(Smart Home Automation and Smart Gardening system)

Major Project Report

submitted

in partial fulfilment

for the award of the Degree of

Bachelor of Technology
in Department of Computer Science &Engineering
(with specialization in Computer Science & Engineering)

Supervisor:
Mukesh Choudhary

Submitted By:
Parul Rateshwar
17ERECS047
Shekhar Meena
17ERECS064
Snigdha Verma
17ERECS067

Department of Computer Science & Engineering

Rajasthan Institute of Engineering & Technology, Jaipur

Rajasthan Technical University, Kota

25th June 2021

Candidate's Declaration

We hereby declare that the work, which is being presented in the Major Project, entitled **"Smart Home Automation and Smart Gardening System"** in partial fulfilment for the award of Degree of "*Bachelor* of Technology" in Dept. Of Computer Science & Engineering **submitted to the Department of Computer Science & Engineering,** Rajasthan Institute of Engineering and Technology, Jaipur Rajasthan Technical University is a record of our own investigations carried under the Guidance of Mr. Mukesh Choudhary, Department of Computer Science & Engineering, Rajasthan Institute of Engineering And Technology**.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Parul Rateshwar,

Shekhar Meena,

Snigdha Verma

B.Tech IV Year, VIII Semester

Enrolment No.: 17ERECS047,

17ERECS064 ,

17ERECS067

Rajasthan Institute of Engineering and Technology**.**

Counter Signed by

Mukesh Choudhary

....

....

Acknowledgement

We take this opportunity to express our gratitude to all those people who have been directly and indirectly with us during the competition of this project.

We pay thank to MR. MUKESH CHOUDHARY who has given guidance and a light to us during this major project. His versatile knowledge about "INTERNET OF THINGS and EMBEDDED SYSTEMS" has eased us in the critical times during the span of this project.

We acknowledge here out debt to those who contributed significantly to one or more steps. We take full responsibility for any remaining sins of omission and commission.

Parul Rateshwar,

Shekhar Meena,

Snigdha Verma

B.Tech IV Year

(Computer Science & Engineering)

Abstract

Home automation system achieved great popularity in the last decades and it increases the comfort and quality of life. This report is an overview of current and emerging home automation systems. Nowadays most home automation systems consist of a smartphone and microcontroller. A smart phone application is used to control and monitor the home appliances using different type of communication techniques. In this report the working principle of different type of wireless communication techniques such as Wi-Fi and 6LowWPAN are studied and their features. In this project work the survey of home automation systems is discussed and their advantages and drawbacks are also highlighted.

Due to the rapid development in the field of the Automation industry, human life is becoming more advanced and better in all aspects. In the present scenario, automated systems are being preferred over the non-automated system. With the rapid growth in the number of consumers using the internet over the past years, the Internet has become an important part of life, and IoT is the newest and emerging internet technology. Internet of things plays an important role in human life as well as in the educational field because they are able to provide information and complete the given tasks while we are busy doing some other work. In this report, a prototype and implementation of Smart Home Automation and smart gardening system with Wi-Fi technology are demonstrated. ESP8266 is used as a Wi-Fi technology. The proposed system consist of a hardware interface and software interface. In the hardware interface, the integration of ESP8266 Wi-Fi technology for controlling home appliances and sensors is manifested, and an application is provided for controlling to multiple users of home, with smart phones, tablets, and laptops. This system is one of the best methods for controlling home devices with ease with multiple users and one of the best method for an energy management system. The access to the whole system is given by its admin only to different users. This system is also expandable for controlling various appliances used at home through sensors as long as it exists on Wi-Fi network coverage.

Internet of things (IoT) and Embedded Systems is an emerging technology today that envisions all objects around us as a part of Internet. Automation of the devices, appliances at home and office is having extensive possibility of research with the innovation of technology in communication. Misuse of power energy can be shortened by automating the devices and appliances. Mobile communication is plays a major role in the automation domain. Android phones are designed with applications to automate the required devices. Creating a mobile app on a Smartphone device so that the user can control electronic devices; see the amount of flow, so the problem is the difficulty in saving electricity can be resolved. IoT includes variety of objects like smart phones, tablets, digital cameras and different sensors. When all these devices are connected together, they enable additional smart processes and services that support our basic needs, environment and health. Massive number of devices connected to internet provides

enormous kinds of services and also produces huge amount of data and information. Cloud computing is an on-demand access to a shared pool of configurable resources that can be provisioned as infrastructures, software and applications. Cloud based platforms help to connect to the things around us so it is easy to connect any objects anytime at any place using in built applications. Applications that use devices like sensors have unique requirements of enormous storage to store big data and huge computation power to enable the real time processing of the data, information and high speed network is required for audio and video streaming. Wireless technologies like Bluetooth, WiFi, and ZigBee have been used in wide range of applications like modern home security systems using low cost, low power, less complexity RF module.

This report discusses the usage of wireless sensor networks in irrigation control by a smart plant monitoring and watering system in which the irrigation process is controlled by valves. Water supplies became pollute and scarce. During the last years, the necessity to use water resources such as rivers, ground water and rain water efficiently has increased rapidly. Population growth, rapid urbanization and climate change have stressed water resources on the planet. The application of wireless soil-moisture sensor networks to detect water content in the soil can utilize water resources very efficiently. Water requirements depend on the type of plants and the soil as well as the season. A study has been made on the clay soil to observe its behavior and its different characteristics. By this study the time of excitation of the sensor could be known and the period of irrigation could be detected. This will be more efficient in terms of the time in which the sensor will be excited and the quantity of water that will be used.

Automation of farm activities can transform agricultural domain from being manual into a dynamic field to yield higher production with less human intervention. The project contains Green garden is developed which can be used to manage farms using modern information and communication technologies. It focuses on developing an automatic irrigation system which monitors soil factors that are essential for luxuriant growth of a plant via a controller.

Several technological advancements have been made over the years, more recently the invention of Internet of Things (IOT). IOT enables connection and interaction of things on a daily basis, this involves automation processes that require little or no human intervention to perform various tasks with sensors, RFID, actuators and microcontrollers among others. Gardening is vital to human existence as it helps sustain lives by providing good ecosystem. This report provides the potential benefits of Smart Garden Automation through the technology of IOT for efficient plantation. This study shows the components required for Smart Home Automation and Smart Gardening System with the existing systems, such as microcontroller and sensors which is linked to a controller and connected via the internet with live feeds of event statistics being displayed through graphs on the controller’s Dashboard.

Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Internet of Things (IOT)

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.

A thing in the internet of things can be a person with a heart monitor implant, a farm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an Internet Protocol (IP) address and is able to transfer data over a network.

Increasingly, organizations in a variety of industries are using IoT to operate more efficiently, better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

The Internet of Things (IoT) describes the network of physical objects—“things”—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. These devices range from ordinary household objects to sophisticated industrial tools. With more than 7 billion connected IoT devices today, experts are expecting this number to grow to 10 billion by 2020 and 22 billion by 2025.

The Internet of Things, or "IoT" for short, is about extending the power of the internet beyond computers and smartphones to a whole range of other things, processes, and environments.

The Internet of Things (IoT) is at the epicenter of the Digital Transformation Revolution that is changing the shape of business, enterprise and people's lives. This transformation influences everything from how we manage and operate our homes to automating processes across nearly all industries. But what is IoT, actually? In this chapter, We are going to share everything you need to know about the Internet of Things.

"The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction."

The Internet of Things means taking all the things in the world and connecting them to the internet**.**

When something is connected to the internet, that means that it can send information or receive information, or both. This ability to send and/or receive information makes things smart, and smarter is better.

Let's use smartphones again as an example. You can listen to any song in the world, but not because your phone has every song stored on it. It's because every song in the world is stored somewhere else (that place is known as "the cloud"), and your phone can request a song, and receive information to stream it.

To be smart, a thing doesn't need to have super storage or a supercomputer inside of it. All a thing has to do is connect to super storage or to a supercomputer.

The internet of things (IoT) is a catch-all term for the growing number of electronics that aren't traditional computing devices, but are connected to the internet to send data, receive instructions or both.

There's an incredibly broad range of things that fall under that umbrella: Internet-connected "smart" versions of traditional appliances like refrigerators and light bulbs; gadgets that could only exist in an internet-enabled world like Alexa-style digital assistants; internet-enabled

The IoT brings the power of the internet, data processing and analytics to the real world of physical objects. For consumers, this means interacting with the global information network without the intermediary of a keyboard and screen; many of their everyday objects and appliances can take instructions from that network with minimal human intervention.

