems such as industrial automation, aerospace, and defense. It provides a publish-subscribe model with support for Quality of Service (QoS) levels and data-centric middleware.
- DDS is suitable for applications that require high-speed data transfer and real-time communication.

Advantages:

- High scalability and support for real-time data distribution
- Reliable and secure communication with built-in authentication and encryption
- Supports Quality of Service (QoS) levels for message delivery assurance



Disadvantages:

- May not be suitable for resource-constrained devices due to its complexity and overhead
- Requires specialized expertise for implementation and maintenance

Zigbee

Accura Tequipment

- Zigbee is a wireless communication protocol designed for low-power and low-data-rate IoT applications such as smart homes, smart lighting, and home automation. It operates on the IEEE 802.15.4 standard and provides a mesh networking topology for reliable communication in a decentralized network.

Advantages:

- Low-power consumption for extended battery life
- Reliable communication with mesh networking topology
- Secure communication with built-in encryption and authentication
- Suitable for low-data-rate applications

Disadvantages:

- Limited range compared to other wireless protocols such as Wi-Fi and Bluetooth
- Limited bandwidth and data rate
- Requires a gateway for internet connectivity

LoRaWAN (Long Range Wide Area Network)

Accura Tequipment

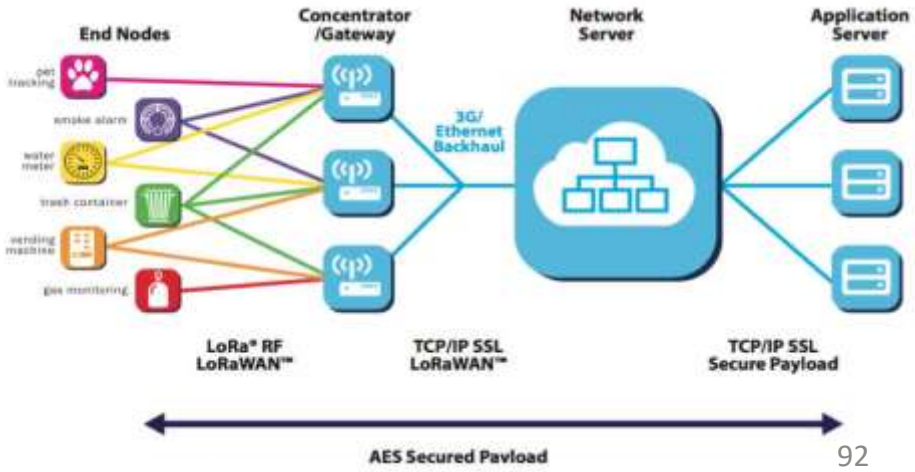
- LoRaWAN is a low-power, long-range wireless communication protocol designed for IoT applications such as smart cities, asset tracking, and agriculture. It operates on the unlicensed radio spectrum and provides a wide area network (WAN) for communication over long distances.

Advantages:

- Long-range communication up to several kilometers
- Low-power consumption for extended battery life
- Secure communication with built-in encryption and authentication
- Suitable for low-data-rate applications

Disadvantages:

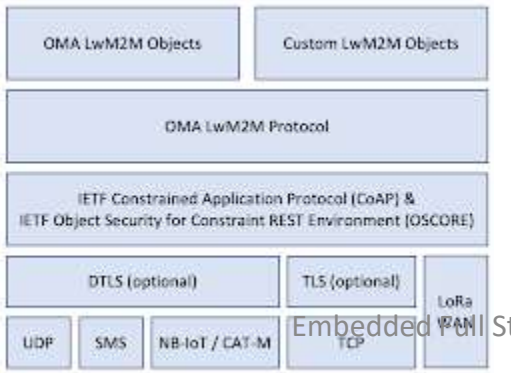
- Limited bandwidth and data rate
- Requires a gateway for internet connectivity
- Limited network coverage in some regions



LWM2M

Accura Tequipment

- LWM2M stands for Lightweight Machine-2-Machine. is a standard for machine-to-machine (M2M) communication.
- It defines a way for devices to connect to the internet and exchange data.
- LWM2M is a very flexible protocol, and it can be used for a wide variety of IoT applications.
- LWM2M defines a way for devices to connect to the internet and exchange data.



Bluetooth

Accura Tequipment

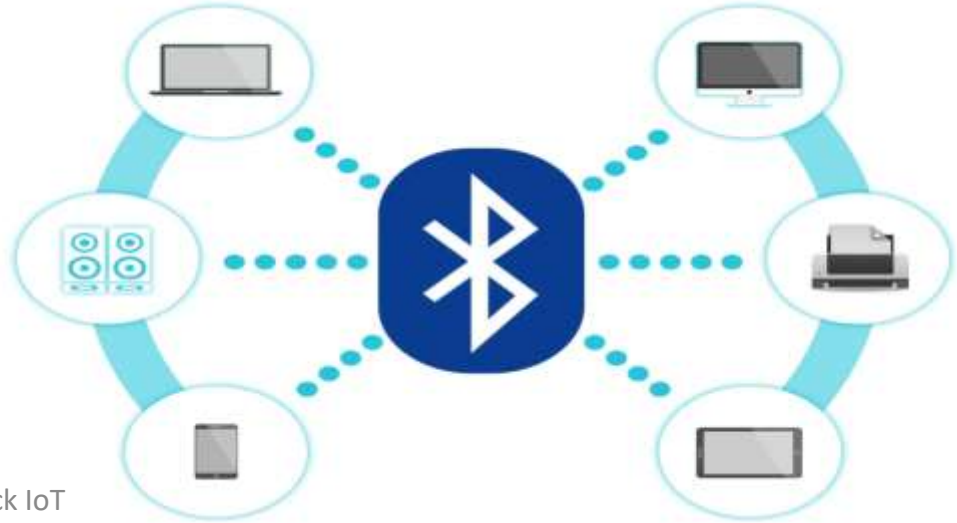
- Bluetooth is a short-range wireless technology for voice and data communication. Invented by Ericsson in 1994.
- Operates in the unlicensed ISM bands from 2.4 GHz to 2.485 GHz.
- Devices can form connections within a range of up to 10 meters.
- Bluetooth enables seamless connectivity between IoT devices within short distances, facilitating efficient data exchange.
- It can be used in Wireless headsets, Wireless PANs and LANs, Data transfer between devices, Medical healthcare, sports, fitness, military.
- The various types of Bluetooth devices are used In-Car Headset, Stereo Headset, Webcam, Bluetooth-equipped Printer, Bluetooth GPS

Advantages:

- Low-cost and easy-to-use.
- Ad-hoc connection without wires.
- Used for voice and data transfer.

Disadvantages:

- Less secure, susceptible to hacking.
- Slow data transfer rate of 3 Mbps.
- No support for routing.



