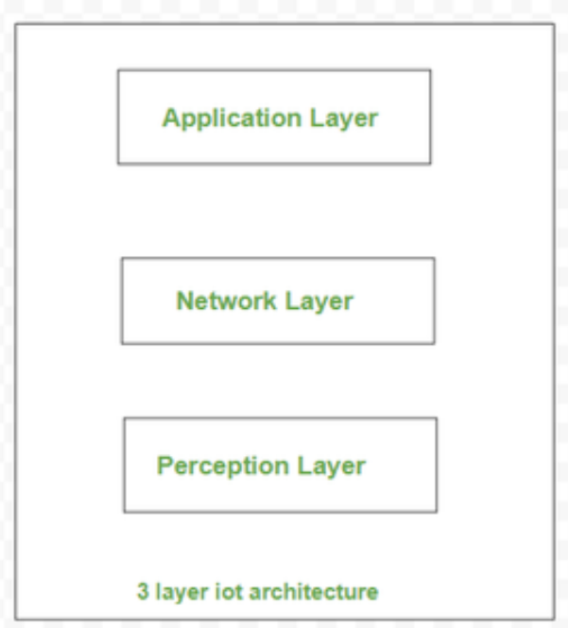
3 layer IoT architecture

IoT architecture is a framework that specifies the physical elements, network technical arrangement and setup, operating procedures, and data formats to be used. IoT architecture can differ greatly based on execution; it must be flexible enough for open protocols to handle many network applications.



1. Perception Layer :
This perception layer is the IoT architecture's physical layer. In these sensors and embedded systems are used mainly. These collect large amounts of data based on the requirements. This also includes edge devices, sensors, and actuators that communicate with the surroundings. It detects certain spatial parameters or detects other intelligent things /objects in the surroundings.

2. Network Layer :
The data obtained by these devices must be distributed and stored. This is the responsibility of the network layer. It binds these intelligent objects to other intelligent/ smart objects. It is also in charge of data transfer. The network layer is in-charge of linking smart objects, network devices, and servers. Its is also used to distribute and analyze sensor data.

3. Application Layer :
The user communicates with this application layer. It is in-charge of providing the customer with software resources. Example: in smart home application, where users press a button in the app to switch on a coffee machine, for example. The application layer is in-charge of providing the customer with application-specific resources. It specifies different uses for the IoT, such as smart houses, smart cities, and smart health.

Types of communication Models

https://www.geeksforgeeks.org/communication-models-in-iot-internet-of-things/?ref=rp

Types of Communications in IOT

IOT Communication :

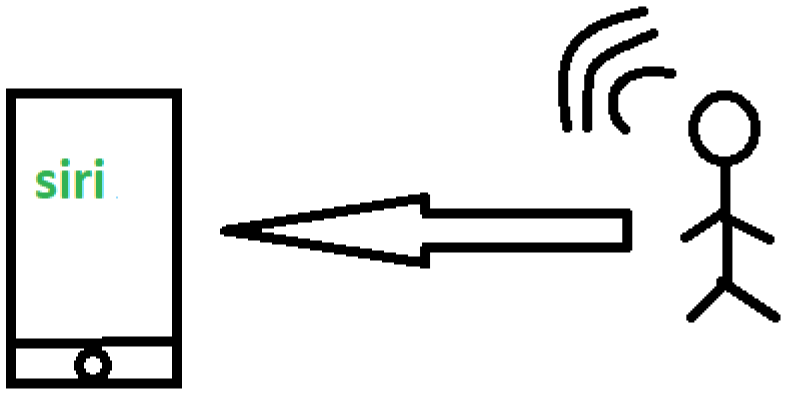
IoT is connection of devices over internet, where these smart devices communicate with each other , exchange data , perform some tasks without any human involvement. These devices are embedded with electronics, software, network and sensors which help in communication. Communication between smart devices is very important in IOT as it enables these devices to gather, exchange data which contribute in success of that IOT product/project.

Types of Communications in IOT :

The following are some communication types in IoT:-

1. **Human to Machine (H2M) :**

In this human gives input to IOT device i.e as speech/text/image etc. IOT device (Machine) like like sensors and actuators then understands input, analyses it and responds back to human by means of text or Visual Display. This is very useful as these machines assist humans in every everyday tasks. It is a combo of software and hardware that includes human interaction with a machine to perform a task.