In the most general terms, the Internet of Things includes any object – or "thing" – that can be connected to an Internet network, from factory equipment and cars to mobile devices and smart watches. But today, the IoT has more specifically come to mean connected things that are equipped with sensors, software, and other technologies that allow them to transmit and receive data – to and from other things. Traditionally, connectivity was achieved mainly via Wi-Fi, whereas today 5G and other types of network platforms are increasingly able to handle large datasets with speed and reliability.

Of course, the whole purpose of gathering data is not merely to have it but to use it. Once IoT devices collect and transmit data, the ultimate point is to analyse it and create an informed action.

1.2 History of Internet of Things team work

The term Internet of Things is 16 years old. But the actual idea of connected devices had been around longer, at least since the 70s. Back then, the idea was often called "embedded internet" or "pervasive computing". But the actual term "Internet of Things" was coined by Kevin Ashton in 1999 during his work at Procter & Gamble. Ashton who was working in supply chain optimization, wanted to attract senior management's attention to a new exciting technology called RFID. Because the internet was the hottest new trend in 1999 and because it somehow made sense, he called his presentation "Internet of Things".

Even though Kevin grabbed the interest of some P&G executives, the term Internet of Things did not get widespread attention for the next 10 years.

The concept of IoT started to gain some popularity in the summer of 2010. Information leaked that Google's StreetView service had not only made 360 degree pictures but had also stored tons of data of people's Wifi networks. People were debating whether this was the start of a new Google strategy to not only index the internet but also index the physical world.

IoT (Internet of Thing) is extension of Internet services (all the devices are connected and controlled with internet connectivity).IoT is emerging topic in industry and engineering circles. Internet has changed human's life by providing anytime, anywhere connectivity with anyone. Internet helps us to bring in with immediate solution for many problems and also able to connect from any of the remote places which contributes to overall cost reduction and energy consumption. Every day the modern people expect new device and new technology to simplify their day to day life. The innovators and researchers are always trying to find new things. To satisfy the people but the process is still infinite. In 1990s, Internet connectivity began to proliferate in enterprise and consumer market, but was still limited in its use because of the low performance of the network interconnects. In the 2000s internet connectivity became the norm for many applications and today is expected as part of many enterprise, industrial and consumer products to provide access to information. However, these devices are still primarily things on the internet that require more human interaction and monitoring through apps and interfaces. For many years Home automation is mainly used as features of science fiction writing, but it's become practical since the early of 20th century that is because of the introduction of electricity and rapid improvement in information technology.

Nowadays, automation system has been increased. It is because by using automation less electricity is consumed, wastage is decreased and energy is saved. Automation is known as home automation system if it is implemented at home. Green house technology was invented by many researchers with the help of automation. It is called as automation system or home watering system. Automation system is proposed to control the system and plants growth is observed with respect to moisture of the soil. The mechanism provides appropriate condition to growing of plants like herbs, floweret that creates the negative effect which is caused by the change in environment. Efficient water is used by this irrigation system. Most of the home gardens that are in the village area are not under roof that can be showed to the change in weather like superhot day or heavy rainy day. The various changes in weather may affect the growth of the plant in the mini garden. The owner will observe the garden that is growing healthily. If the home owner is away from the plant for a long time it is difficult for the owner to observe the plants. We need on system to control and monitor the plants. Smart irrigation does not mean only consuming less water. It provides water supply to the plants based on its need. In smart irrigation system monitoring the soil moisture is one of the important features. Based on this the owner can know that sufficient water is provided to the plants or not for its growth. The growth of the plant is not possible without sufficient water. we can say that the system is smart if it is monitoring continuously the condition of the plant.

As the world is trending into modern technologies and implementations it is a necessary goal to irrigate home garden time to time and in efficient manner, to give required sunrays and avoid extra warm sunrays to enter in garden by light detector, to monitor this all system on IOT far distances communication also. Many researches are working in the field of irrigation. Most projects signify the use of wireless sensor network collect data from different sensors deployed at various nodes and send it through the wireless protocol. The collected data provide the information about the various environmental factors. Monitoring the environmental factors is not the complete solution to increase the yield of garden trees. There are number of other factors that decrease the productivity to a greater extent. Hence automation must be implemented in irrigating fields to overcome these problems. So, to provide solution to all such problems, it is necessary to develop an integrated system which will take care of watering the home garden. But complete automation in irrigation is not achieved due to various issues. So we are implementing a full proof system which will protect our garden plants from overwatering and over direct sunlight. Proposed system contains a temperature sensor and light sensor to manage shelter opening and closing according to sunlight, Moisture sensor for automatic irrigation of plants according to soil dampness.

1.3 How Internet of Things works

An IoT ecosystem consists of web-enabled smart devices that use embedded systems, such as processors, sensors and communication hardware, to collect, send and act on data they acquire from their environments. IoT devices share the sensor data they collect by connecting to an IoT gateway or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices -- for instance, to set them up, give them instructions or access the data.

When talking about how does IoT work, the process begins with devices that have built-in sensors. These devices are connected to IoT platforms which stores data from all the connected devices. The important data is then used to perform tasks that fulfil the needs of people.

When we say the data is stored in the IoT platforms, it doesn’t mean that all the data is useful. Devices carefully select only particular data that is relevant to execute an action. These pieces of information can detect patterns, recommendations and problems before they occur.

In this way, IoT application works with smart systems that automate tasks to address specific needs.

1.4 Importance of Internet of Things

The importance of IoT is its influence. The Industrial IoT goes beyond the simple exchange of data by connecting devices, systems and people in intelligent, real-world applications to enable environments such as automated factories, smart cities and connected healthcare.

The Industrial IoT is defined as the automated interaction between smart devices and systems, exchanging data to an offsite or cloud-based solution for meaningful, time-sensitive analytics using components with very low energy consumption, that are simple to install and based on industry standards. Traditional embedded systems are at the heart of this phenomenon, having evolved from standalone systems to become a network of connected devices and systems.

IoT is regarded as the significant frontier that can improve almost all activities in our lives. Most of the devices, which have not previously been connected to the internet, can be networked and respond the same way as smart devices. By 2020, the world is set to be completely IoT oriented. Here are the benefits, which come with this technology.

- IoT promotes efficient resource utilization.
- It minimizes human efforts in many life aspects.
- Enabling IoT will reduce the cost of production and maximizing the returns
- It makes analytics decisions faster and accurately
- It boosts the real-time marketing of products
- Provide a better client experience
- It guarantees high-quality data and secured processing

Considering the complex ecosystem of IoT, there is a need of underlining the competitive advantage of IoT and the stakeholders enabling the users to continue being in complete control of safely sharing their data and depend as much as they can their content.

1.5 Embedded System and History

An embedded system combines mechanical, electrical, and chemical components along with a computer, hidden inside, to perform a single dedicated purpose. There are more computers on this planet than there are people, and most of these computers are single-chip microcontrollers that are the brains of an embedded system. Embedded systems are a ubiquitous component of our everyday lives. We interact with hundreds of tiny computers every day that are embedded into our houses, our cars, our bridges, our toys, and our work. As our world has become more complex, so have the capabilities of the microcontrollers embedded into our devices. Therefore the world needs a trained workforce to develop and manage products based on embedded microcontrollers.

As its name suggests, Embedded means something that is attached to another thing. An embedded system can be thought of as a computer hardware system having software embedded in it. An embedded system can be an independent system or it can be a part of a large system. An embedded system is a microcontroller or microprocessor based system which is designed to perform a specific task. For example, a fire alarm is an embedded system; it will sense only smoke.

An embedded system has three components −

- It has hardware.
- It has application software.
- It has Real Time Operating system (RTOS) that supervises the application software and provide mechanism to let the processor run a process as per scheduling by following a plan to control the latencies. RTOS defines the way the system works. It sets the rules during the execution of application program. A small scale embedded system may not have RTOS.

An **embedded system** is a computer system—a combination of a computer processor, computer memory, and input/output peripheral devices—that has a dedicated function within a larger mechanical or electronic system. It is *embedded* as part of a complete device often including electrical or electronic hardware and mechanical parts. Because an embedded system typically controls physical operations of the machine that it is embedded within, it often has real-time computing constraints. Embedded systems control many devices in common use today.