Bluetooth® Low Energy (LE)

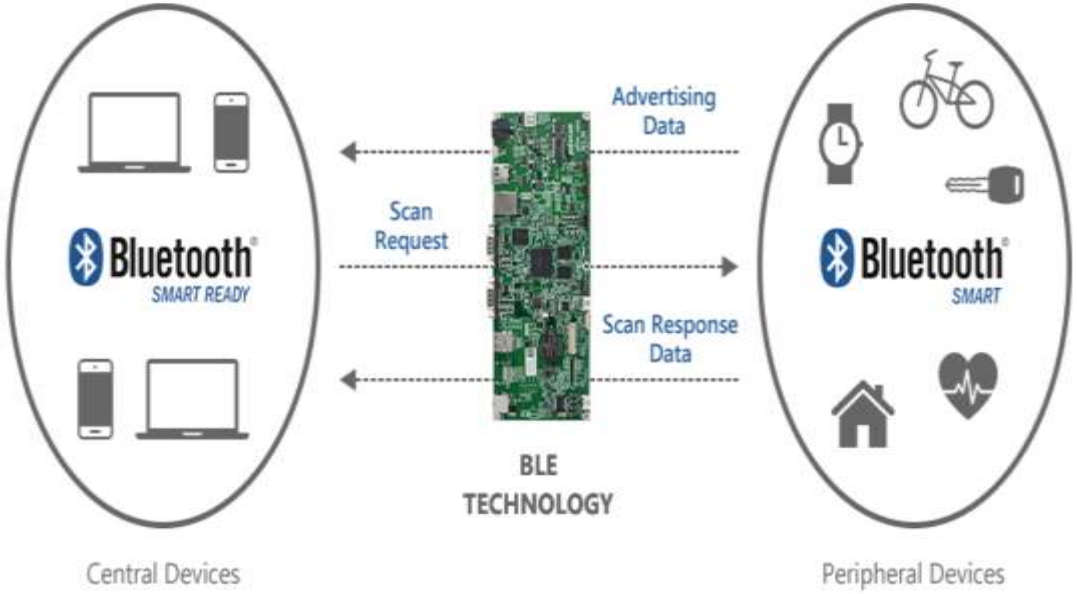
Accura Tequipment

- The Bluetooth Low Energy (LE) radio is designed for very low power operation.
- Transmitting data over 40 channels in the 2.4GHz unlicensed ISM frequency band, the Bluetooth LE radio provides developers a tremendous amount of flexibility to build products that meet the unique connectivity requirements of their market.
- Bluetooth LE supports multiple communication topologies, expanding from point-to-point to broadcast and, most recently, mesh, enabling Bluetooth technology to support the creation of reliable, large-scale device networks.
- Initially it is known for its device communications capabilities, Bluetooth LE is now also widely used as a device positioning technology to address the increasing demand for high accuracy indoor location services.
- Bluetooth LE now includes features that enable one device to determine the presence, distance, and direction of another device.

BLE Communication

Accura Tequipment

- BLE communication consists of two main components: Advertising and Connecting.
- BLE Advertising is a one-way communication method. Devices that want to be discovered transmit packets of data in set intervals. On the other hand, there are listening devices (smart phones) scanning other devices, receiving and processing advertising packets and filtering them.
- The device which advertises is called a Bluetooth Peripheral whereas the one doing the scanning is a Bluetooth Central device.
- After a device is discovered, a connection is established. It is now possible to read the services that a Bluetooth Low Energy device offers, and for each service its characteristics (Implementation of a GATT profile) Each characteristic provides some value, which can be read, written, or both.



Cellular

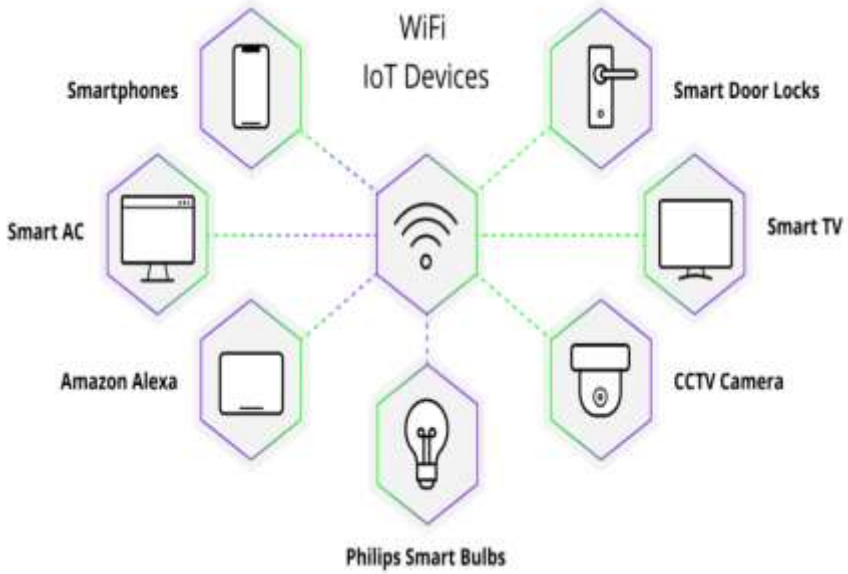
- Cellular IoT refers to the technology that connects physical objects to the Internet using existing cellular networks, which are primarily utilized by smartphones.
- This technology leverages the infrastructure of current cellular networks, eliminating the need to establish separate dedicated networks for IoT devices.
- By utilizing existing cellular networks, Cellular IoT reduces the need for additional investment in developing separate infrastructure for IoT connectivity. This leads to cost savings and quicker deployment of IoT solutions.
- Cellular IoT offers wide coverage and functional connectivity in most countries.
- Cellular IoT can utilize different cellular standards such as 3G, 4G/LTE, or 5G, depending on the specific requirements of the IoT application.
- Cellular IoT enables the connection of various IoT devices across diverse sectors, including but not limited to smart cities (e.g., streetlights), agriculture (e.g., irrigation systems), and healthcare (e.g., medical equipment monitoring).

Benefits of Cellular IoT

- **Extensive Coverage:** Utilizes existing cellular networks, ensuring wide coverage for IoT devices across cities or regions.
- **Cost-saving:** Leverages pre-existing cellular infrastructure, minimizing the need for additional investment in IoT connectivity.
- **Remote Management:** Facilitates centralized management and troubleshooting of IoT devices from anywhere via online platforms, reducing downtime and operational costs.
- **Flexibility in Connectivity:** Offers a range of cellular IoT technologies to meet diverse device and application needs, ensuring scalability and adaptability.
- **Security:** Adheres to GSMA standards and supports additional encryption layers, providing robust security measures for cellular IoT devices and data.

WiFi

- Wi-Fi is widely adopted across homes, commercial spaces, and industrial settings, making it easily accessible for IoT implementations in these environments.
- Wi-Fi offers high-speed data transfer, facilitating the efficient exchange of large amounts of data, which is crucial for many IoT applications.
- Wi-Fi is well-suited for Local Area Network (LAN) environments, providing reliable connectivity over short to medium-range distances, ideal for IoT devices within a confined area.
- With various standards like 802.11n, Wi-Fi provides flexibility for technologists to choose the most suitable standard for their deployment needs, ensuring compatibility and performance.
- One drawback is that many Wi-Fi standards consume significant power, making them unsuitable for low-power or battery-powered IoT devices, restricting their use in certain deployments where power efficiency is critical.
- Wi-Fi's range limitations and scalability challenges can hinder its feasibility for IoT deployments that require long-range connectivity or involve a large number of devices spread across a wide area.
- Compared to other wireless technologies like cellular, for instance, WiFi transmits data at much higher frequencies – 2.4 GHz or 5 GHz.