H2M communication

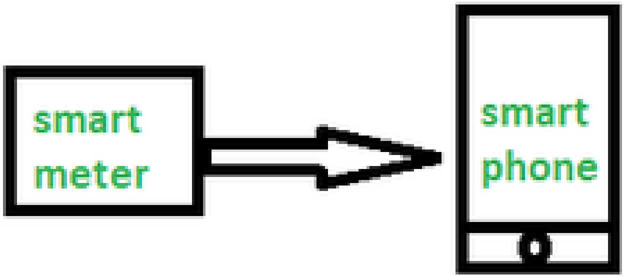
Merits: This H2M has a user-friendly interface that can be quickly accessed by following the instructions. It responds more quickly to any fault or failure. Its features and functions can be customized.

Examples:

- Facial recognition.
- Bio-metric Attendance system.
- Speech or voice recognition.

2. **Machine to Machine (M2M) :**

In this the interaction or communication takes place between machines by automating data/programs. In this machine level instructions are required for communication. Here communication takes place without human interaction. The machines may be either connected through wires or by wireless connection. An M2M connection is a point-to-point connection between two network devices that helps in transmitting information using public networking technologies like Ethernet and cellular networks. IoT uses the basic concepts of M2M and expands by creating large "cloud" networks of devices that communicate with one another through cloud networking platforms.



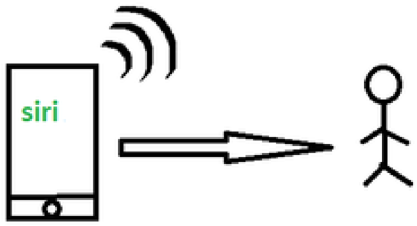
M2M communication

Advantages –
This M2M can operate over cellular networks and is simple to manage. It can be used both indoors and outdoors and aids in the communication of smart objects without the need for human interaction. The M2M contact facility is used to address security and privacy problems in IoT networks. Large-scale data collection, processing, and security are all feasible.
Disadvantages –
However, in M2M, use of cloud computing restricts versatility and creativity. Data security and ownership are major concerns here. The challenge of achieving interoperability between cloud/M2M IoT systems is daunting. M2M connectivity necessitates the existence of a reliable internet connection.
Examples:

- Smart Washing machine sends alerts to the owners' smart devices after completion of washing or drying of clothes.
- Smart meters tracks amount of energy used in household or in companies and automatically alert the owner.

3. **Machine to Human (M2H) :**

In this machine interacts with Humans. Machine triggers information(text messages/images/voice/signals) respective / irrespective of any human presence. This type of communication is most commonly used where machines guide humans in their daily life. It is way of interaction in which humans co-work with smart systems and other machines by using tools or devices to finish a task.



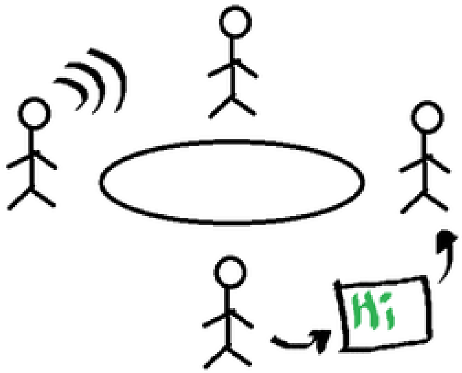
M2H communication

Examples:

- Fire Alarms
- Traffic Light
- Fitness bands
- Health monitoring devices

4. Human to Human (H2H) :

This is generally how humans communicate with each other to exchange information by speech, writing, drawing, facial expressions, body language etc. Without H2H, M2M applications cannot produce the expected benefits unless humans can immediately fix issues, solve challenges, and manage scenarios.