In 2009 it was estimated that ninety-eight percent of all microprocessors manufactured were used in embedded systems.

1.6 Team Work

1.6.1 Parul Rateshwar (17ERECS047)

- **Coding of MCU (Sensor) :**

  First of all, we have defined two variables; one for the soil moisture sensor pin and the other for storing the output of the sensor.

  In the setup function, the "Serial.begin(9600)" command will help in communication between the MCU and serial monitor. Then, we will print the "Reading From the Sensor …" on the serial monitor.

  In the loop function, we will read from the sensor analog pin and will store the values in the "output_ value" variable. Then, we will map the output values to 0-100, because the moisture is measured in percentage. When we took the readings from the dry soil, then the sensor value was 550 and in the wet soil, the sensor value was 10. So, we mapped these values to get the moisture. After that, we printed these values on the serial monitor.

- **Controlling of Real-Time Graph**

  Adafruit IO is used in consideration to cut down cost of additional LCD display, as well as it is convenient to control and monitor process anywhere anytime.

  It works as follows :

  Connect to your WiFi network.

  Connect to Adafruit via Wifi.

Read data from Sensor.

Upload Sensor Data to Adafruit.

And then it is represented via a real-time graph.

- **Debugging of Soil Moisture Sensor Code :**

  Measurements are made in air

  Factory calibration and your soil type

  VWC, RWC, PAW, RAW and ε

  Temperature can affect your measurements

  Salinity and electrical conductivity

- **Circuit Designing**

  In this project, a simple garden monitoring system based on Internet of Things is designed. An IoT device built on ESP8266 and equipped with sensors like DHT-11 Temperature and Humidity sensor, Moisture sensor The **ESP8266** Wi-Fi modem is interfaced The device is also interfaced with a water pump controlled by L293D motor driver IC. The water pump is automatically controlled based on the values of the various environmental factors. The various environmental factors like temperature, humidity, soil moisture and light intensity can be monitored on the Adafruitdashboard.io dashboard. to connect with a Wi-Fi hotspot.

- **Considering Protocols**

  Home automation technology witnessed a huge leap forward.

  Products like Amazon Echo and Google Home were finally able to break into the mainstream consumer space and start an industry trend.

  Consequently, it is expected to continue this trend with technologies like facial recognition, voice commands and biometrics finding wider application in smart home solutions.

  Leading tech companies are investing huge amounts in R&D to create their own connected home and office ecosystems.

The ultimate vision is to create completely automated systems which can recognize and respond to select individuals or user groups, within a home or office environment.

A prototype has been implemented to show the real illustration of the proposed approach.

An android mobile application has been developed to display the real-time profiles of environmental factors like soil, moisture, and humidity.

With the help of this system, users will be able to treat their gardens in a better way in terms of plant health and growth.

This report work replaces the need for gardeners and issues faced during the maintenance of gardens in big cities.

The purpose of this research is to introduce and prosper the IoT innovation towards smart cities in our society.

- **Including Libraries for MCU (Sensor)**

MQTT Adafruit library:

MQTT stands for Message Queuing Telemetry Transport. MQTT is a machine to machine internet of things connectivity protocol. It is an extremely lightweight and publish-subscribe messaging transport protocol. This protocol is useful for the connection with the remote location where the bandwidth is a premium. These characteristics make it useful in various situations, including constant environment such as for communication machine to machine and internet of things contexts. It is a publish and subscribe system where we can publish and receive the messages as a client. It makes it easy for communication between multiple devices. It is a simple messaging protocol designed for the constrained devices and with low bandwidth, so it's a perfect solution for the internet of things applications.

- **Testing the MCU (Sensor) Code**

  Code is tested with random values and figures in order to check whether any error occurs or not.

1.6.2 Shekhar Meena (17ERECS064)

- **Circuit & Designing**

  Upload the Arduino code using the cable and connect the USB to a power bank or a port for power. Connected the USB in the diagram also in a port or power bank for power. No need to supply separate power to the bolt module.

  After uploading the code check if all devices working properly including LCD, KeyPad, LDR, temperature sensor, and LED array. Don't connect heavy load in place of theLED array as it may damage the transistor. Use a MOSFET or Motor driver circuit instead. Check if serialBegin command is working properly by clicking on serial monitor tab in Arduino interface.

  Next check the bolt connections, relay, micro USB, motor and SPST switch. Use spst switch instead of a sensor.

- **Internet Connectivity**

  The **concept of Home Automation** aims to bring the control of operating your every day home electrical appliances to the tip of your finger, thus giving user affordable lighting solutions, better energy conservation with optimum use of energy. Apart from just lighting solutions, the concept also further extends to have a overall control over your home security as well as build a centralised home entertainment system and much more. The **Internet of Things** (or commonly referred to as IoT) based Home Automation system, as the name suggests aims to control all the devices of your smart home through internet protocols or cloud based computing.

The IoT based Home Automation system offer a lot of flexibility over the wired systems s it comes with various advantages like ease-of-use, ease-of-installation, avoid complexity of running through wires or loose electrical connections, easy fault detection and triggering and above and all it even offers easy mobility.

- **ESP Connection to Sensor**

  Connect the two pins of the moisture sensor to the two pins on the Amplifier circuit using jumper wires. Connect the Vcc from the Amplifier to the 3.3V pin on the NodeMCU. Connect the GND pin to the ground (GND) pin on the NodeMCU. Connect the Analog pin to the A0 pin on the NodeMCU.

- **Stepping Down Voltage by Buck Converter**

  Buck converter is a step-down power supply module. It is used to bring down the power supply to the desired voltage in order to safe guard the circuit.
  Voltage is stepped down to 5v.

- **Relay Connection**

  Connect **NodeMCU** Ground (**GND**) pin to **-ve** pin of **Relay**.

  Connect **NodeMCU** Supply (**3v3**) pin to **+ve** pin of **Relay**.

  Connect **NodeMCU** Digital pin (**D4**) to **Input** pin of **Relay**.

- **Hardware Debugging**

  Connecting the ESP8266 Board to the computer via USB connector.

With the use of the tester, we check the regulated power supply.

Confirming that the supply is less than or equal to the permissible limit.

- **Extra:**

  We can also run a test code to check the proper functionality of the board.

- **Note :**

  Do check that the LEDs on the board are functioning properly.

- **AC Supply**

  For Home appliances 220v AC Supply is connected. It is directly connected to the relay circuit.

1.6.3 Snigdha Verma (17ERECS067)

- **Circuit Designing**

  The most basic and crucial requirement in a home automation system, the interface is the basic communication protocol and hardware combination used for sending and receiving messages between devices and the user. Designers have many options for executing communication between devices, the user, and the overall system, depending upon the system, range, size of house, ease of use, etc. If a user wants to control the home appliances through the Internet, the designer needs to add an Ethernet/Wi-Fi interface to connect the system to the home network. If the user wants to control the system using Bluetooth from a cell phone, the designer needs to add a Bluetooth interface to communicate with the device.

  The designer needs to determine the sensing requirements of the user and decide upon the required sensor to perform the task. He or she also needs to assess the sensor specifications required for different needs and usability in different environments.

- **Coding ESP**

  The ESP8266 is a low-cost WiFi module built by Espressif Systems. Its popularity has been growing among the hardware community thanks to it's nice features and stability, to the point that it can be easily programmed using your Arduino IDE.

  Requirements:

  An ESP8266 module.

  Arduino UNO, UARTbee or any UART to the USB device.

  This module has powerful onboard processing and storage capability that allows it to be integrated with the sensors and other application through its GPIOs.

  Controlling relay:

  Relays come in several form factors. Basically, a relay will be a rectangular block with at least 4 electrical connectors exposed.

  When you activate the relay by applying an appropriate voltage to the control circuit, the electromagnetic coil in the relay activates and pulls the switch open, stopping any current flowing through the switched circuit.

- **Debugging of ESP Code**

  The error messages are displayed in predefined codes. Unless you know the code, you can never understand what it says. ESP8266 error messages can be categorized into 3 types –

  1) Reset causes, 2) Boot modes, and 3) Fatal Exceptions.

  Though some can argue that boot mode is not exactly an "error message", but sometimes it really helps to diagnose a runtime error.

  Note 1: Just enabling Debug port and setting Debug level won't wok, You need to upload you code after enabling these settings.

  Note 2: Debug port work's at 115200 BAUD if you have not set the serial.