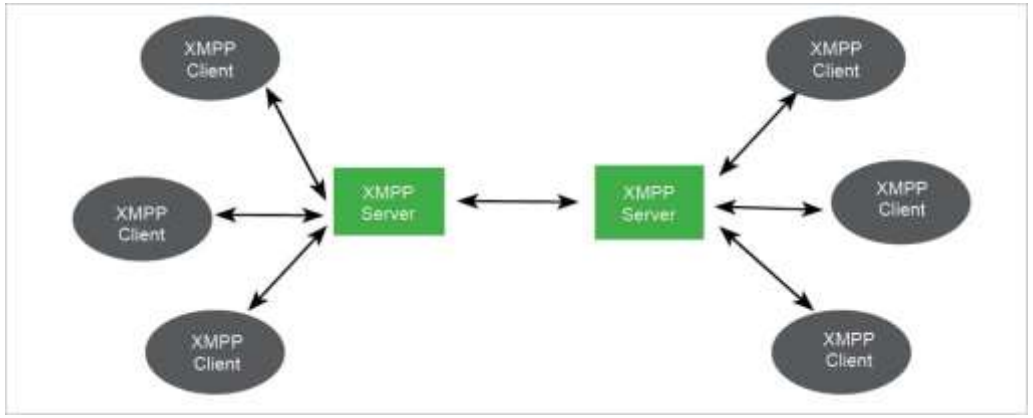
XMPP

- XMPP (Extensible Messaging and Presence Protocol) was initially developed by the Jabber open-source community in the early 2000s.
- XMPP was originally created for real-time human-to-human communication, providing a standardized way for messaging and presence information exchange. Over time, XMPP evolved to support Machine-to-Machine (M2M) communication, becoming a lightweight middleware solution.
- XMPP is utilized for routing XML data efficiently, enabling structured but extensible data exchange between multiple entities on a network.
- XMPP finds significant application in consumer-oriented Internet of Things (IoT) deployments, particularly with smart appliances and similar connected devices.
- XMPP is an open-source protocol, fostering collaboration and innovation in its development and implementation.
- The XMPP Standards Foundation provides support and guidance for the continued development and standardization of the XMPP protocol.
- XMPP's architecture allows for scalability and flexibility, making it suitable for a wide range of applications beyond its original messaging and presence functionalities.
- XMPP promotes interoperability, enabling different systems and devices to communicate seamlessly, which is crucial for the diverse ecosystem of IoT devices and applications.
- Despite being developed over two decades ago, XMPP remains relevant and widely used in various communication and IoT scenarios due to its robustness and adaptability.

Communication between a client and server in XAMP Protocol

Accura Tequipment

- It uses Port 5222 for the client to server (C2S) communication.
- It utilizes Port 5269 for server to server (S2S) communication.
- Discovery and XML streams are used for S2S and C2S communication.
- It uses security mechanisms such as TLS (Transport Layer Security) and SASL (Simple Authentication and Security Layer).
- There are no in-between servers for federation, unlike e-mail.



Z-Wave

- Z-Wave is a wireless mesh network communication protocol built on low-power radio frequency technology. Like Bluetooth and Wi-Fi, Z-Wave lets smart devices communicate with encryption, thereby providing a level of security to the IoT deployment.
- Z-Wave it is a wireless communication protocol used by automatic or automotive appliances for the purpose of connection and communication.
- It is invented in 1999 by Zensys a Danish-American company.
- It was designed to provide the **reliable, low-latency transmission** of small data packets using low-energy radio waves at data rates up to 100kbit/s with a throughput of up to 40kbit/s (9.6kbit/s using old chips) and are suitable for control and sensor applications.

Characteristics of Z-Wave :

- Uses RF for signaling and control
- Frequency : 900 MHz (ISM)
- Range : 30 meter
- Data rates : upto 100 kbps
- FSK Modulation

Applications of Z-Wave :

- Home automation
- Water management using flood sensors
- Fingerprint scanner

IoT Communication models

- IoT devices are found everywhere and will enable circulatory intelligence in the future.
- For operational perception, it is important and useful to understand how various IoT devices communicate with each other.
- Communication models used in IoT have great value.
- The IoTs allow people and things to be connected any time, any space, with anything and anyone, using any network and any service.

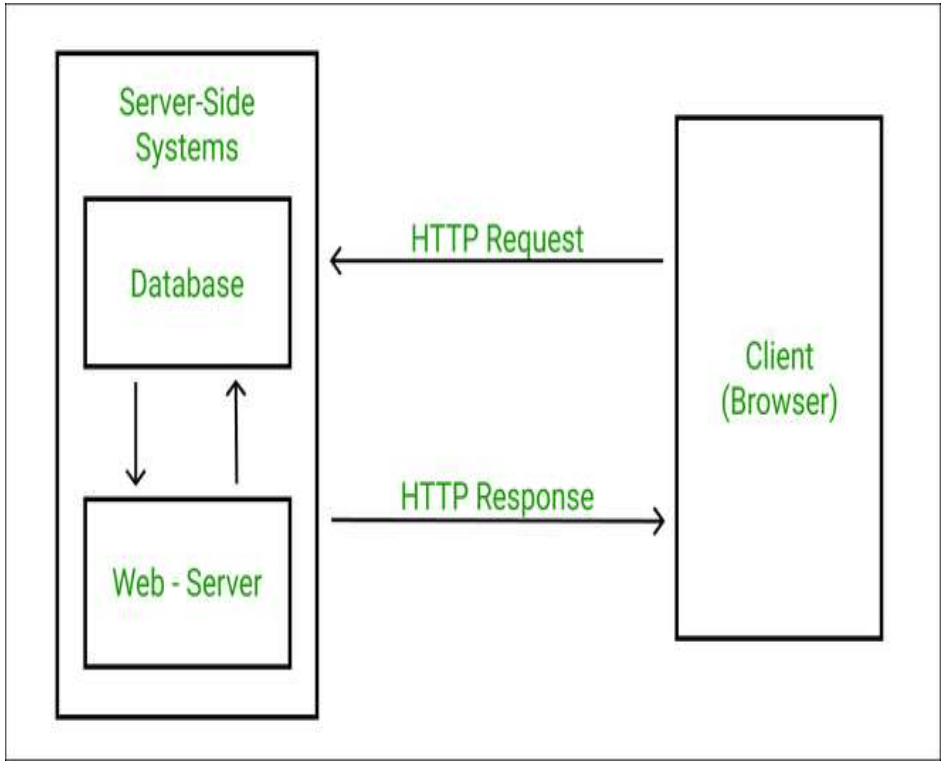
Types of Communication Model are

1. **Request-Response Model**
2. **Publish-Subscribe Model**
3. **Push-Pull Model**
4. **Exclusive Pair Model**

