Unit 4. Introduction to Raspberry pi and Arduiano

4.1 Introduction on IoT Devices

IoT devices are the nonstandard computing devices that connect wirelessly to a network and have the ability to transmit data, such as the many devices on the internet of things (IoT).

IoT involves extending internet connectivity beyond standard devices, such as desktops, laptops, smartphones and tablets, to any range of traditionally "dumb" or non-internet-enabled physical devices and everyday objects. Embedded with technology, these devices can communicate and interact over the internet. They can also be remotely monitored and controlled.

Example:

In a smart home devices are designed to sense and respond to a person's presence. When a person arrives home, their car communicates with the garage to open the door. Once inside, the thermostat is already adjusted to their preferred temperature, and the lighting is set to a lower intensity and color, as their smart watch data indicates it has been a stressful day. Other smart home devices include sprinklers that adjust the amount of water given to the lawn based on the weather forecast and robotic vacuum cleaners that learn which areas of the home must be cleaned most often.

How do IoT devices work?

IoT devices vary in terms of functionality, but IoT devices have some similarities in how they work. First, IoT devices are physical objects designed to interact with the real world in some way. The device might be a sensor on an assembly line or an intelligent security camera. In either case, the device is sensing what's happening in the physical world.

The device itself includes an integrated CPU, network adapter and firmware, which is usually built on an open source platform. In most cases, IoT devices connect to a Dynamic Host Configuration Protocol server and acquire an IP address that the device can use to function on the network. Some IoT devices are directly accessible over the public internet, but most are designed to operate exclusively on private networks.

Although not an absolute requirement, many IoT devices are configured and managed through a software application. Some devices, however, have integrated web servers, thus eliminating the need for an external application.

Once an IoT device has been configured and begins to operate, most of its traffic is outbound. A security camera, for example, streams video data. Likewise, an industrial sensor streams sensor data. Some IoT devices such as smart lights, however, do accept inputs.

What is IoT device management?

Several challenges can hinder the successful deployment of an IoT system and its connected devices, including security, interoperability, power/processing capabilities, scalability and availability. Many of these can be addressed with IoT device management either by adopting standard protocols or using services offered by a vendor.

Device management helps companies integrate, organize, monitor and remotely manage internet-enabled devices at scale, offering features critical to maintaining the health, connectivity and security of the IoT devices along their entire lifecycles. Such features include:

- Device registration and activation
- Device authentication/authorization
- Device configuration
- Device provisioning
- Device monitoring and diagnostics
- Device troubleshooting
- Device firmware updates

Available standardized device management protocols include the Open Mobile Alliance's Device Management and Lightweight Machine-to-Machine.

IoT device management services and software are also available from vendors, including Amazon, Microsoft, Google, IBM, GE and many others.

IoT device connectivity and networking

The networking, communication and connectivity protocols used with internet-enabled devices largely depend on the specific IoT application deployed. Just as there are many different IoT applications, there are many different connectivity and communication options.

Communication protocols include CoAP, DTLS, MQTT, DDS and AMQP. Wireless protocols include IPv6, LPWAN, Zigbee, Bluetooth Low Energy, Z-Wave, RFID and NFC. Cellular, satellite, Wi-Fi and Ethernet can also be used.

Each option has its tradeoffs in terms of power consumption, range and bandwidth, all of which must be considered when choosing connected devices and protocols for a particular IoT application.

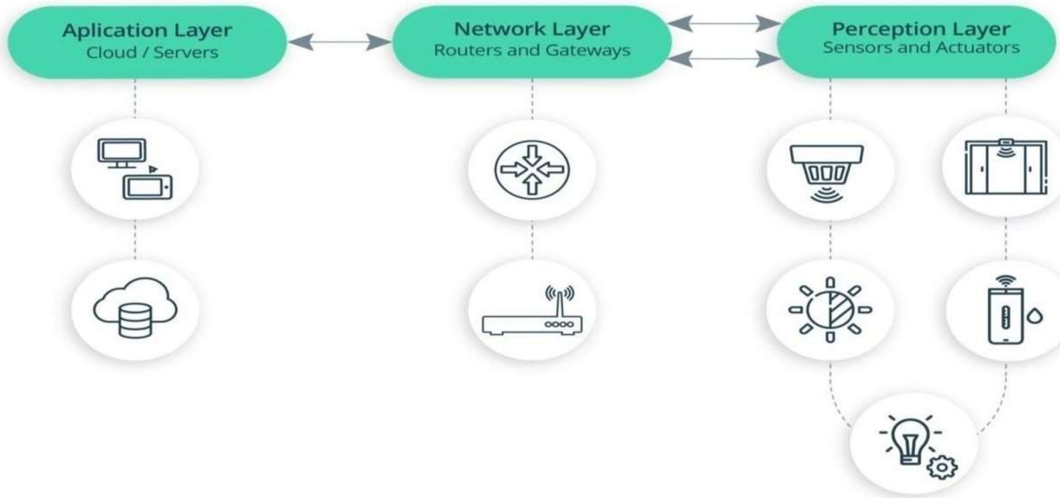
In most cases, IoT devices connect to an IoT gateway or another edge device where data can either be analyzed locally or sent to the cloud for analysis. Some devices have integrated data processing capabilities that minimized the amount of data that must be sent to the cloud or to the data center. This type of processing often uses machine learning capabilities that are integrated into the device, and is becoming increasingly popular as IoT devices create more and more data.