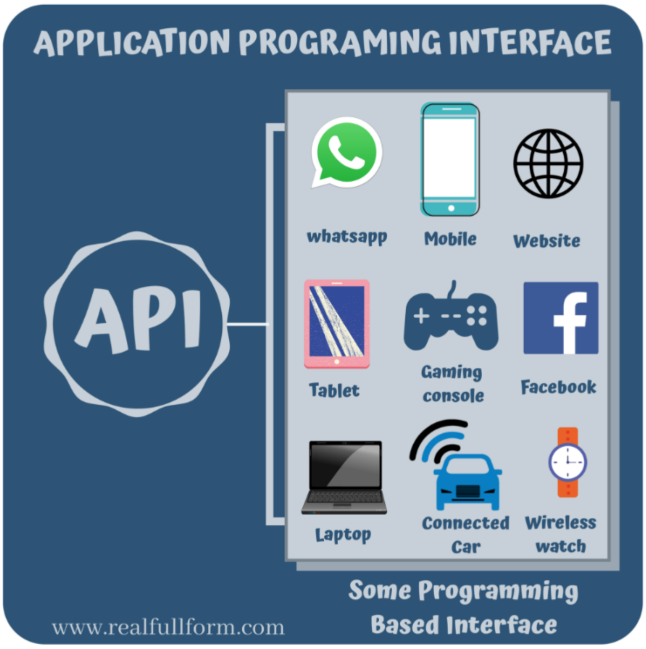


H2H communication

For, communication of IoT devices many protocols are used. These IoT protocols are modes of communication which give security to the data being exchanged between IoT connected devices. Example bluetooth, wifi, zigbee etc.

Application programming interface

An **API** is an intermediate software agent that allows dependent applications to communicate with each other. **APIs** provide a set of protocols, routines, and developer tools enabling software developers to extract and share information and let applications interact in an accessible manner.



REST

REST, or Representational State Transfer, is a commonly used API category that is not dependent on a specific protocol. It offers a flexible integration option that allows developers to use a standardized set of processes to achieve their goals. The architectural style is straightforward and streamlines the connection between the client and server. REST is considered a relatively user-friendly API to work with, and many developers are experienced in this technology.

SOAP

SOAP, or Simple Object Access Protocol, is an API that connects different platforms together through HTTP and XML. The structure and requirements for SOAP are more rigid than REST, and it's defined by a specific protocol. Web applications have started moving away from this older API type, as it's harder to implement flexible integration. However, this structure does allow for more stringent security measures and includes stateful operations without custom coding.

ASP.NET

ASP.NET is a specific form of a REST API designed around .NET technology. In most cases, a Windows server is required to easily work with this technology. The primary benefit to the .Net API type is the structured framework that's in place. If you're working with Windows-based technology, you can send HTTP protocol messages to a variety of platforms. The entire framework is lightweight and easy to work with, which can speed up development time and add flexibility into your third-party integration.

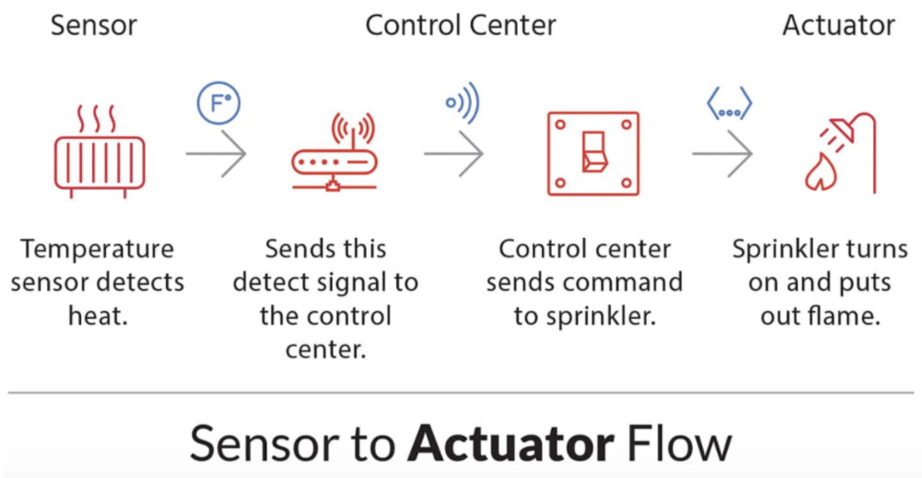
APIs are a powerful tool for working closely with your payment solution. You can improve your user experience by keeping customers on your site rather than sending them elsewhere, and you have many ways to work with the data and features available from your card payment solutions. When you're choosing a new payment provider, take a look at the API types they offer and what you can do with it in your existing infrastructure.