Request-Response Model

Accura Tequipment

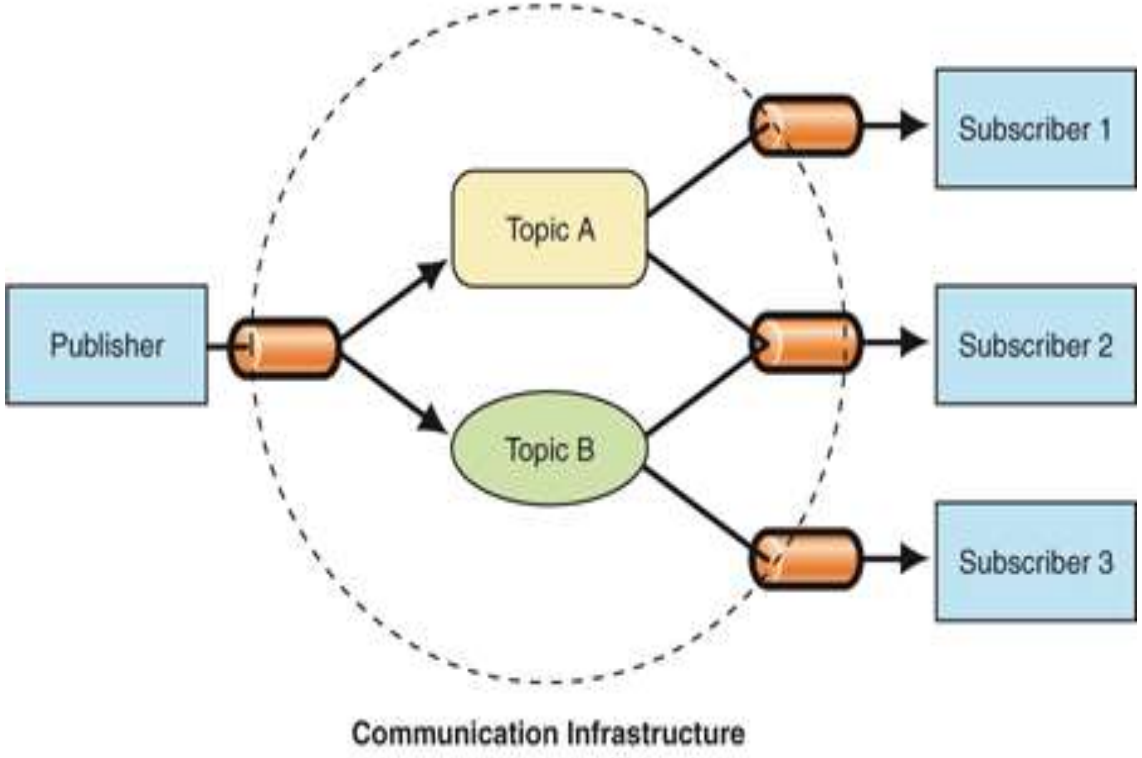
- Request-response model is communication model in which the client sends requests to the server and the server responds to the requests. When the server receives a request, it decides how to respond, fetches the data, retrieves resource representation, prepares the response, and then sends the response to the client.
- Request-response is a stateless communication model and each request-response pair is independent of others.
- HTTP works as a request-response protocol between a client and server. A web browser may be the client, and an application on a computer that hosts a web site may be the server.
- Example: A client (browser) submits an HTTP request to the server; then the server returns a response to the client. The response contains status information about the request and may also contain the requested content



Publish-Subscribe Model

Accura Tequipment

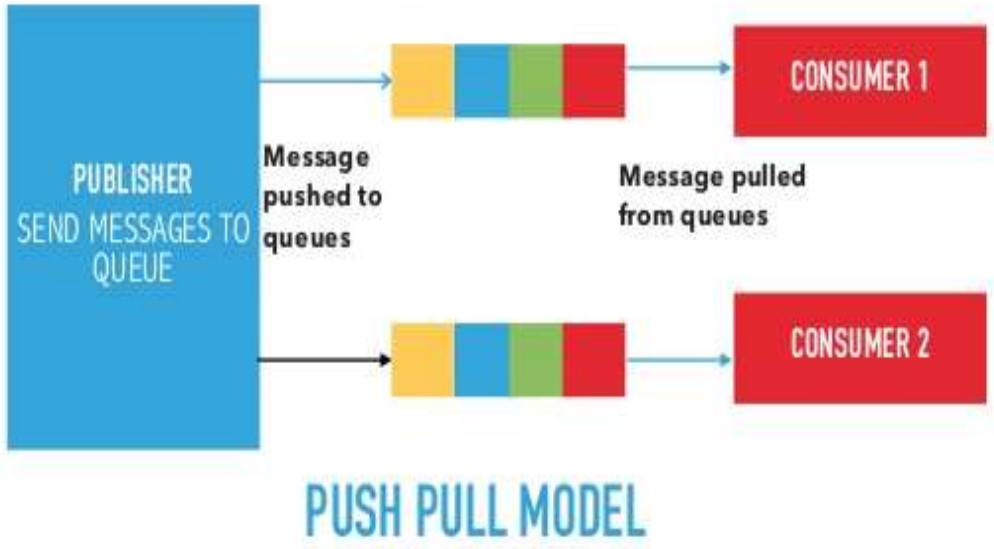
- Publish-Subscribe is a communication model that involves publishers, brokers and consumers.
- Publishers are the source of data. Publishers send the data to the topics which are managed by the broker.
- Publishers are not aware of the consumers.
- Consumers subscribe to the topics which are managed by the broker.
- When the broker receive data for a topic from the publisher, it sends the data to all the subscribed consumers.



Push-Pull Model

Accura Tequipment

- Push-Pull is a communication model in which the data producers push the data to queues and the consumers Pull the data from the Queues.
- Producers do not need to be aware of the consumers.
- Queues help in decoupling the messaging between the Producers and Consumers.
- Queues also act as a buffer which helps in situations when there is a mismatch between the rate at which the producers push data and the rate at which the consumer pull data.



Exclusive Pair Model

Accura Tequipment

- Exclusive Pair is a bidirectional, fully duplex communication model that uses a persistent connection between the client and server.
- Connection is setup it remains open until the client sends a request to close the connection.
- Client and server can send messages to each other after connection setup.
- Exclusive pair is stateful communication model and the server is aware of all the open connections.

Application Programming Interface

API stands for **Application Programming Interface**. In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications. This contract defines how the two communicate with each other using requests and responses.

OR

APIs are mechanisms that enable two software components to communicate with each other using a set of definitions and protocols.

For example, the weather bureau's software system contains daily weather data. The weather app on your phone "talks" to this system via APIs and shows you daily weather updates on your phone.

Working of API

- API architecture is usually explained in terms of client and server.
- The application sending the request is called the client, and the application sending the response is called the server.
- In the weather example, the bureau's weather database is the server, and the mobile app is the client
- There are four different ways that APIs can work depending on when and why they were created.
  - SOAP APIs
  - RPC APIs
  - WebSocket APIs
  - REST APIs

SOAP APIs

- These APIs use Simple Object Access Protocol.
- Client and server exchange messages using XML.
- This is a less flexible API that was more popular in the past.