4.2 Basic Building blocks of an IoT Device

The IoT building blocks or the fundamental tiers of IoT architecture

As IoT platform building blocks are very diverse in terms of industry-specific use cases, there's no iconic IoT architecture that can fully address all possible applications. The most widely used technical building blocks of IoT architecture contain three layers:

- **The perception layer**, the purpose of which is to interact with the physical world via basic building blocks of IoT devices augmented with sensors and actuators.
- **The network layer**, which is intended to conduct superficial analysis of data captured by sensors using IoT gateways and transmit that data to a server for further processing over specific communication protocols.
- **The application layer**, which is responsible for the interconnection between the previous two hardware-related layers and business applications to deliver application-specific services to end users.



The basic building blocks of an IoT device are considerably more than just a set of sensors and actuators. Within the concept of the Internet of Things software development, these devices are connected into a single network with analytical and managerial systems to enable data-driven decision-making and automated device management. Let's get a proper understanding of what the role of each architectural element is and how the IoT system building blocks are combined to complete the IoT puzzle.

IoT sensors:

The practice of collecting and analyzing data on a particular object using sensors is nothing new. A variety of sensors have been used across industries for years, specifically in vehicles, buildings, industrial facilities, and factories, where they have deservedly become the backbone of industrial automation.

The greatest thing about sensors is their ability to convert information from the surrounding environment into data suitable for further processing and analysis. In fact, anything referred to as "smart" is built upon sensing devices.

Data collected by sensors can have varying degrees of complexity, ranging from something as simple as a temperature or humidity reading to as complex as video transmission. Depending on the use case, a standalone sensor can be used or multiple sensors can be bundled into a group or embedded into a device (such as a smartphone or camera) to provide more comprehensive data, becoming basic building blocks of IoT devices.

IoT actuators:

Both sensors and actuators are transducers, which means they're responsible for converting signals from one form to another. Yet unlike sensors, which are aimed at tracking data that comes from a device, actuators are responsible for performing actions. Actions can be, for instance, switching a light on or off, generating a sound, or locking a door. Typically, sensors and actuators work together to remotely monitor and control physical processes or systems. In this setup, however, sensors and actuators are only endpoints that connect physically to the environment, whereas analysis of the information they collect happens in a control layer, commonly referred to as an IoT gateway.

IoT gateways:

An IoT gateway is a device that's used for basic analysis of data coming from connected sensors. In some cases, when data analysis requires a small amount of computing resources, IoT gateways can serve as decision points, sending certain control commands to actuators which, in turn, perform appropriate actions. If more sophisticated data processing is needed or information is to be stored for further analysis, gateways send it to a server located either on-premises or in the cloud.

Both microcomputers and microprocessors can be used as gateways for IoT applications, with the most widely used being open-source platforms such as **Raspberry** Pi and **Arduino**.

IoT connectivity:

Now that it's clear what basic building blocks of an IoT device are used to create the IoT infrastructure, let us move on to how they interact with each other and the cloud environment or physical data centers.

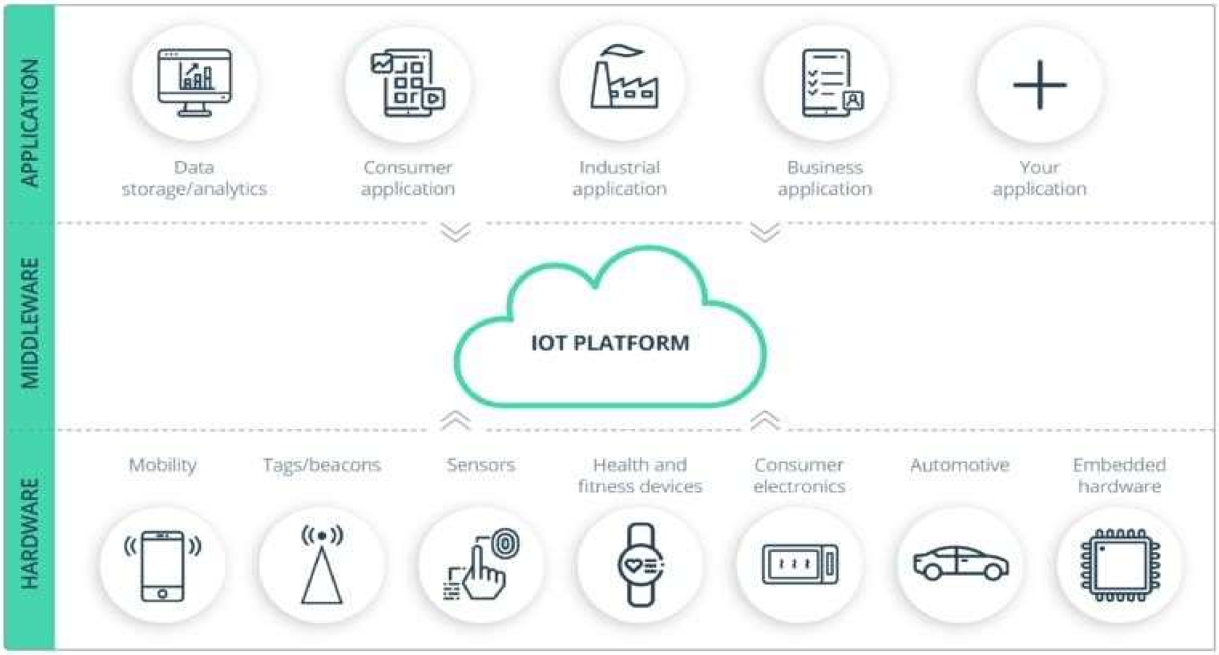
End devices and servers can be connected in various ways: via cellular or satellite networks, Wi-Fi, Bluetooth, low-power wide-area networks (LPWAN), as well as Ethernet (wired LAN).