- **Architecture**

  Home automation refers to the ability of your home to make its own decisions depending on environment conditions and give you the option to control it from a remote location. The heart project is the WiFi enabled board that needs no introduction; the ESP8266 based **NodeMCU** development board. It is an open source platform for developing WiFi based embedded systems and it is based on the popular ESP8266 WiFi Module.

  NodeMCU was born out of the desire to overcome the limitations associated with the first versions of the ESP8266 module which was not compatible with breadboards, it was difficult to power and even more difficult to program. The NodeMCU board is easy to use, low cost and that quickly endeared it to the heart of makers and it is one of the most popular boards today.

  We will add a 2-channel relay module to the ESP8266 board. The project flow involves the control of NodeMCU's GPIOs from a webpage on any device connected on the same network as the board. The status of the GPIOs control the coils of the relays and that causes the relay to alternate between normally open (NO) and normally closed (NC) condition depending on the state of the GPIO, thus, effectively turning the connected appliance "ON" or "OFF".

- **Including Libraries for MCU**

  ESP8266WiFi library:

  The Wi-Fi library for ESP8266 has been developed based on ESP8266 SDK, using the naming conventions and overall functionality philosophy of the Arduino WiFi library. Over time, the wealth of Wi-Fi features ported from ESP8266 SDK to esp8266 / Arduino outgrew Arduino WiFi library and it became apparent that we would need to provide separate documentation on what is new and extra.

  This documentation will walk you through several classes, methods and properties of the ESP8266WiFi library. If you are new to C++ and Arduino, don't worry. We will start from general concepts and then move to detailed description of members of each particular class including usage examples.

- **Testing the MCU Code**

  Connecting the ESP8266 Board to the computer via USB connector.

  With the use of the tester, we check the regulated power supply.

  Confirming that the supply is less than or equal to the permissible limit.
- **Extra:**

  We can also run a test code to check the proper functionality of the board.
- **Note :**

  Do check that the LEDs on the board are functioning properly.

- **AC Supply**

  For Home appliances 220v AC Supply is connected. It is directly connected to the relay circuit.

- **Controlling of Appliance**

  The controller is the device that receives the sensors' signals, processes them and makes computations on them, and then sends instruction signals to the actuators

  Dashboard: The user can display data from a temperature-humidity sensor alongside data from an air quality sensor and add a button to turn on the appliances in the house.

  Triggers: Use triggers in Adafruit IO to control and react to your data. Configure triggers to email you when your system goes offline, react to a temperature sensor getting too hot, and publish a message to a new feed.

It can :

Display your data in real-time, online

Make your project internet-connected: Control motors, read sensor data, and more!

Connect projects to web services like Twitter, RSS feeds, weather services, etc.

Connect your project to other internet-enabled devices

The best part? All of the above is do-able for free with Adafruit IO

Chapter 2

LITERATURE SURVEY

Many practical developments of considerable interests in the field of Internet of Things and Embedded Systems need continuous and uniform review of relevant data science theories. Behalf of this many programmers have addressed literature survey of given practical as well as theoretical aspects. Although all literature has witnessed that programming and Hardware Connectivity is a complex task. The task is even more challenging for introductory modules. Failure rates are not marginal One of the common problems shared by Computer Science Departments is the lack of basic programming skills reported by module leaders of courses following first programming courses and how to equip students with better programming skills and Hardware Implementation after the introductory courses. First these courses typically emphasize on the principles of automated software, hardware and design from the very beginning. An alternative approach is based on starting with a more traditional procedural approach first. The evidence from past research seems to suggest that Internet of Things is generally more complex than the other technologies as it consists of a mesh of different components by using qualitative analysis.

It is therefore suggested to introduce Internet of Things and Embedded Systems for the basic aspects of course. The use of IOT is expected to both reduce the Human intervention and Cost of connecting different appliances. In other words, programmers are able to devote particular attention to Internet of things, Embedded Systems, Architecture of IOT, Protocols involved in IOT and Embedded Systems, Data Collection etc. The aim is to evaluate the success of using Internet of Things and Embedded Systems to introduce basic concepts.

The major implementations of Internet of Things and Embedded Systems in the current IT industries are in Typical IoT Platform-enabled applications, Transportation / Mobility, OmniBus fleet operations optimization, Energy, retail and Surveillance, GUI based implementations, etc.

The IOT module involves various Technologies such as GSM based home automation system using cell phones: Because of the mobile phone and GSM technology, the GSM based home automation is lure to research. The SMS based home automation, GPRS based home automation and dual tone multi frequency (DTMF) based home automation, these options we considered mainly for communication in GSM. It shows how the home sensors and devices interact with the home network and communicates through GSM and SIM (subscriber identity module). The system use transducer which convert machine function into electrical signals which goes into microcontroller. The sensors of system convert the physical qualities like sound, temperature and humidity into some other quantity like voltage. The microcontroller analysis all signal and convert them into command to understand by GSM module. Select appropriate communication method among SMS, GPRS and DTFC based on the command which received GSM module.

Wi-Fi based home automation system using cell phones: Wi-Fi based home automation system mainly consist three modules, the server, the hardware interface module, and the software package. In the system model layout, Wi-Fi technology is used by server, and hardware Interface module to communicate with each other. The same technology uses to login to the server web based application. The server is connected to the internet, so remote users can access server web based application through the internet using compatible web browser. Software of the latest home automation system is split to server application software, and Microcontroller (Arduino) firmware. The Arduino software, built using C language, using IDE comes with the microcontroller itself. Arduino software is culpable for gathering events from connected sensors, and then applies action to actuators and pre programmed in the server. Another job is to report and record the history in the server DB. The server application software package for the proposed home automation system is a web based application built using asp.net. The server application software can be accessed from internal network or from internet if the server has real IP on the internet using any internet navigator supports asp.net technology. Server application software is culpable of, maintain the whole home automation system, setup, configuration. Server use database to keep log of home automation system components, we choose to use XML files to save system log.

Home automation using Android Controller: The devices of home are associated to the Controller and the Connection is established between the Android device and Controller. The devices of house are link to the input/output ports of the board (EMBEDDED SYSTEM) and their current situation will have passed to the Controller. The microcontroller board (Arduino Controller) is based on the ESP-8266. It has a USB host connection to associate with Android based phones. The two important features of Android Open Accessory Protocol 2.0(AOAP) are as follows: It has output that is from the Android device to the component and it also support for the component serves as one or more Human Interface Devices (HID) to the Android device. It depends upon Android and Arduino platform in which both are FOSS (Free Open Source Software). Including MCU and various others sensors for Gardening and Home automation systems and it will automatically notice the user through cell phone.

Wireless Home Automation system using IoT This system uses mobiles or computers to control basic home control and function automatically through internet from anywhere around the world globally, an automated home is sometimes called a smart home. It is meant to save the electric power and human energy. The proposed system is a distributed home automation system, consists of server i.e. Wi-Fi module, sensors. Server controls and monitors the various sensors, and can be easily configured to handle more hardware interface module (sensors). The Arduino board, with built in Wi-Fi module acts as web server. Automation System can be accessed from the web browser of any local PC using server IP, or remotely from any PC or mobile handheld device connected to the internet with appropriate web browser through server real IP (internet IP). Wi-Fi technology is selected to be the network infrastructure that connects server and the sensors. Wi-Fi is chosen to improve system security (by using secure Wi-Fi connection), and to increase system mobility and scalability.