RPC APIs

- These APIs are called Remote Procedure Calls.
- The client completes a function (or procedure) on the server, and the server sends the output back to the client.

Accura Tequipment

WebSocket APIs

- WebSocket API is another modern web API development that uses JSON objects to pass data.
- WebSocket API supports two-way communication between client apps and the server. The server can send callback messages to connected clients, making it more efficient than REST API.

REST APIs

- These are the most popular and flexible APIs found on the web today.
- The client sends requests to the server as data.
- The server uses this client input to start internal functions and returns output data back to the client.

Different Types of API

APIs are classified both according to their architecture and scope of use.

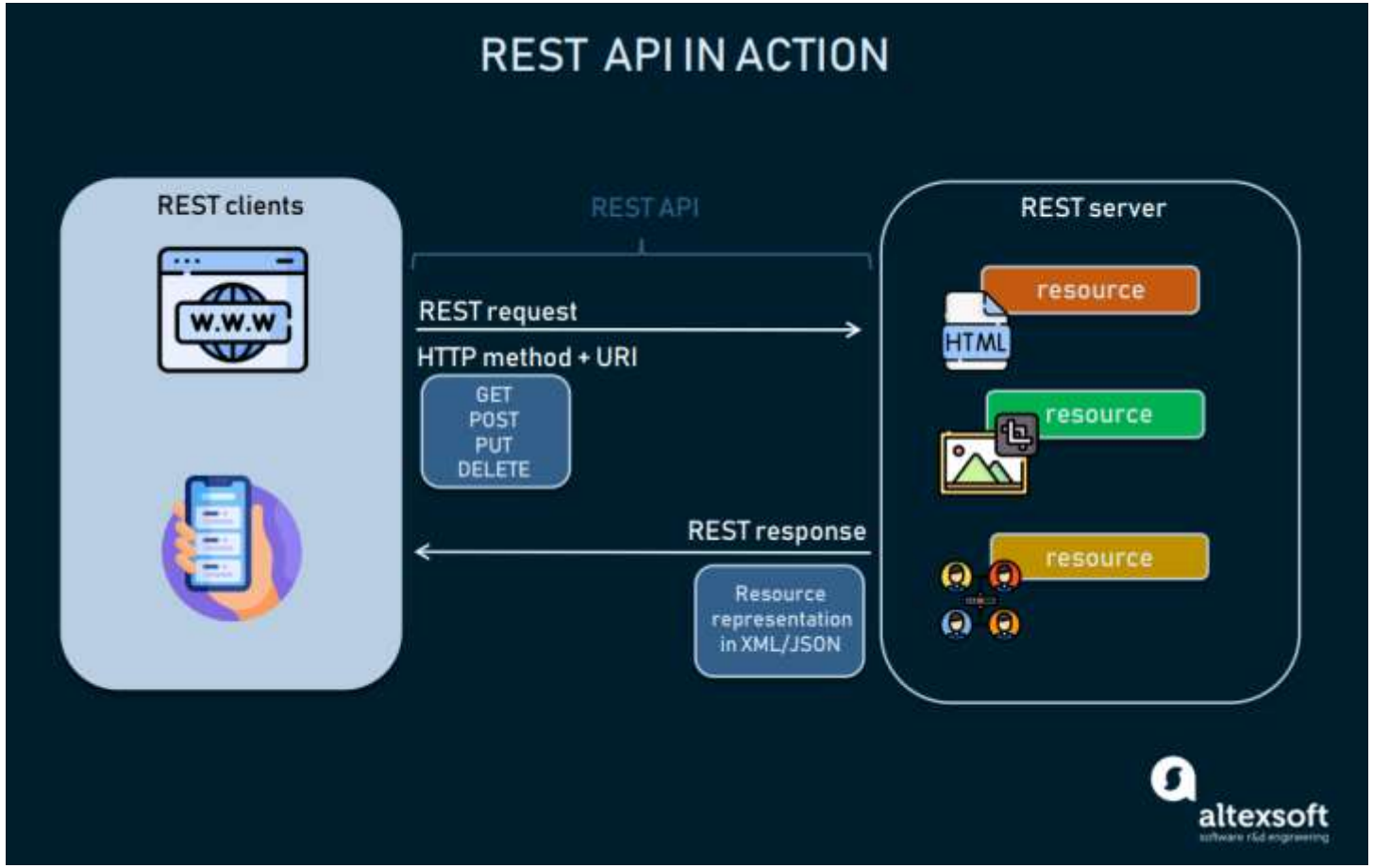
- **Private APIs:** These are internal to an enterprise and only used for connecting systems and data within the business.
- **Public APIs:** These are open to the public and may be used by anyone. There may or not be some authorization and cost associated with these types of APIs.
- **Partner APIs:** These are only accessible by authorized external developers to aid business-to-business partnerships.
- **Composite APIs:** These combine two or more different APIs to address complex system requirements or behaviors.

REST API

- REST stands for **Representational State Transfer**.
- REST API is a way of accessing web services in a simple and flexible way without having any processing.
- REST defines a set of functions like GET, PUT, DELETE, etc. that clients can use to access server data.
- Clients and servers exchange data using HTTP.
- The main feature of REST API is statelessness. Statelessness means that servers do not save client data between requests.
- REST technology uses less bandwidth, simple and flexible making it more suitable for internet usage. It’s used to fetch or give some information from a web service. All communication done via REST API uses only HTTP request.