	Short Range	Local Area	Wide Area
Range (typical)	< 10m/30ft	< 100m/300ft	Outdoor (Km/Miles)
Content distribution Focus on high data rates	Bluetooth	WiFi	GSM, lte, 5G
Sense & control Low energy/long battery life	Bluetooth SMART		GPRS, NB-IoT
Proprietary solutions	ANT+	WAVE, enocean	SIGFOX, LoRa, INGENU
Typical applications	Personal appliance (wrist band, smart watch, step counter, keybord, mouse, pointer, etc.)	Idoor networks (internet, email, phone securitu, energy management, home monitoring, etc.)	Outdoor networks (phone, chat, internet, smart city, industry 4.0, agriculture, smart logistics, etc.)

Major considerations when deciding on a particular connectivity option include range, bandwidth, power consumption and, of course, cost. However, regardless of which connectivity method you choose, they all accomplish the same task – getting data to the server, be it in the cloud or on-premise.

IoT platforms:

An IoT platform is middleware between the hardware-related layers of an IoT ecosystem, on the one hand, and the application layers, on the other. It's a multi-tiered technology that helps to collect, store, process, and visualize data, as well as integrate it with other parts of the value chain, such as the cloud or end-user app. Additionally, it takes care of cross-device compatibility and scalability and has out-of-the-box features that accelerate application development. An IoT platform helps to manage connected devices using various protocols, configurations, and network topologies and even conducts over-the-air firmware updates.



There are loads of IoT platforms for any taste, budget, and needs. According to Statista, Microsoft Azure, Google Cloud, and Amazon Web Services are among the top platforms used to run connected devices in software projects. However, choosing one of the market giants doesn't guarantee your project's success. In most cases, IoT platforms are tailored for specific industries or focus on a certain type of device — for example, beacons, smart watches, or even augmented reality headsets. Picking the right platform is crucial to addressing the needs of your business. In making your choice, you should start with defining your IoT strategy, the kinds of problems you're going to solve, and the results you want to achieve.

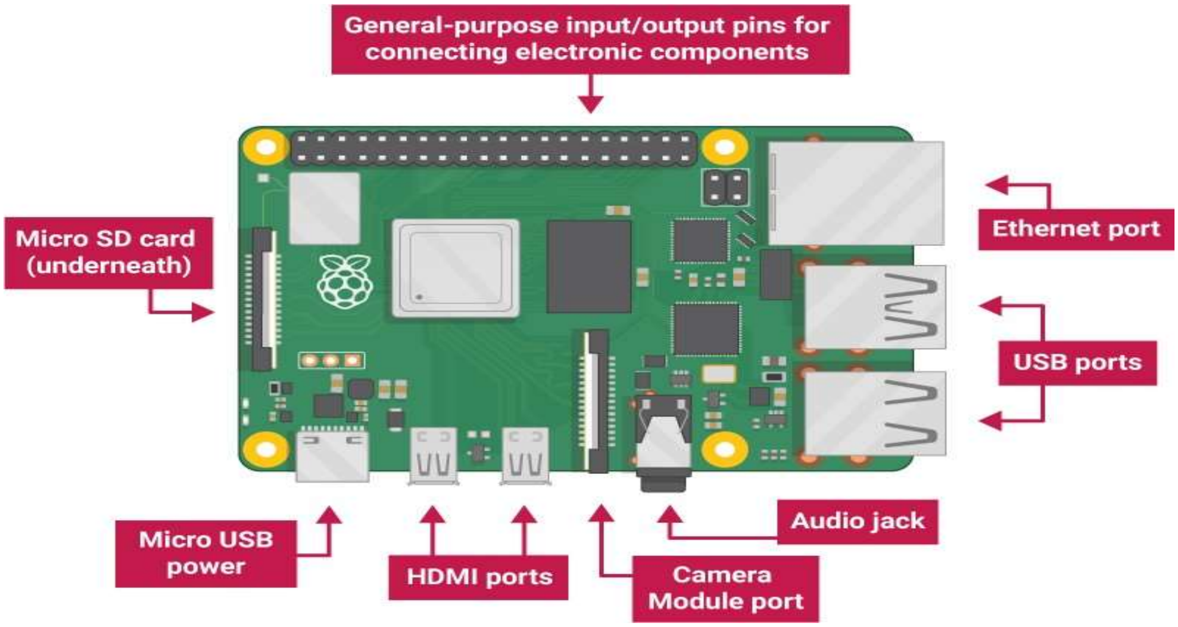
4.3 Introduction to Raspberry pi (Concepts, purpose, Application areas)