A Survey on an Efficient IOT Based Smart Home and Smart Garden proposes an efficient implementation for IoT for monitoring and automation system and it uses the portable devices as a user interface. Portable devices can communicate with home automation network through an Internet gate, by means of low power communication protocols like Wi-Fi etc. This project aims at controlling home appliances and Garden via Smartphone using Wi-Fi as communication protocol and arduino uno and ESP- 8266. The user here will move directly with the system through a web-based interface over the web whereas home appliances like lights and a plant etc. are remotely controlled through easy website. This survey also describes how to provide fully smart environment condition monitoring by various sensors (MCU, Soil Moisture sensor, Light and Level) for providing necessary data to automatically detection and resolution of any problem in the devices. Controlling the home appliances via World Wide Web. Based on the idea of "A Survey on Internet of Things Based Home Automation System". A Remote Password Operated Home Appliances Control Project Algorithm was designed to read the data from Wi-fi module, Initialize the UART protocol and display the status of the electrical Plant Monitoring System. The system is installed beside the conventional electrical switches on the board. The risk of dangerous electric shocks can be avoided by using low voltage switches. The system uses a GUIs- either on the personal computer or on the other on Smartphone. The status of the appliances i.e. weather it is on/off can be known by using this GUI based Controller. Any changes in the status of the appliances, immediate intimation is shown on the GUI. The window GUI will act as a server to forward or transmit any data to/from the Smartphone and the main control board, after the Smartphone's Hotspot is connected to the Prototype. In case, the Wi-fi connection between the Prototype or Controller board fails, then connection can be reestablished by using USB cable. The user can monitor and control the devices from any remote location at any time using IOT. This Survey describes "Smart Home Automation embedded with Smart green Gardening System using IOT and Embedded systems ". A System hardware is divided into three parts i.e. Software, Hardware and Arduino controller. Relay, LPT port, transistor, diode resister are integrated on PCB. They have connected two devices to the PCB i.e. light and DC Pump. MCU soil Moisture sensor is connected to Arduino .It will sense humidity and Dryness of plant as well. Arduino and PCB are connected. Arduino and PCB will interact with each other through PC. They have measured Dryness, conductivity and humidity of plant. They have set time by which it continuously senses dryness. it continuously senses Dryness, Water- level and Turn on light after every 5 seconds.

This paper was an attempt to introduce ideas, movements, technical approaches and considerations for strategic planning for the Internet of Things. It was a simple review of research in IoT models, privacy issues and considerations for businesses to include in their way forward for IoT.

Attempts are being made to harness the seismic shift caused by the IoT movement. While unification or standardization is lacking it seems like it may be left to economic competition on what moves forward. This paper was a very small fraction of what IoT literature provides and was meant to educate individuals as it has done for the author in order to initiate discussions at their organizations. This survey describes The Internet of Things (IoT) and Embedded Systems are an atmosphere in which objects, animals or people are make available with distinct identifiers and the capability to transfer data through a network without requiring human-to-human or human-to-computer interaction and two or more systems could be joined together to get a complex prototype. 'Internet of Things' defines a number of skills and research disciplines that allow the Internet to reach into the real world of physical objects. Technologies like short-range wireless communications, RFID, ad hoc and wireless sensor networks (WSNs) which is the part of Internet of Things (IoT). The home automation embedded with smart and Green Garden system using Internet of Things, Embedded Systems and wireless sensor network work satisfactorily by connecting simple appliances to it and the appliances were successfully controlled remotely through internet. This paper describes the concept of WSN, IoT and architecture of Home Automation. Based on surveyed study the comparison of home automation systems Microcontroller, user interface, a communication interface and their performance factor are compared. There are a number of do-it-yourself (DIY) platforms available that allow to create Internet of Things (IoT) system quickly and easily with low cost and high performance e.g. Raspberry pi, Arduino, other microcontrollers, etc. In this review explained the home automation embedded with mart and Green Gardening system e.g. Web based, mobile-based and Internet based. In future Internet of Things (IoT) and Embedded Systems will be more smart and fast. It would be extended to the large-scale environment such as colleges, offices and factories etc.

It describe a design for home automation system using ready-to-use, cost effective and energy efficient devices including Wi-fi chip, Arduino microcontrollers, MCU modules and relay boards. Arduino microcontrollers are used to receive the on/off commands from the protocol. The use of Soil Moisture sensors and Real- Time Graph make a smart Gardening system. it presents a control design of wireless sensor network system based on MCU Module for greenhouse, which consists of some sensor nodes placed in the greenhouse and a master node connected to upper monitoring center. The sensor nodes collect signals of greenhouse soil dryness humidity, control the actuators, and transmit the data through the wi-fi real time graph. It also describes irrigation system is a method of allowing water to pass to the roots of plants, either onto the soil surface or directly onto the root zone, through valve. Thus, this survey proposes a design for smart home garden irrigation system that implements ready-to-use, energy-efficient, and cost effective devices. Arduino codes, which are implemented in this system, is integrated with multi-sensors such as soil moisture sensors, and light sensors. This proposed system managed to reduce cost, minimize waste water, and reduce physical human interface. It is conducted with GUI using Android application to activate watering activity. It states the sensors can be specialized in measuring humidity, water level, soil dryness and so on. In this survey, a solution to transform a normal house in a smart house while reducing the energy consumption is proposed by us. ADAfruit.io is used to create controller which are used to monitor the parameters of the garden and automate the watering process and also control the appliances of the home. NodeMCU is used to connect different sensors which collect the parameters of soil and transmits the information to firebase through inbuilt Wi-Fi. It presents IOT based smart garden monitoring system which senses the requirement of the plant and provide it with water as the soil loses its moisture. Different soils have different fertility and moisture level so we have soil and moisture sensor used in this to detect this problem. In our country there are six different seasons and each day have different temperature and humidity level so to check the moisture and humidity for the better health and survival of plant dryness and humidity sensor are used which regularly sends data to the server. In this way it manages to perform its operations automatically.

This survey lets a student to a wide path of knowledge regarding Internet of Things and Embedded Systems so that a student could prepare itself for a brief introduction to the world of trending technologies and could nurture its growth in IOT and do hand- on several prototypes which could create a plinth- level in IOT.

We have done this survey to create a base of our project and to help us to go deep into the world of Internet of Things and Embedded Systems to create a step- wise methodology for our project.

Chapter 3

Stepping Into World of IOT

3.1 Definition

The Internet of Things, or IoT, refers to the billions of physical devices around the world that are now connected to the internet, all collecting and sharing data. Thanks to the arrival of super-cheap computer chips and the ubiquity of wireless networks, it's possible to turn anything, from something as small as a pill to something as big as an aeroplane, into a part of the IoT. Connecting up all these different objects and adding sensors to them adds a level of digital intelligence to devices that would be otherwise dumb, enabling them to communicate real-time data without involving a human being. The Internet of Things is making the fabric of the world around us smarter and more responsive, merging the digital and physical universes.

The Internet of things (IoT) describes the network of physical objects—a.k.a. "things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the Internet.

Things have evolved due to the convergence of multiple technologies, real-time analytics, machine learning, ubiquitous computing, commodity sensors, and embedded systems. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), and others all contribute to enabling the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", including devices and appliances (such as lighting fixtures, thermostats, home security systems and cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. The IoT can also be used in healthcare systems.

There are a number of serious concerns about dangers in the growth of the IoT, especially in the areas of privacy and security, and consequently industry and governmental moves to address these concerns have begun including the development of international standards.

3.2 How big is the Internet of Things?

Big and getting bigger -- there are already more connected things than people in the world.

Tech analyst company IDC predicts that in total there will be 41.6 billion connected IoT devices by 2025, or "things." It also suggests industrial and automotive equipment representthe largest opportunity of connected "things,", but it also sees strong adoption of smart home and wearable devices in the near term.

Another tech analyst, Gartner, predicts that the enterprise and automotive sectors will account for 5.8 billion devices this year, up almost a quarter on 2019. Utilities will be the highest user of IoT, thanks to the continuing rollout of smart meters. Security devices, in the form of intruder detection and web cameras will be the second biggest use of IoT devices. Building automation – like connected lighting – will be the fastest growing sector, followed by automotive (connected cars) and healthcare (monitoring of chronic conditions).

- With the development of wireless networking technologies, the emergence of advanced data analytics, a reduction in the cost of connected devices, an increase in cloud platform adoption, the market is expected to grow at a positive rate.

- The internet of things technology helps in connecting various smart devices together to ease the operation and sharing of data amongst themselves. There are various smart devices, such as sensors, smartphones, and wearables, which collect necessary data from the devices which are further utilized to enhance customer's experience. The increasing need for data analysis and integration of analytics is expected to propel the utilization of the Internet of Things market over the forecast period.

- Industry 4.0 and IoT are at the center of new technological approaches for development, production, and management of the entire logistics chain, otherwise known as smart factory automation. Massive shifts in manufacturing due to industry 4.0 and acceptance of IoT require enterprises to adopt agile, smarter, and innovative ways to advance production with technologies that complement and augment human labor with robotics and reduce industrial accidents caused by a process failure.