Unit 2
Development of things using Arduino platform

Introduction to IoT Node With sensor an actuators

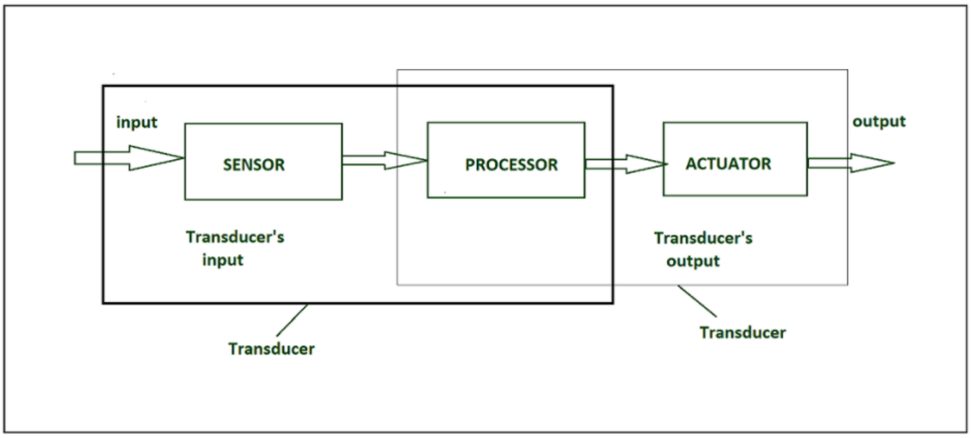
Many challenges come with the data collection, handling, communication, and processing of the data. These IoT devices collect a high amount of information, and it is up to the end user to decide which data is relevant for their situation, which places to process or store it, and the desired communication level. Storage, pre-processing, and processing of data can be done on a remote server or on the edge of the network itself.

Sensors, actuators, compute servers, and the communication network forms the core infrastructure of an IoT Framework.



Sensors in Internet of Things(IoT)

Sensors are used for sensing things and devices etc.
A device that provides a usable output in response to a specified measurement.
The sensor attains a physical parameter and converts it into a signal suitable for processing (e.g. electrical, mechanical, optical) the characteristics of any device or material to detect the presence of a particular physical quantity.
The output of the sensor is a signal which is converted to a human-readable form like changes in characteristics, changes in resistance, capacitance, impedance etc.



Transducer :

- A transducer converts a signal from one physical structure to another.
- It converts one type of energy into another type.
- It might be used as actuators in various systems.

Sensors characteristics :

1. Static
2. Dynamic

1. Static characteristics :

It is about how the output of a sensor changes in response to an input change after steady state condition.

- **Accuracy –**

Accuracy is the capability of measuring instruments to give a result close to the true value of the measured quantity. It measures errors. It is measured by absolute and relative errors. Express the correctness of the output compared to a higher prior system. Absolute error = Measured value – True value
Relative error = Measured value/True value

- **Range –**

Gives the highest and the lowest value of the physical quantity within which the sensor can actually sense. Beyond these values, there is no sense or no kind of response.
e.g. RTD for measurement of temperature has a range of -200`c to 800`c.

- **Resolution –**

Resolution is an important specification towards selection of sensors. The higher the resolution, better the precision. When the accretion is zero to, it is called threshold.
Provide the smallest changes in the input that a sensor is able to sense.

- **Precision –**

It is the capacity of a measuring instrument to give the same reading when repetitively measuring the same quantity under the same prescribed conditions. It implies agreement between successive readings, NOT closeness to the true value.
It is related to the variance of a set of measurements.
It is a necessary but not sufficient condition for accuracy.

- **Sensitivity –**

Sensitivity indicates the ratio of incremental change in the response of the system with respect to incremental change in input parameters. It can be found from the slope of the output characteristics curve of a sensor. It is the smallest amount of difference in quantity that will change the instrument's reading.

- **Linearity –**

The deviation of the sensor value curve from a particular straight line. Linearity is determined by the calibration curve. The static calibration curve plots the output amplitude versus the input amplitude under static conditions. A curve's slope resemblance to a straight line describes the linearity.