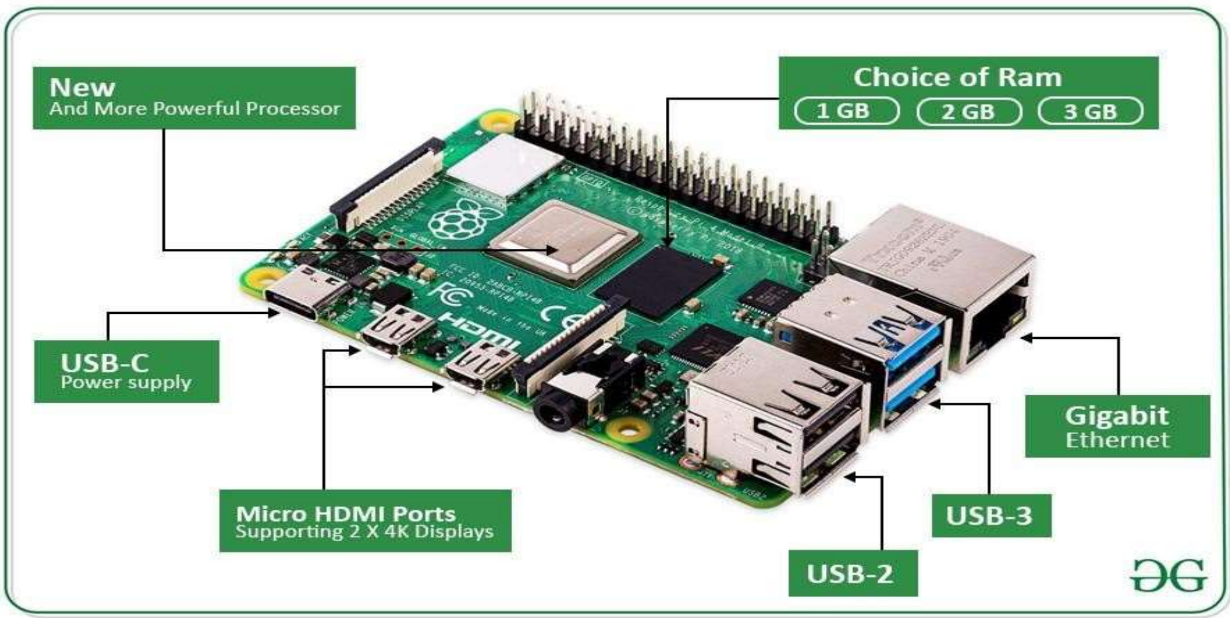
Concepts:

Raspberry Pi, developed by Raspberry Pi Foundation in association with Broadcom, is a series of small single-board computers and perhaps the most inspiring computer available today.

From the moment you see the shiny green circuit board of Raspberry Pi, it invites you to tinker with it, play with it, start programming, and create your own software with it. Earlier, the Raspberry Pi was used to teach basic computer science in schools but later, because of its low cost and open design, the model became far more popular than anticipated.

It is one of the best-selling British computers and most of the boards are made in the Sony factory in Pencoed, Wales.





Screen Mirroring of Android Smartphone using Raspberry Pi

Generations and Models:

In 2012, the company launched the Raspberry Pi and the current generations of regular Raspberry Pi boards are **Zero, 1, 2, 3, and 4**.

Generation 1 Raspberry Pi had the following four options −

- Model A

- Model A +
- Model B
- Model B +

Among these models, the **Raspberry Pi B models** are the original credit-card sized format.

On the other hand, the **Raspberry Pi A models** have a smaller and more compact footprint and hence, these models have the reduced connectivity options.

Raspberry Pi Zero models, which come with or without GPIO (general-purpose input output) headers installed, are the most compact of all the Raspberry Pi boards types.

The table below gives the connectivity specifications of various Raspberry Pi boards focusing on the version's video out quality, video in, Ethernet, bluetooth, Wi-Fi and external storage –

Raspberry Pi Version	Video Out Quality	Video In	Ethernet	Bluetooth	Wi-Fi	External Storage
Raspberry Pi 4 Model B	4kp60	CSI Camera Connector	Gigabit Ethernet	Bluetooth 5.0	Dual Band- 2.4 GHz and 5GHz	MicroSD
Raspberry Pi 3 Model B+	1080p60	CSI Camera Connector	10/100 Mbit/s	Bluetooth 4.2/BLE	Dual Band- 2.4 GHz and 5GHz	MicroSD
Raspberry Pi 3 Model B	1080p60	CSI Camera Connector	10/100 Mbit/s	Bluetooth 4.1	2.4 GHz	MicroSD
Raspberry Pi 3 Model A+	1080p60	CSI Camera Connector	____	Bluetooth 4.2/BLE	Dual Band- 2.4 GHz and 5GHz	MicroSD
Raspberry Pi Zero Wireless with	1080p60	CSI Camera Connector	____	Bluetooth 4.1	2.4 GHz	MicroSD

Headers						
Raspberry Pi Zero Wireless	1080p60	CSI Camera Connector	____	Bluetooth 4.1	2.4 GHz	MicroSD
Raspberry Pi Zero	1080p60	CSI Camera Connector	____	____	____	MicroSD
Raspberry Pi 2 Model B	1080p60	CSI Camera Connector	10/100 Mbit/s	____	____	MicroSD
Raspberry Pi 1 Model B +	1080p60	CSI Camera Connector	10/100 Mbit/s	____	____	MicroSD

Purpose:

All over the world, people use the Raspberry Pi to learn programming skills, build hardware projects, do home automation, implement Kubernetes clusters and Edge computing, and even use them in industrial applications.