Working of REST API

Accura Tequipment



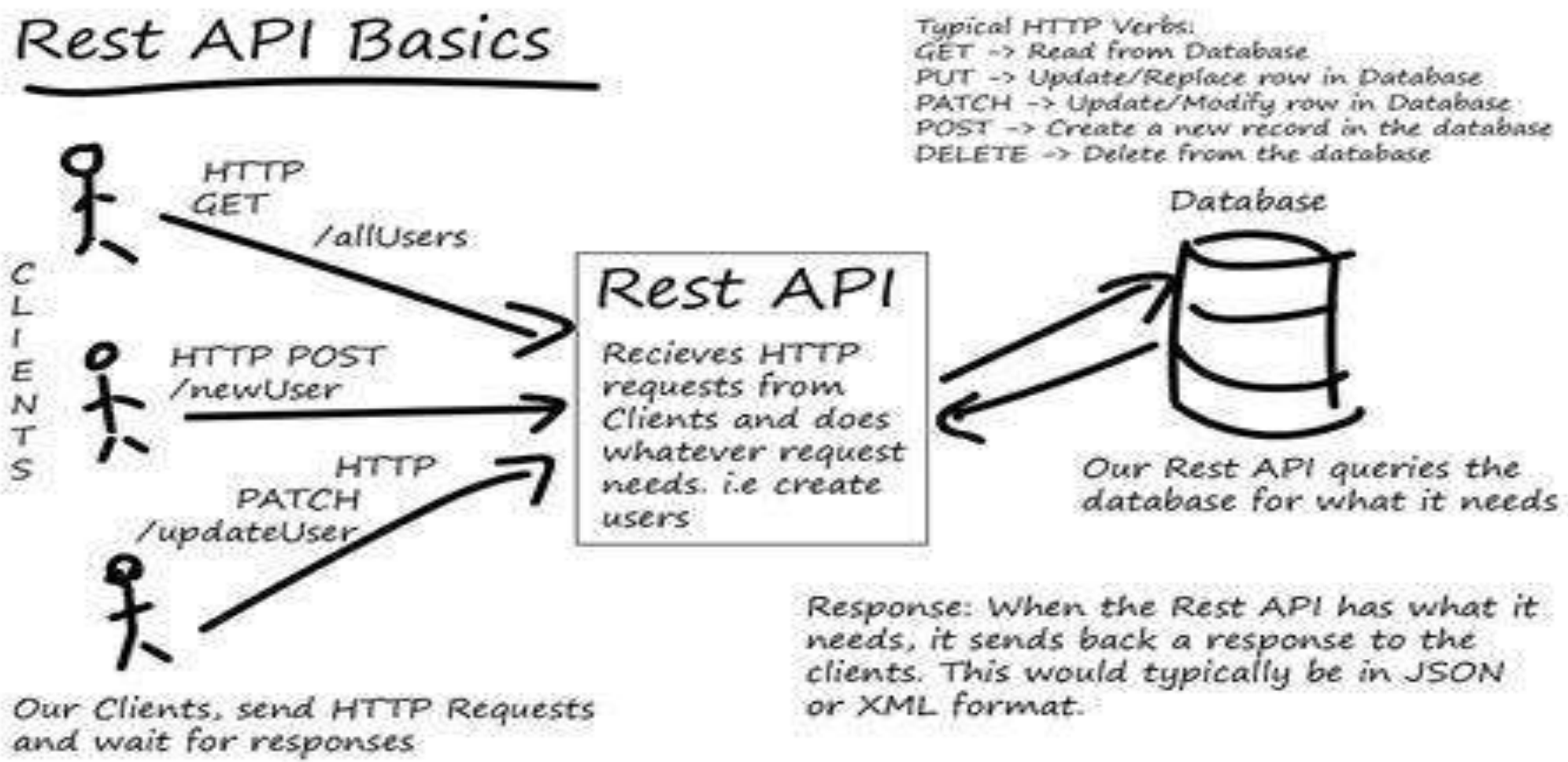
- A request is sent from client to server in the form of a web URL as HTTP GET or POST or PUT or DELETE request. After that, a response comes back from the server in the form of a resource which can be anything like HTML, XML, Image, or JSON. But now JSON is the most popular format being used in Web Services.

- In HTTP there are five methods that are commonly used in a REST-based Architecture i.e., POST, GET, PUT, PATCH, and DELETE. These correspond to create, read, update, and delete (or CRUD) operations respectively.

- There are other methods which are less frequently used like OPTIONS and HEAD.

Accura Tequipment

Rest API Basics



Accura Tequipment

GET:

- The HTTP GET method is used to read (or retrieve) a representation of a resource. In the safe path, GET returns a representation in XML or JSON and an HTTP response code of 200 (OK).
- In an error case, it most often returns a 404 (NOT FOUND) or 400 (BAD REQUEST).

POST:

- The POST verb is most often utilized to create new resources. In particular, it's used to create subordinate resources. That is, subordinate to some other (e.g. parent) resource.
- On successful creation, return HTTP status 201, returning a Location header with a link to the newly-created resource with the 201 HTTP status.

Accura Tequipment

PUT:

- It is used for updating the capabilities.
- PUT can also be used to create a resource in the case where the resource ID is chosen by the client instead of by the server. In other words, if the PUT is to a URI that contains the value of a non-existent resource ID.
- On successful update, return 200 (or 204 if not returning any content in the body) from a PUT. If using PUT for create, return HTTP status 201 on successful creation.
- PUT is not safe operation but it’s idempotent

Accura Tequipment

PATCH:

- It is used to modify capabilities.
- The PATCH request only needs to contain the changes to the resource, not the complete resource.
- This resembles PUT, but the body contains a set of instructions describing how a resource currently residing on the server should be modified to produce a new version. This means that the PATCH body should not just be a modified part of the resource, but in some kind of patch language like JSON Patch or XML Patch.
- PATCH is neither safe nor idempotent.

DELETE:

- It is used to delete a resource identified by a URI.
- On successful deletion, return HTTP status 200 (OK) along with a response body.

IoT Enabling Technologies

The IoT network is a heterogeneous network connecting many small hardware constraint devices and also where traditional security architectures and techniques cannot be applied. Therefore, it requires a different set of specialized techniques and architecture to provide security to the IoT network.

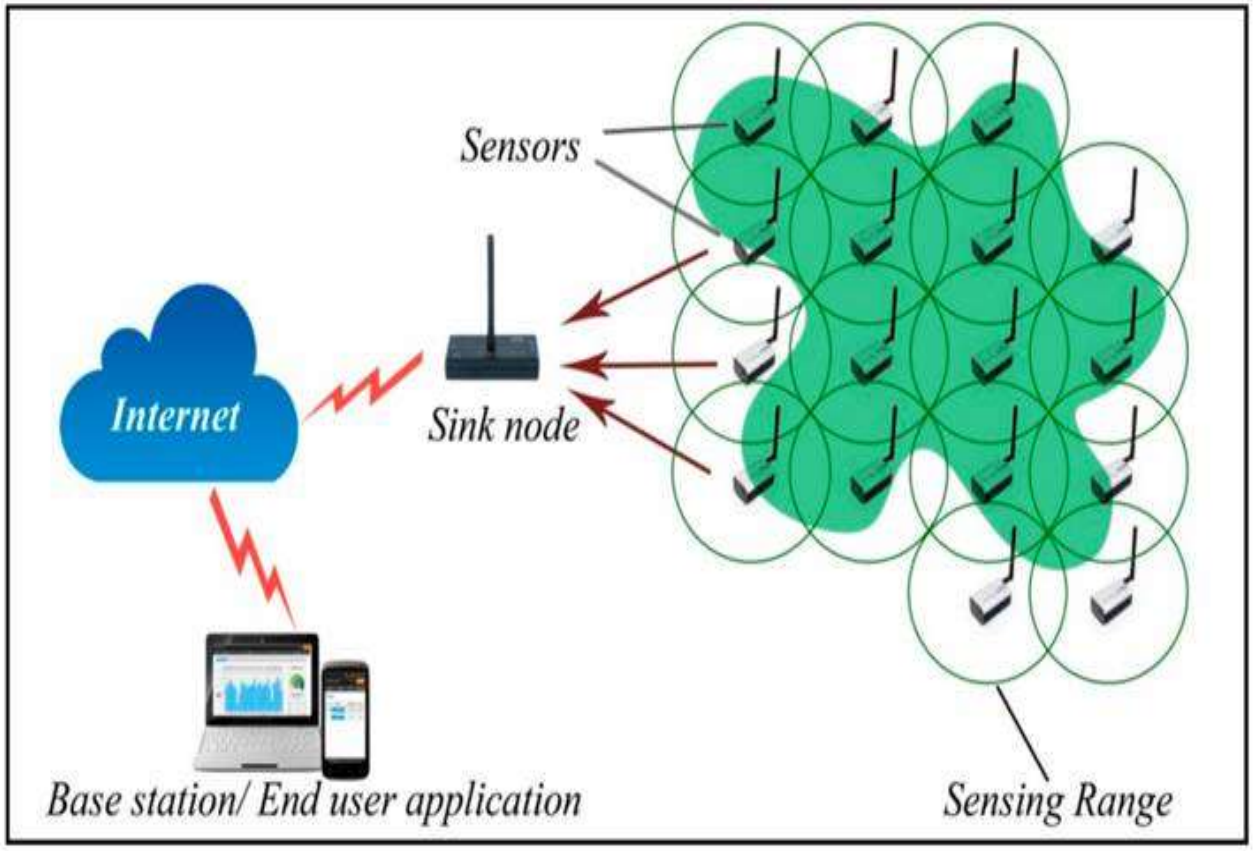
IoT(internet of things) enabling technologies are