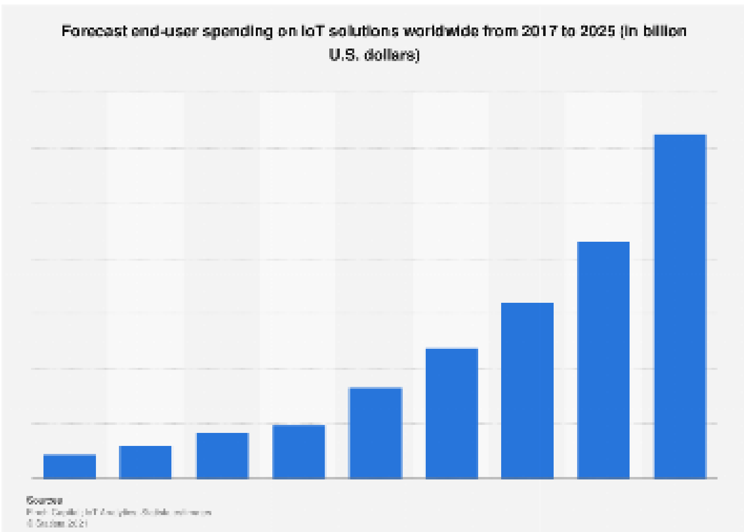


Fig.3.1 IOT endpoint Market by segment 2018-2020

3.3 Building blocks of IOT

Four things form basic building blocks of the IoT system –sensors, processors, gateways, applications. Each of these nodes has to have its own characteristics in order to form an useful IoT system.

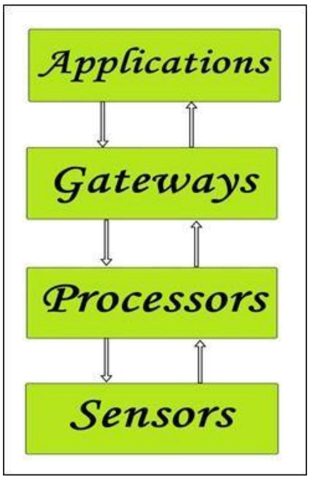


Fig.3.2: Simplified block diagram of the basic building blocks of the IoT

3.3.1 Sensors:

- These form the front end of the IoT devices. These are the so-called "Things" of the system. Their main purpose is to collect data from its surroundings (sensors) or give out data to its surrounding (actuators).
- These have to be uniquely identifiable devices with a unique IP address so that they can be easily identifiable over a large network.
- These have to be active in nature which means that they should be able to collect real-time data. These can either work on their own (autonomous in nature) or can be made to work by the user depending on their needs (user-controlled).
- Examples of sensors are gas sensor, water quality sensor, moisture sensor, etc.

3.3.2 Processors:

- Processors are the brain of the IoT system. Their main function is to process the data captured by the sensors and process them so as to extract the valuable data from the enormous amount of raw data collected. In a word, we can say that it gives intelligence to the data.

Processors mostly work on real-time basis and can be easily controlled by applications. These are also responsible for securing the data – that is performing encryption and decryption of data.

- Embedded hardware devices, microcontroller, etc are the ones that process the data because they have processors attached to it.

3.3.3 Gateways:

- Gateways are responsible for routing the processed data and send it to proper locations for its (data) proper utilization.

- In other words, we can say that gateway helps in to and fro communication of the data. It provides network connectivity to the data. Network connectivity is essential for any IoT system to communicate.

- LAN, WAN, PAN, etc are examples of network gateways.

3.3.4 Applications:

- Applications form another end of an IoT system. Applications are essential for proper utilization of all the data collected.

- These cloud-based applications which are responsible for rendering the effective meaning to the data collected. Applications are controlled by users and are a delivery point of particular services.

- Examples of applications are home automation apps, security systems, industrial control hub, etc.

3.4 Architecture of IOT

The Internet of things (IoT) describes the network of physical objects—a.k.a. "things"—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the Internet.

Things have evolved due to the convergence of multiple technologies, real-time analytics, machine learning, ubiquitous computing, commodity sensors, and embedded systems. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), and others all contribute to enabling the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", including devices and appliances (such as lighting fixtures, thermostats, home security systems and cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. The IoT can also be used in healthcare systems.

There are a number of serious concerns about dangers in the growth of the IoT, especially in the areas of privacy and security, and consequently industry and governmental moves to address these concerns have begun including the development of international standards.

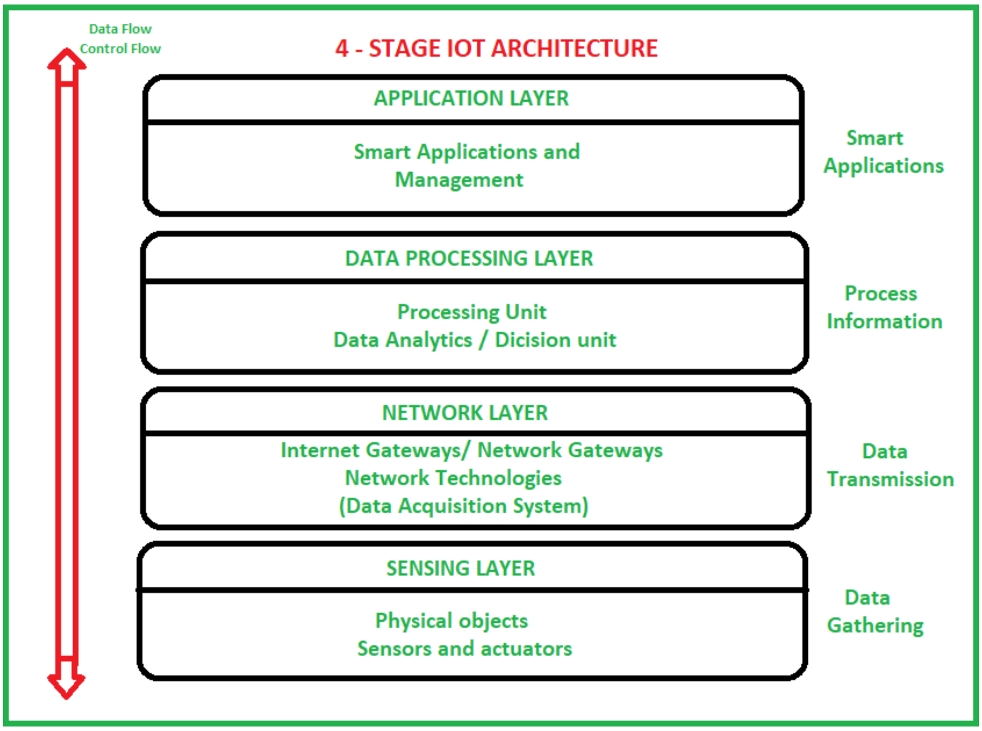


Fig.3.3: IOT Architecture

So, from the above image it is clear that there is 4 layers are present that can be divided as follows: Sensing Layer, Network Layer, Data processing Layer, and Application Layer. These are explained as following below.

- **Sensing layer:** Sensors, actuators, devices are present in this Sensing layer. These Sensors or Actuators accepts data(physical/environmental parameters), processes data and emits data over network.

- **Network Layer:** Internet/Network gateways, Data Acquisition System (DAS) are present in this layer. DAS performs data aggregation and conversion function (Collecting data and aggregating data then converting analog data of sensors to digital data etc). Advanced gateways which mainly opens up connection between Sensor networks and Internet also performs many basic gateway functionalities like malware protection, and filtering also some times decision making based on inputted data and data management services, etc.

- **Data processing Layer:** This is processing unit of IoT ecosystem. Here data is analyzed and pre-processed before sending it to data center from where data is accessed by software applications often termed as business applications where data is monitored and managed and further actions are also prepared. So here Edge IT or edge analytics comes into picture.

- **Application Layer:** This is last layer of 4 stages of IoT architecture. Data centers or cloud is management stage of data where data is managed and is used by end-user applications like agriculture, health care, aerospace, farming, defense, etc.

3.5 Working of IOT

When talking about how IoT works, the process begins with devices that have built-in sensors. These devices are connected to IoT platforms which stores data from all the connected devices. The important data is then used to perform tasks that fulfil the needs of people.

When we say the data is stored in the IoT platforms, it doesn't mean that all the data is useful. Devices carefully select only particular data that is relevant to execute an action. These pieces of information can detect patterns, recommendations and problems before they occur.

In this way, IoT application works with smart systems that automate tasks to address specific needs.

However, if you still have any doubts regarding how does it work, check out this video:-

With this attended to, let's dive deeper into IoT market and see what are the prime components of Internet of Things technology.