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python.

It is widely used to make gaming devices, fitness gadgets, weather stations, and much more. But apart from that, it is used by thousands of people of all ages who want to take their first step in computer science.

The Raspberry Pi operates in the open source ecosystem: it runs Linux (a variety of distributions), and its main supported operating system, Pi OS, is open source and runs a suite of open source software. The Raspberry Pi Foundation contributes to the Linux kernel and various other open source projects as well as releasing much of its own software as open source.

Application areas:

1. **Replace Your Desktop PC With a Raspberry Pi:**

The simplest way to use a Raspberry Pi is as a desktop computer. This is particularly suited to the Raspberry Pi 3, 4, and Raspberry Pi 400.

Along with the Pi itself, the microSD card, and power supply, you'll need a HDMI cable and a suitable display. For the Pi 3 and 4, you'll also need a keyboard and mouse – USB or Bluetooth. Later Raspberry Pis have built-in Wi-Fi and Bluetooth.

All you need now is an operating system. Raspberry Pi OS and Ubuntu are ideal, and both come with the relevant Raspberry Pi applications built in.

2. Build a Raspberry Pi Retro Gaming Machine:

As well as streaming media, the Raspberry Pi is ideal as a retro gaming machine. Compact and powerful enough to be used in several ways, the device is suitable as a full size arcade cabinet, or even as part of a Game Boy-esque handheld.

Various retro gaming operating systems are available, all with controller support. Many classic gaming platforms can be emulated, from MS-DOS and 16-bit consoles to the Commodore 64.

3. Build a Stop-Motion Camera:

Stop-motion video is becoming increasingly popular as an art form, with uploaders of all ages sharing their movies on YouTube and social media.

But how is stop-motion made? You can find out with a Raspberry Pi and a dedicated camera module.

Using the Python programming language, a suitable mount (overhead for Gilliam-esque paper craft animation, a standard tripod for clay- or toy-based), and a well-lit area, you'll also need to rig a button to the Pi's GPIO.

Beware: stop-motion photography is a time-consuming process. Some practice is needed to get good results, and you'll need to add a soundtrack.

4. **Build a Raspberry Pi Web Server:**

Need to host a website? A Raspberry Pi can be configured to host a homepage or blog, with either static web pages or database-driven content.

You'll need Apache, and for more advanced websites, a full LAMP stack, with PHP and MySQL alongside Apache. It's useful if you also set up FTP.

Once these steps are completed, save HTML files into the /www/ directory, and your web server is ready. Or you might install some specific web software like WordPress.

5. **Build a Network Monitoring Tool:**

Need to monitor the devices on your network? Concerned about a lack of connectivity, or want quick notice when your website is offline?

Several network monitoring tools are available, but none is as easy to install and configure as Nagios. Once installed on your Raspberry Pi, you can monitor uptime, view a visualization of the devices on your network, and more.

While you could use a PC, a Raspberry Pi 2 or later is ideal for this project.

6. Print With Your Raspberry Pi

7. Build a Minecraft Game Server

8. Broadcast a Pirate FM Radio Station

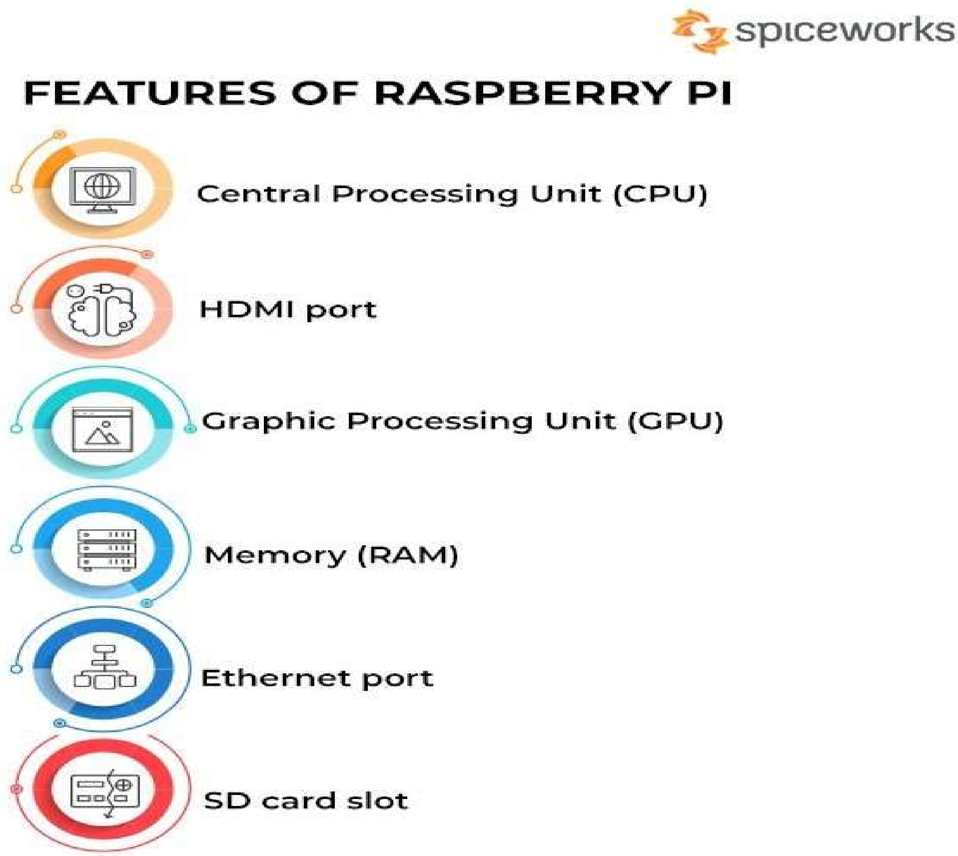
9. Create a Twitter Bot

10. Stream Live Video to YouTube

11. Stream PC Games to Raspberry Pi

12. Install Games on the Raspberry Pi

4.4 Components of Raspberry pi



Central Processing Unit (CPU)