- **Drift –**

The difference in the measurement of the sensor from a specific reading when kept at that value for a long period of time.

- **Repeatability –**

The deviation between measurements in a sequence under the same conditions. The measurements have to be made under a short enough time duration so as not to allow significant long-term drift.

Dynamic Characteristics :

Properties of the systems

- **Zero-order system –**

The output shows a response to the input signal with no delay. It does not include energy-storing elements.
Ex. potentiometer measure, linear and rotary displacements.

- **First-order system –**

When the output approaches its final value gradually. Consists of an energy storage and dissipation element.

- **Second-order system –**

Complex output response. The output response of the sensor oscillates before steady state.

Sensor Classification :

- Passive & Active

- Analog & digital
- Scalar & vector

1. **Passive Sensor –**

Can not independently sense the input. Ex- Accelerometer, soil moisture, water level and temperature sensors.

2. **Active Sensor –**

Independently sense the input. Example- Radar, sounder and laser altimeter sensors.

3. **Analog Sensor –**

The response or output of the sensor is some continuous function of its input parameter. Ex- Temperature sensor, LDR, analog pressure sensor and analog hall effect.

4. **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).

5. **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.

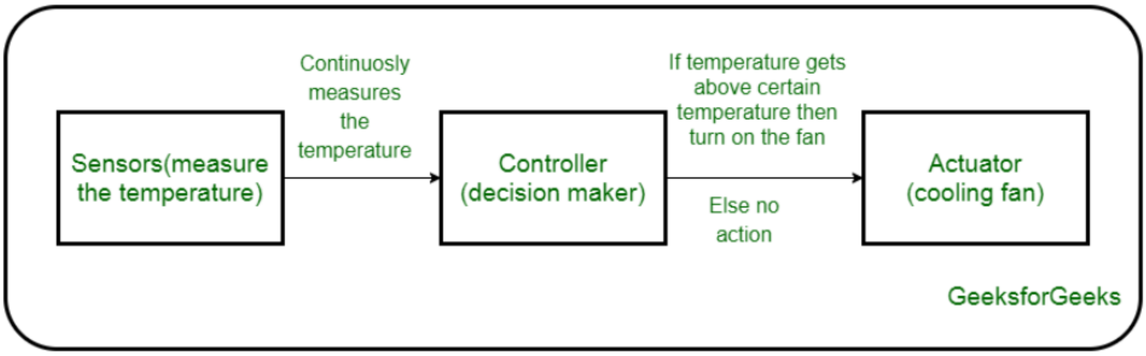
6. **Vector sensor –**

The response of the sensor depends on the magnitude of the direction and orientation of input parameter. Example – Accelerometer, gyroscope, magnetic field and motion detector sensors.

Actuators in IoT

An IoT device is made up of a Physical object ("thing") + Controller ("brain") + Sensors + Actuators + Networks (Internet). An actuator is a machine component or system that moves or controls the mechanism or the system. Sensors in the device sense the environment, then control signals are generated for the actuators according to the actions needed to perform.

A servo motor is an example of an actuator. They are linear or rotatory actuators, can move to a given specified angular or linear position. We can use servo motors for IoT applications and make the motor rotate to 90 degrees, 180 degrees, etc., as per our need.



Working of IoT devices and use of Actuators

The control system acts upon an environment through the actuator. It requires a source of energy and a control signal. When it receives a control signal, it converts the source of energy to a mechanical operation. On this basis, on which form of energy it uses, it has different types given below.

Types of Actuators :

1. Hydraulic Actuators –

A hydraulic actuator uses hydraulic power to perform a mechanical operation. They are actuated by a cylinder or fluid motor. The mechanical motion is converted to rotary, linear, or oscillatory motion, according to the need of the IoT device. Ex- construction equipment uses hydraulic actuators because hydraulic actuators can generate a large amount of force.

Advantages :

- Hydraulic actuators can produce a large magnitude of force and high speed.
- Used in welding, clamping, etc.
- Used for lowering or raising the vehicles in car transport carriers.

Disadvantages :

- Hydraulic fluid leaks can cause efficiency loss and issues of cleaning.
- It is expensive.