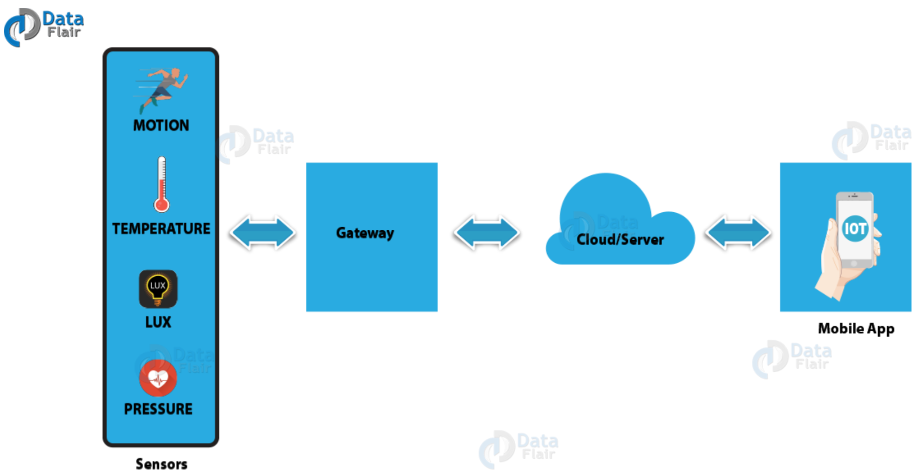


Fig.3.4: Working of IOT

3.6 Characteristics of IOT

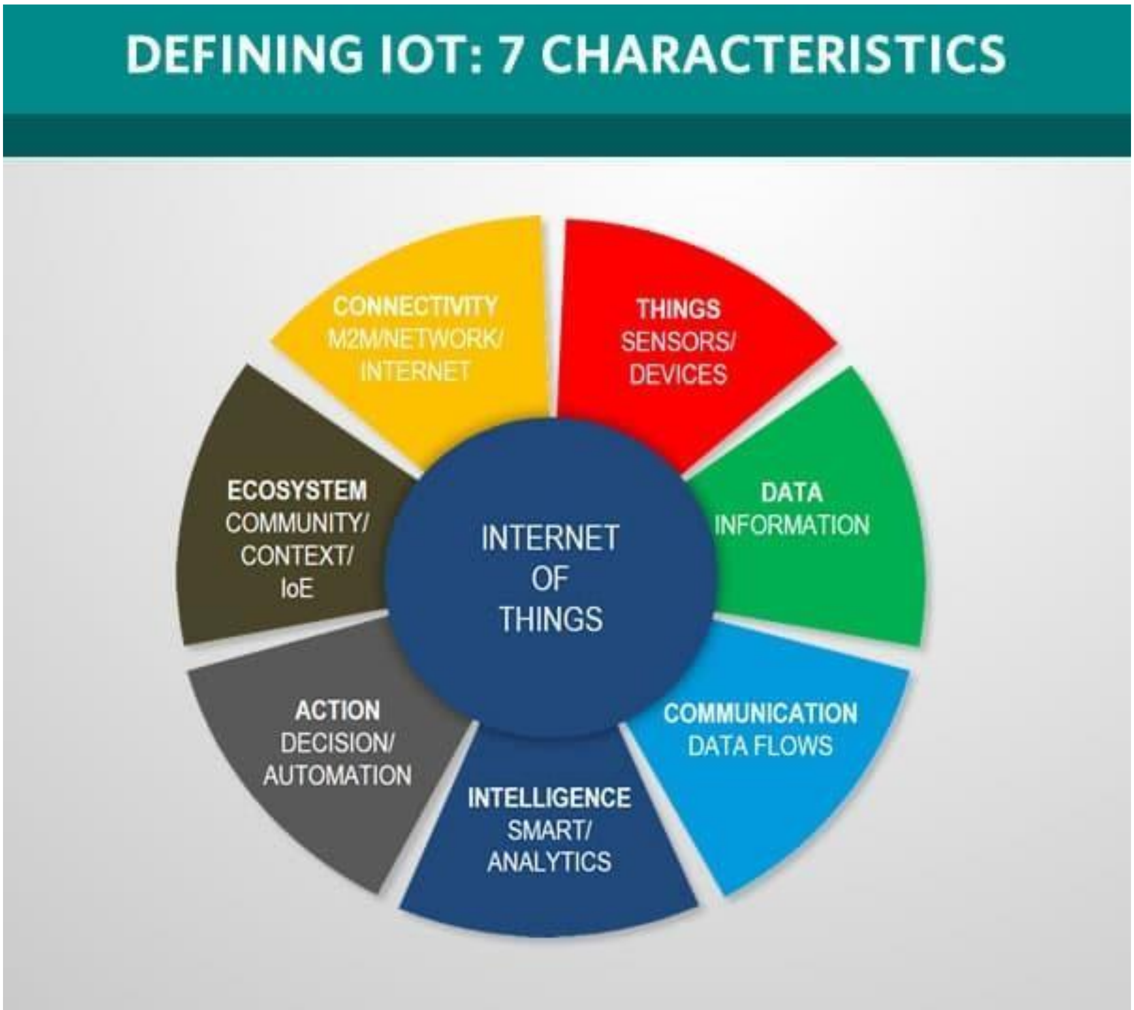


Fig.3.5: Characteristics of IOT

- **Connectivity.** This doesn't need too much further explanation. With everything going on in IoT devices and hardware, with sensors and other electronics and connected hardware and control systems there needs to be a connection between various levels.

- **Things**. Anything that can be tagged or connected as such as it's designed to be connected. From sensors and household appliances to tagged livestock. Devices can contain sensors or sensing materials can be attached to devices and items.

- **Data**. Data is the glue of the Internet of Things, the first step towards action and intelligence.

- **Communication**. Devices get connected so they can communicate data and this data can be analyzed. Communication can occur over short distances or over a long range to very long range. Examples: Wi-Fi, LPWA network technologies such as LoRa or NB-IoT.

- **Intelligence**. The aspect of intelligence as in the sensing capabilities in IoT devices and the intelligence gathered from big data analytics (also artificial intelligence).

- **Action**. The consequence of intelligence. This can be manual action, action based upon debates regarding phenomena (for instance in smart factory decisions) and automation, often the most important piece.

- **Ecosystem**. The place of the Internet of Things from a perspective of other technologies, communities, goals and the picture in which the Internet of Things fits. The Internet of Everything dimension, the platform dimension and the need for solid partnerships.

3.7 Pros and cons of IOT

3.7.1 Advantages:

- ability to access information from anywhere at any time on any device;
- improved communication between connected electronic devices;
- transferring data packets over a connected network saving time and money; and
- Automating tasks helping to improve the quality of a business's services and reducing the need for human intervention.

3.7.2 Disadvantages:

- As the number of connected devices increases and more information is shared between devices, the potential that a hacker could steal confidential information also increases.
- Enterprises may eventually have to deal with massive numbers -- maybe even millions -- of IoT devices, and collecting and managing the data from all those devices will be challenging.
- If there's a bug in the system, it's likely that every connected device will become corrupted.
- Since there's no international standard of compatibility for IoT, it's difficult for devices from different manufacturers to communicate with each other.

3.8 Security concerns of IOT

The internet of things connects billions of devices to the internet and involves the use of billions of data points, all of which need to be secured. Due to its expanded attack surface, IoT security and IoT privacy are cited as major concerns.

In 2016, one of the most notorious recent IoT attacks was Mirai, a botnet that infiltrated domain name server provider Dyn and took down many websites for an extended period of time in one of the biggest distributed denial-of-service (DDoS) attacks ever seen. Attackers gained access to the network by exploiting poorly secured IoT devices.

Because IoT devices are closely connected, all a hacker has to do is exploit one vulnerability to manipulate all the data, rendering it unusable. Manufacturers that don't update their devices regularly -- or at all -- leave them vulnerable to cybercriminals.

Additionally, connected devices often ask users to input their personal information, including names, ages, addresses, phone numbers and even social media accounts -- information that's invaluable to hackers.

Hackers aren't the only threat to the internet of things; privacy is another major concern for IoT users. For instance, companies that make and distribute consumer IoT devices could use those devices to obtain and sell users' personal data.

Beyond leaking personal data, IoT poses a risk to critical infrastructure, including electricity, transportation and financial services.