Every computer has a Central Processing Unit, and so does the Raspberry Pi. It is the computer's brain and carries out instructions using logical and mathematical operations. Raspberry Pi makes use of the ARM11 series processor on its boards.

HDMI Port

Raspberry Pi board has an HDMI or High Definition Multimedia Interface port that allows the device to have video options of the output from the computer displayed. An HDMI cable connects the Raspberry Pi to an HDTV. The supported versions include 1.3 and 1.3. It also comes with an RCA port for other display options.

Graphic Processing Unit (GPU)

This unit, GPU or Graphic Processing Unit, is another part of the Raspberry pi board. Its primary purpose is to hasten the speed of image calculations.

Memory (RAM)

Random Access Memory is a core part of a computer's processing system. It is where real-time information is stored for easy access. The initial Raspberry Pi had 256MB RAM. Over the years, developers gradually and significantly improved the size. Different Raspberry Pi models come with varying capacities. The model with the maximum capacity presently is the Raspberry Pi 4 with 8GB RAM space.

Ethernet Port

The Ethernet port is a connectivity hardware feature available on B models of Raspberry Pi. The Ethernet port enables wired internet access to the minicomputer. Without it, software updates, web surfing, etc., would not be possible using the Raspberry Pi. The Ethernet port found on Raspberry computers uses the RJ45 Ethernet jack. With this component, Raspberry Pi can connect to routers and other devices.

SD Card Slot

Like most other regular computers, Raspberry Pi must have some sort of storage device. However, unlike conventional PCs, it does not come with a hard drive, nor does it come with a memory card. The Raspberry Pi board has a Secure Digital card or SD card slot where users must insert SD cards for the computer to function. The SD card functions like a hard drive as it contains the operating system necessary for turning the system on. It also serves to store data.

General Purpose Input and Output (GPIO)

These are upward projecting pins in a cluster on one side of the board. The oldest models of the Raspberry Pi had 26 pins, but most have 40 GPIO pins. These pins are pretty sensitive and should be handled carefully. They are essential parts of the Raspberry Pi device as they add to its diverse applications. GPIO pins are used to interact with other electronic circuits. They can read and control the electric signals from other boards or devices based on how the user programs them.

LEDs

These are a group of five light-emitting diodes. They signal the user on the present status of the Raspberry Pi unit. Their function covers

- **PWR (Red):** This functions solely to indicate power status. When the unit is on, it emits a red light and only goes off when the unit is switched off, or disconnected from the power source.
- **ACT (Green):** This flashes to indicate any form of SD card activity.
- **LNK (Orange):** LNK LED gives off an orange light to signify that active Ethernet connectivity has been established.
- **100 (Orange):** This light comes on during Ethernet connection when the data speed reaches 100Mbps.
- **FDX (Orange):** FDX light also comes during Ethernet connection. It shows that the connection is a full-duplex.

USB Ports

Universal service bus (USB) ports are a principal part of Raspberry Pi. They allow the computer to connect to a keyboard, mouse, hard drives, etc. The first model of Raspberry Pi had only two USB 2.0 ports. Subsequent models increased this number to four. Raspberry Pi 4 and Pi 400, much newer models, come with a mix of USB 2.0 and USB 3.0 ports.

Power Source

Raspberry Pi has a power source connector that typically uses a 5V micro USB power cable. The amount of electricity any Raspberry Pi consumes depends on what it’s used for and the number of peripheral hardware devices connected.

4.5 Introduction to Arduiano (Concept, purpose and Application areas)

Concept:

Arduino is a software as well as hardware platform that helps in making electronic projects. It is an open source platform and has a variety of controllers and microprocessors. There are various types of Arduino boards used for various purposes.

The Arduino is a single circuit board, which consists of different interfaces or parts. The board consists of the set of digital and analog pins that are used to connect various devices and components, which we want to use for the functioning of the electronic devices.

Most of the Arduino consists of 14 digital I/O pins.