- It requires noise reduction equipment, heat exchangers, and high maintenance systems.

2. Pneumatic Actuators –

A pneumatic actuator uses energy formed by vacuum or compressed air at high pressure to convert into either linear or rotary motion. Example- Used in robotics, use sensors that work like human fingers by using compressed air.

Advantages :

- They are a low-cost option and are used at extreme temperatures where using air is a safer option than chemicals.
- They need low maintenance, are durable, and have a long operational life.
- It is very quick in starting and stopping the motion.

Disadvantages :

- Loss of pressure can make it less efficient.
- The air compressor should be running continuously.
- Air can be polluted, and it needs maintenance.

3. Electrical Actuators –

An electric actuator uses electrical energy, is usually actuated by a motor that converts electrical energy into mechanical torque. An example of an electric actuator is a solenoid based electric bell.

Advantages :

- It has many applications in various industries as it can automate industrial valves.
- It produces less noise and is safe to use since there are no fluid leakages.
- It can be re-programmed and it provides the highest control precision positioning.

Disadvantages :

- It is expensive.
- It depends a lot on environmental conditions.

Other actuators are –

4. **Thermal/Magnetic Actuators –**

These are actuated by thermal or mechanical energy. Shape Memory Alloys (SMAs) or Magnetic Shape-Memory Alloys (MSMAs) are used by these actuators. An example of a thermal/magnetic actuator can be a piezo motor using SMA.

5. **Mechanical Actuators –**
   A mechanical actuator executes movement by converting rotary motion into linear motion. It involves pulleys, chains, gears, rails, and other devices to operate. Example – A crankshaft.

- Soft Actuators
- Shape Memory Polymers

- Light Activated Polymers
- With the expanding world of IoT, sensors and actuators will find more usage in commercial and domestic applications along with the pre-existing use in industry.

7 ways to align IoT user interfaces

UX ORIENTATIONS	DESCRIPTION	SAMPLE SCENARIO	UX CONSIDERATIONS
Objective	The goal, task, improvement, cost, energy or time-saving outcome motiviating the user.	Gesture- or voice-based request for turning on the lights is superior to scrolling through mobile app. No confirmation needed.	■ Quick request ■ Quick feedback ■ Necessity of confirmation ■ Task/info complexity
Personal context	The user's persona, demographics, health, culture, etc., must dictate design decisions and protections.	Wearables for seniors might be stylish and voice-activated to avoid concerns around disability, usability or loss of control.	■ Task/info complexity ■ Task/info sensitivity ■ Form-factor sensitivity
Environmental context	Spatial, location-based, temporal, environmental or social condsiderations, including fixed vs. mobile IoT configurations.	Voice interaction is great for in-home, hands-free use cases such as cooking, but suboptimal while others are sleeping or when in public.	■ Frequency of interaction ■ Conspicuousness ■ Environmental conditions ■ Hands-free/heads-up ■ Noise allowance ■ Task/info complexity ■ Single product vs. multiple
Universality and inclusion	Ease of comprehension and usability to reduce or eliminate need for training, as well as to avoid alienating particular segments or disenfranchising less tech-savvy users.	A check mark upon completion of a task is a universally recognized confirmation signal. Be aware of cultural variations in gestures, colors, avatars, etc.	■ Access to service, skill ■ Quick feedback ■ Status indications ■ Connectivity to others
Energy or power	What power requirements inform optimal UX including implications and risks of outage.	Connected door locks should never rely solely on Wi-Fi, in case connectivity is compromised. Consider the "no power" scenario.	■ Low- vs. high-energy needs ■ Local vs. cloud ■ Minimum viable connectivity protocol ■ Battery vs. fixed
Security	Safety and security improvemnts and risks introduced.	Information delivery of sensitive alerts like health reports may not be best vocalized, but rather sent directly to a password-protected mobile app.	■ Hands-free ■ Heads-up ■ Task/info complexity ■ Task/info sensitivity ■ Risk mitigation
Consent	Level of personal, sensitive or private data required for transaction or interaction.	If a user requests a task that requires systems to share data with a third party unverified by the user, alert user and solicit permission for transaction.	■ Task/info complexity ■ Task/info sensitivity ■ Agency, control