- Wireless Sensor Network
- Cloud Computing
- Big Data Analytics
- Communications Protocols
- Embedded System

Wireless Sensor Network(WSN)

Accura Tequipment

- A Wireless Sensor Network (WSN) is a collection of devices which communicate through wireless channels.
- A WSN comprises distributed devices with sensors which are used to monitor the environmental and physical conditions.
- A wireless sensor network consists of end nodes, routers and coordinators.
- End nodes have several sensors attached to them where the data is passed to a coordinator with the help of routers.
- The coordinator also acts as the gateway that connects WSN to the internet.



Cloud Computing

- It provides us the means by which we can access applications as utilities over the internet.
- Cloud means something which is present in remote locations.
- With Cloud computing, users can access any resources from anywhere like databases, webservers, storage, any device, and any software over the internet.

Characteristics

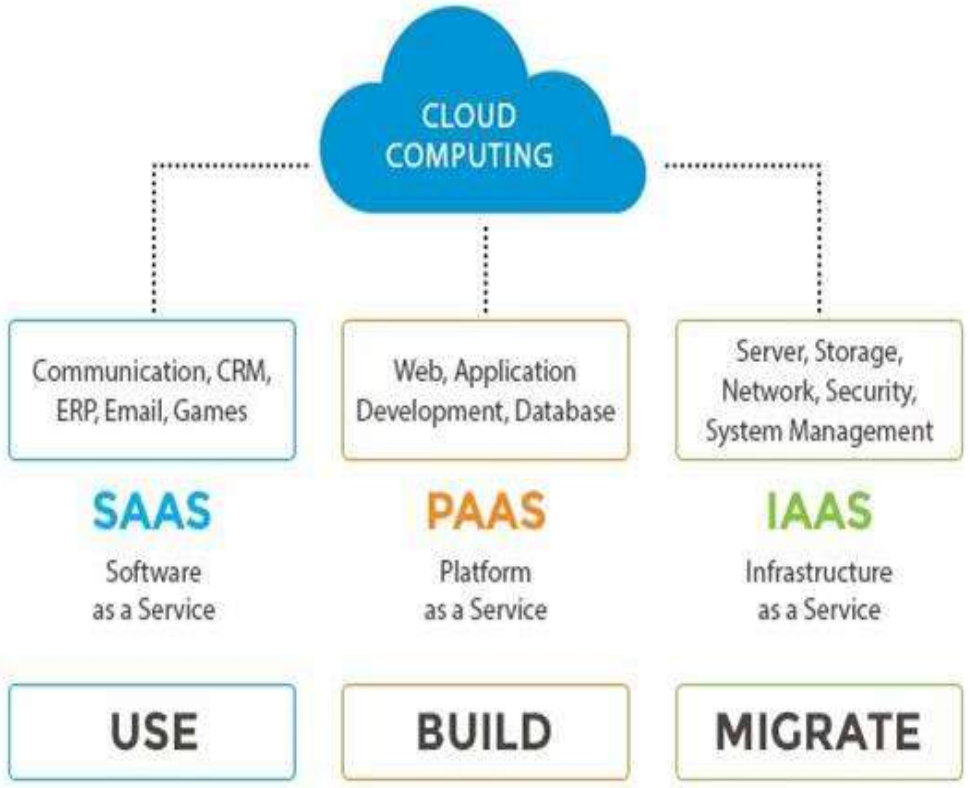
- On demand
- Autonomic
- Scalable
- Pay-per-use
- Ubiquitous

Accura Tequipment

Cloud computing offers three basic service models using which users can subscribe to cloud resources.

These service models are:

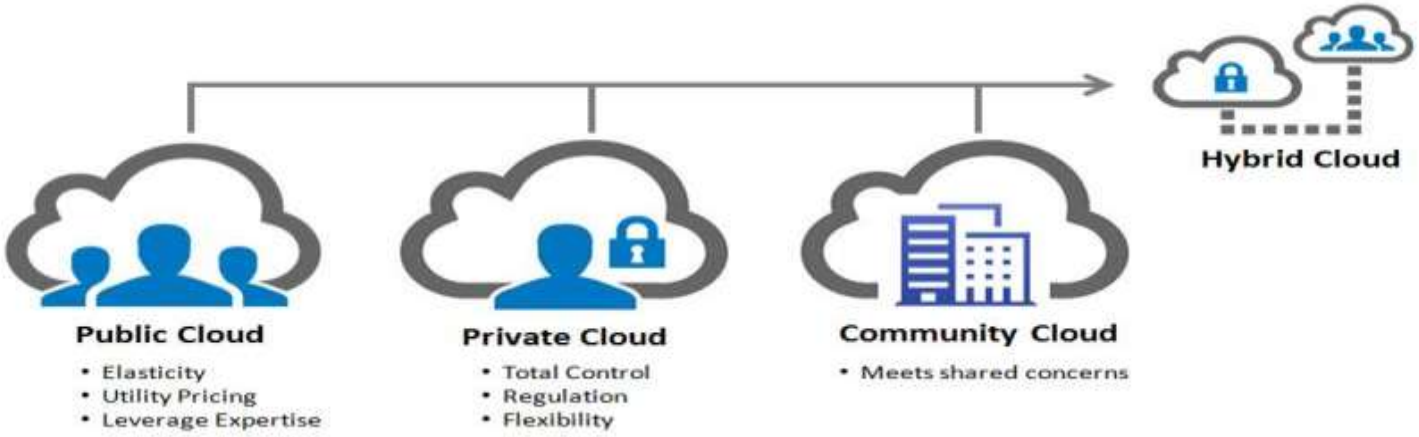
1. **IaaS (Infrastructure as a service)**
2. **PaaS (Platform as a service)**
3. **SaaS (Software as a service)**