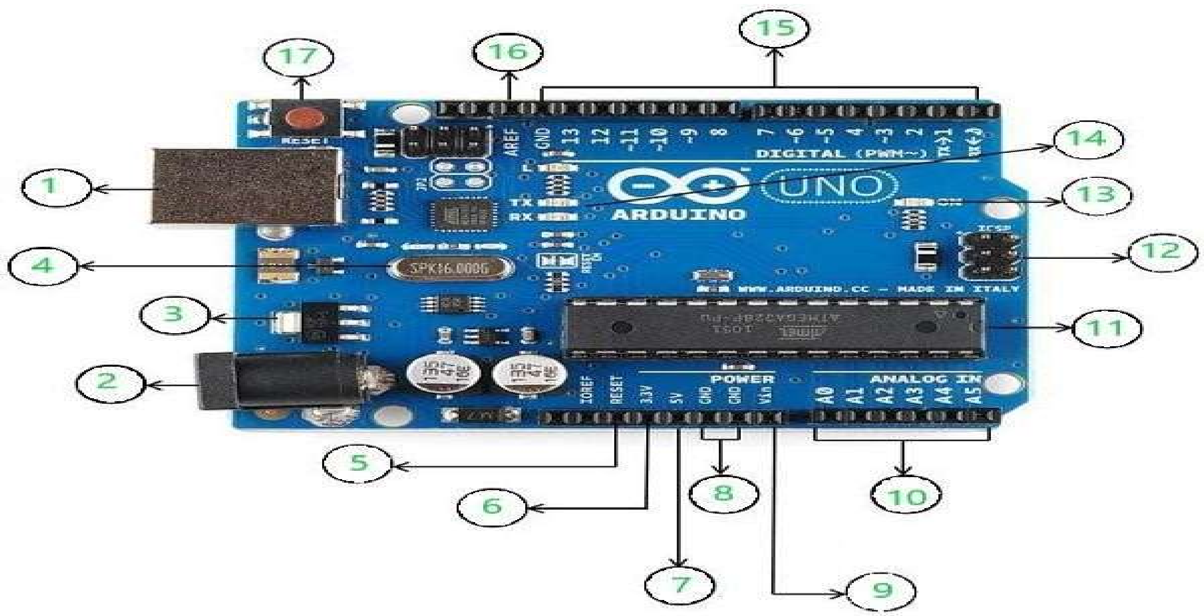
The analog pins in Arduino are mostly useful for fine-grained control. The pins in the Arduino board are arranged in a specific pattern. The other devices on the Arduino board are USB port, small components (voltage regulator or oscillator), microcontroller, power connector, etc.

Features:

The features of Arduino are listed below:

- Arduino programming is a simplified version of C++, which makes the learning process easy.
- The Arduino IDE is used to control the functions of boards. It further sends the set of specifications to the microcontroller.
- Arduino does not need an extra board or piece to load new code.
- Arduino can read analog and digital input signals.
- The hardware and software platform is easy to use and implement.

The Hardware:



Using the above image as a reference, the labeled components of the board respectively are-

1. **USB:** can be used for both power and communication with the IDE
2. **Barrel Jack:** used for power supply
3. **Voltage Regulator:** regulates and stabilizes the input and output voltages
4. **Crystal Oscillator:** keeps track of time and regulates processor frequency
5. **Reset Pin:** can be used to reset the Arduino Uno
6. **3.3V pin:** can be used as a 3.3V output
7. **5V pin:** can be used as a 5V output
8. **GND pin:** can be used to ground the circuit
9. **Vin pin:** can be used to supply power to the board
10. **Analog pins(A0-A5):** can be used to read analog signals to the board
11. **Microcontroller(ATMega328):** the processing and logical unit of the board
12. **ICSP pin:** a programming header on the board also called SPI
13. **Power indicator LED:** indicates the power status of the board
14. **RX and TX LEDs:** receive(RX) and transmit(TX) LEDs, blink when sending or receiving serial data respectively
15. **Digital I/O pins:** 14 pins capable of reading and outputting digital signals; 6 of these pins are also capable of PWM
16. **AREF pins:** can be used to set an external reference voltage as the upper limit for the analog pins
17. **Reset button:** can be used to reset the board

Purpose:

The Arduino is used for various purposes, such as:

- Finger button
- Button for motor activation
- Light as a sensors
- LED button
- Designing
- The Building of electronic devices

Application areas:

Smart Homes

With Arduino boards we can control the home activities with the control systems such as motion sensors, outlet control, temperature sensors, blower control, garage door control, air flow control, sprinkler control and bill of materials.

Defence

With RADAR (Radio Detection and Ranging) is a radio wavesbased object-detection system that can find out the range, altitude, direction, or speed of objects. A Radar can have different sizes and have different performance specifications.

It be used for air-traffic control at airports, long range surveillance and early-warning systems in ships. This system is the heart of a missile guidance system. In war a number of small portable radar systems are maintained and operated, as well as systems that occupy several large rooms.

Industries

Due to the easy programming environment, signal types, and easy adaptation in new set up, Arduino is used in Many industries. Arduino boards are low cost, and flexible alternatives to the usual industrial devices for adding remote control and monitoring functionality to small legacy industrial systems. With the growth of wireless technologies such as WiFi and cloud services in few past years, the wireless systems become monotonous in our daily life.

Traffic Signal Control

Today Arduino is used for the control of traffic lights, it can also be used for the real time control system with programmable timings, pedestrian lighting etc. In traffic control system the junction timing adjusts automatically to accommodate movement of vehicles smoothly avoiding waiting time at junction.

Medical

An Arduino based heartbeat monitor counts the number of heartbeats in a minute. In this a heartbeat sensor module is attached that senses the heartbeat upon putting a finger on the sensor. Arduino is used for designing many medical equipment's such as customizable Breathalyzer, little automatic slipper foot massager, Open source EEG/ECG/EMG, Thermometer, WI-Fi Body Scale with Arduino Board etc.

Laboratories

In laboratory for the designing and learning circuit designing Arduino provides a useful platform. There may be chances of some damages or any something wrong by the beginners and it may also be costly for the students to use new electronic parts. In this Arduino Simulator offers a solution to these problems, no damage done to your components, no money spends on hardware, faster circuit prototyping and no mess with cabling at all. Arduino based automated slide movement microscope is very cost-effective laboratory device.

Body Control with Arduino

With There are many body control devices with Arduino used for health care such as handSight gloves, breathalyzer microphone, heart rate monitoring system etc. Arduino based heart rate monitor is more advanced than a simply measure a user's heart rate. Our heart rate monitors talks! Each button gives a verbal description of its functionality and makes the measurements visible on the screen. This monitor will save the last four readings, display them, average them, and also offer some inspirational quotes. This sensor is used for fever, hypothermia, and activity levels and patterns detections. This device can sense the facial expressions. With the help of this Arduino device we can find out breathing rate, breathing depth, activity level and arousal level. For the movement monitoring we use this Arduino device, it can detect the occurrence of muscle contractions and strength of muscle contractions. We can check organ health and arousal and it can be used for diagnostic for medical intervention and enhance social interaction .

Electroddermal Activity: Another name of this sensor is Galvanic Skin response (GSR), used for emotional and physical arousal. Eye trackers are utilized as a part of exploration on the visual framework, in brain research, in psycholinguistics, showcasing, as an information gadget for human PC cooperation, and in item plan. In HandSight sensors are used to sense lights, find distance of physical objects. In the breathalyzer microphone there is collection of blood-alcohol content level data sets. It can also easily be adapted for karaoke.

Aerospace

Classical control theory to an airplane flap model and integration of RC vehicles in a Robotic Arena.

Automatic Vehicle Control

In this paper a test bed infrastructure designed for vehicle driving that includes control-system experimentation, strategies, and sensors. These four facilities and instruments.

4.6 Difference between Raspberry pi and Arduiano

Basis	Arduino	Raspberry Pi
License	Arduino is an open-source project. Both its software and hardware design are open source.	Both hardware and software of Raspberry Pi are closed source.
Control Unit	From Atmega Family	From ARM Family
Clock Frequency	16 MHz (Arduino UNO)	Up to 1.5 GHz in Raspberry Pi 4 B
RAM	Requires less RAM (2kB)	Requires large RAM (more than 1 GB)
CPU Architecture	8-bit	64-bit
Logic level	Arduino’s logic level is 5V.	Raspberry Pi’s logic level is 3V.
Power Consumption	Consumes about 200 MW of power	Consumes about 700 MW of power
Based on	Arduino is a Microcontroller	Raspberry Pi is based on a microprocessor
Hardware Structure	Simple hardware structure	Complex hardware Structure

Basis	Arduino	Raspberry Pi
Software	Arduino boards are programmable using C/C++ languages.	Raspberry Pi supports its own Linux-based operating system Raspberry Pi OS. You can also install the OS you like.
Internet	Arduino does not have internet support. You need additional modules or shields to connect it to the internet.	Raspberry Pi has a built-in Ethernet port and WiFi support.
Cost	Arduino boards are cheaper.	Raspberry Pi boards are expensive.
How they handle power drop	Arduino devices begin executing code when they are turned on. Therefore, when power is turned off, abruptly, you won't end up with a corrupt operating system or errors. The code will simply start again when plugged in.	Raspberry Pi requires the same care as a PC. You have to shut the operating system down properly.
Current drive strength	Higher current drive strength	Lower current drive strength
Capability	Arduino is generally used to perform single (and simple) tasks repeatedly.	Raspberry Pi can perform multiple tasks simultaneously.
Wireless connectivity	Arduino does not support Bluetooth or WiFi.	Raspberry Pi supports Bluetooth and WiFi.
Applications	Traffic light countdown timer, Parking lot counter, Weighing machines, etc.	Robot controller, Game servers, Stop motion cameras, etc.

Pros and Cons of Raspberry Pi

Pros

- Since it supports an operating system, It can perform complex operations like Weather monitoring, Controlling robots, etc.
- You can use it as a portable computer because it has everything- from CPU (Central Processing Unit) to ethernet port and WiFi support.
- It has a large number of GPIO (General-Purpose Input/Output) pins (the famous model of Raspberry Pi has 40 GPIO pins). Therefore, it can support a large number of sensors.
- It has superior processing power. The 4 B variant of Raspberry Pi comes with a 1.6 GHz processor.
- It can run all kinds of applications (including MS Office and Email).

Cons

- Raspberry Pi"s hardware and software are closed-source. It means that you cannot customize your own Raspberry Pi single-board computer (SBC).
- Raspberry Pi does not have any internal storage, it requires a micro SD card to work as internal storage.
- It sometimes overheats during heavy operations.

Pros and Cons of Arduino

Pros

- Both hardware and software of Arduino are open-source. You have the liberty to select from the codes already available or you can customize your own Arduino board.
- It is less expensive than Raspberry Pi.
- It is good for beginners as it is easy to learn and use.
- It is quite easy to program Audrino through IDE (Integrated Development Environment).
- Arduino has a huge community and a wide range of applications.

Cons

- It has very less processing power when compared to Raspberry Pi.
- Arduino boards do not support internet and wireless connectivity.
- 8-bit CPU architecture
- Arduino is incapable of performing complex tasks.