Basic deployment models of cloud

Accura Tequipment

Cloud computing supports four basic deployment models

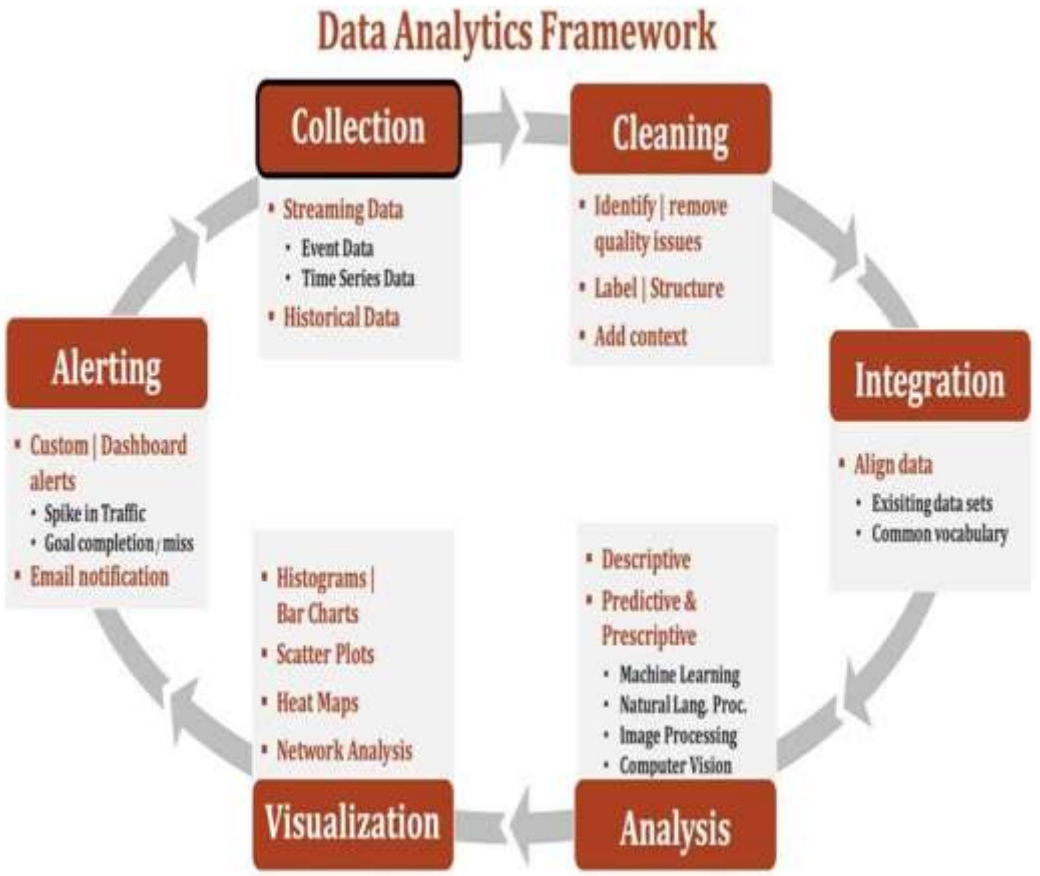


- **Public cloud :** In a public cloud the resources are shared between several users
- **Private cloud :** In a private cloud all the resources are used by a single organization
- **Community cloud :** A community cloud is one whose resources are shared by two or more companies having shared goals
- **Hybrid cloud :** A combination of the previous three clouds is a hybrid cloud

Big Data Analytics

Accura Tequipment

- It refers to the method of studying massive volumes of data or big data
- Big data is gathered from a variety of sources including social network videos, digital images, sensors and sales transaction records
- Several steps involved in analyzing big data – The data analytics framework consists of six steps namely: collection, cleaning, integration, analysis, visualization and alerting.



Communications Protocols

- They are the backbone of IoT systems and enable network connectivity and linking to applications

- Communication protocols allow devices to exchange data over the network

- A group of protocols designed to work together is known as a protocol suite; when implemented in software they are a protocol stack. They are used in

1. Data encoding
2. Addressing schemes

Embedded Systems

- It is a combination of hardware and software used to perform special tasks
- It includes microcontroller and microprocessor memory, networking units (Ethernet Wi-Fi adapters), input output units (display keyword etc.) and storage devices (flash memory)
- It collects the data and sends it to the internet
- **The characteristics of an embedded system are:**
  - Single-functioned
  - Tightly constrained
  - Reactive and Real time
  - Memory
  - Connected

USER INTERFACE

Accura Tequipment

- For any given IoT system, there needs to be a way of interacting with it. And just as we saw with sensors/devices, connectivity standards, and IoT platforms, there are many options you can pursue depending on your specific application and business needs.
- Users need a way to view and understand the data captured by IoT. That's where the user interface comes in. In the simplest terms, a user interface (UI for short) is the means by which a user and a computer system interact. Many think of UIs as just software or apps on phones and computers, but a user interface could be anything from a smartwatch to voice-controlled Amazon Echo to the buttons on a smart tractor dashboard.
- Apple pioneered the first graphical user interface (GUI) in 1983 with the introduction of Lisa. A graphical user interface is a visual way of interacting with a computer using items like buttons, windows, and icons. This meant that people didn't have to learn complex command languages to interact with computers and thus made the computer more accessible to everyday users. When they made the leap to touch interfaces and smartphone technology in 2009, Apple helped to further open up the door for new types of interfaces.

Mobile App

- Native apps are what most people think of when they think of mobile UIs.
- Native apps are applications that you download directly onto your phone.
- The advantage of native apps is that you have greater access to the phone's capabilities and can create a better overall user experience (we'll talk more about user experience below).
- The disadvantage is that they can take more time and resources to build, particularly because you need to build for both iOS and Android (iOS is the operating system created by Apple for iPhones and Android is an open-source operating system from Google) since they're not compatible with each other.

Web Apps

- Like a website, a web app is accessed by going to a certain url (e.g. https://examplewebapp.com).
- However, while websites are largely informational (like Wikipedia), web apps are built to have certain functionalities (like controlling a device remotely).
- The advantage of web apps is that they can work on both iOS and Android because you're just using a web browser instead of actually downloading something.
- Also, because you don't need to download anything, it can make it easier to get into the hands of users (just send them the url link).
- The disadvantage is that you have somewhat limited access to the phone's full capabilities (like the inability to send push notifications) and less control over the overall user experience.

Hybrid Apps

- As the name implies, hybrid apps are between Mobile apps and web apps. You still download something, like a web app, but when you open the app it is essentially opening a web page meaning that it can act like a web app.
- This can be a good option if you know you'll be creating Mobile apps eventually, but you want to get a minimum viable product into the hands of users early, and can therefore benefit from the speedier development offered by web apps.

Desktop App

- A desktop application is a software program that can be run on a standalone computer to perform a specific task by an end-user.
- Some desktop applications such as word editor, photo editing app and media player allow you to perform different tasks while other such as gaming apps are developed purely for entertainment.
- When you purchase a computer or a laptop, there is a set of apps that are already installed on your desktop. You can also download and install different desktop applications directly from the Internet or purchased from software vendors.
- The examples of some of the popular desktop applications are, word processing applications such as Microsoft Word and WPS Office which are designed to edit the textual content, gaming applications such as Minesweeper and Solitaire which are used for entertainment, web browsers such as Internet Explorer, Chrome and Firefox which help you to connect to the Internet from your computer, media player applications such as iTunes, Windows Media Player, and VLC media player which let you listen to music, watch videos and movies and create collections of media content.

IoT System Architecture and Operation with REST API