3.9 Government regulations on IOT

One of the key drivers of the IoT is data. The success of the idea of connecting devices to make them more efficient is dependent upon access to and storage & processing of data. For this purpose, companies working on the IoT collect data from multiple sources and store it in their cloud network for further processing. This leaves the door wide open for privacy and security dangers and single point vulnerability of multiple systems. The other issues pertain to consumer choice and ownership of data and how it is used. Though still in their infancy, regulations and governance regarding these issues of privacy, security, and data ownership continue to develop. IoT regulation depends on the country. Some examples of legislation that is relevant to privacy and data collection are: the US Privacy Act of 1974, OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data of 1980, and the EU Directive 95/46/EC of 1995.

Current regulatory environment:

A report published by the Federal Trade Commission (FTC) in January 2015 made the following three recommendations:

- **Data security** – At the time of designing IoT companies should ensure that data collection, storage and processing would be secure at all times. Companies should adopt a "defense in depth" approach and encrypt data at each stage.

- **Data consent** – users should have a choice as to what data they share with IoT companies and the users must be informed if their data gets exposed.

- **Data minimisation** – IoT companies should collect only the data they need and retain the collected information only for a limited time.

Chapter 4

ARDUINO

4.1 Introduction

Arduino is an open-source hardware and software company, project and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. Its hardware products are licensed under a CC-BY-SA license, while software is licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially from the official website or through authorized distributors.

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analogue input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs. The microcontrollers can be programmed using the C and C++ programming languages, using a standard API which is also known as the "Arduino language". In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) and a command line tool (arduino-cli) developed in Go.

The Arduino project began in 2005 as a tool for students at the Interaction Design Institute Ivrea in Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors.

The name Arduino comes from a bar in Ivrea, Italy, where some of the founders of the project used to meet. The bar was named after Arduin of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014.

4.2 History

The Arduino project was started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller at a cost of $50. In 2003 Hernando Barragán created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas. Casey Reas is known for co- creating, with Ben Fry, the Processing development platform. The project goal was to create simple, low cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2005, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, extended Wiring by adding support for the cheaper ATmega8 microcontroller. The new project, forked from Wiring, was called Arduino.

The initial Arduino core team consisted of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis.

Following the completion of the platform, lighter and less expensive versions were distributed in the open-source community. It was estimated in mid-2011 that over 300,000 official Arduinos had been commercially produced and in 2013 that 700,000 official boards were in users' hands.

4.3 Trademark Dispute

In early 2008, the five co-founders of the Arduino project created a company, Arduino LLC, to hold the trademarks associated with Arduino. The manufacture and sale of the boards was to be done by external companies, and Arduino LLC would get a royalty from them. The founding bylaws of Arduino LLC specified that each of the five founders transfer ownership of the Arduino brand to the newly formed company.

At the end of 2008, Gianluca Martino's company, Smart Projects, registered the Arduino trademark in Italy and kept this a secret from the other co-founders for about two years. This was revealed when the Arduino company tried to register the trademark in other areas of the world (they originally registered only in the US), and discovered that it was already registered in Italy. Negotiations with Martino and his firm to bring the trademark under control of the original Arduino Company failed. In 2014, Smart Projects began refusing to pay royalties. They then appointed a new CEO, Federico Musto, who renamed the company Arduino SRL and created the website arduino.org, copying the graphics and layout of the original arduino.cc. This resulted in a rift in the Arduino development team.

In January 2015, Arduino LLC filed a lawsuit against Arduino SRL.

In May 2015, Arduino LLC created the worldwide trademark Genuino, used as brand name outside the United States.

At the World Maker Faire in New York on 1 October 2016, Arduino LLC co-founder and CEO Massimo Banzi and Arduino SRL CEO Federico Musto announced the merger of the two companies. Around that same time, Massimo Banzi announced that in addition to the company a new Arduino Foundation would be launched as "a new beginning for Arduino.", but this decision was withdrawn later.

In April 2017, Wired reported that Musto had "fabricated his academic record. On his company's website, personal LinkedIn accounts, and even on Italian business documents, Musto was, until recently, listed as holding a PhD from the Massachusetts Institute of Technology. In some cases, his biography also claimed an MBA from New York University." Wired reported that neither university had any record of Musto's attendance, and Musto later admitted in an interview with Wired that he had never earned those degrees. The controversy surrounding Musto continued when, in July 2017, he reportedly pulled many Open source licenses, schematics, and code from the Arduino website, prompting scrutiny and outcry.

By 2017 Arduino AG owned many Arduino trademarks. In July 2017 BCMI, founded by Massimo Banzi, David Cuartielles, David Mellis and Tom Igoe, acquired Arduino AG and all the Arduino trademarks. Fabio Violante is the new CEO replacing Federico Musto, who no longer works for Arduino AG.

4.4 Hardware

Arduino is open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available.

Although the hardware and software designs are freely available under copyleft licenses, the developers have requested the name Arduino to be exclusive to the official product and not be used for derived works without permission. The official policy document on use of the Arduino name emphasizes that the project is open to incorporating work by others into the official product. Several Arduino-compatible products commercially released have avoided the project name by using various names ending in -duino.

Most Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, or ATmega2560) with varying amounts of flash memory, pins, and features. The 32-bit Arduino Due, based on the Atmel SAM3X8E was introduced in 2012. The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits. These may connect with add-on modules termed shields. Multiple and possibly stacked shields may be individually addressable via an I²C serial bus. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator. Some designs, such as the LilyPad, run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions.

Arduino microcontrollers are pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory. The default bootloader of the Arduino Uno is the Optiboot bootloader. Boards are loaded with program code via a serial connection to another computer. Some serial Arduino boards contain a level shifter circuit to convert between

RS-232 logic levels and transistor–transistor logic (TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools, instead of the Arduino IDE, standard AVR in-system programming (ISP) programming is used.

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila, Duemilanove, and current Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.

Many Arduino-compatible and Arduino-derived boards exist. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education, to simplify making buggies and small robots. Others are electrically equivalent, but change the form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use different processors, of varying compatibility.

4.4.1 Official Boards

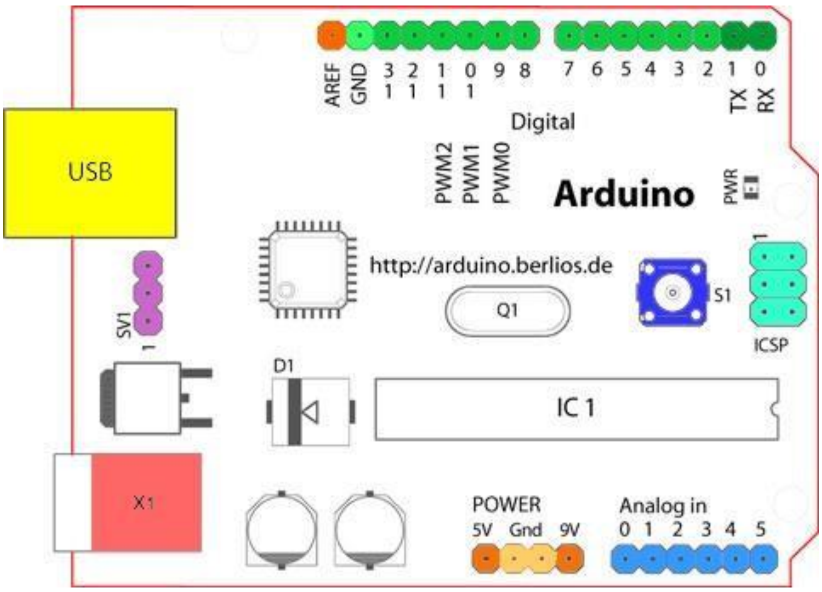


Fig.4.1: Arduino Board Diagram

4.4.1.1 Digital Pins

In addition to the specific functions listed below, the digital pins on an Arduino board can be used for general purpose input and output via the pinMode(), digitalRead(), and digitalWrite() commands. Each pin has an internal pull-up resistor which can be turned on and off using digitalWrite() (w/ a value of HIGH or LOW, respectively) when the pin is configured as an input. The maximum current per pin is 40 mA.

- **Serial:** 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. On the Arduino Diecimila, these pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip. On the Arduino BT, they are connected to the corresponding pins of the WT11 Bluetooth module. On the Arduino Mini and LilyPad Arduino, they are intended for use with an external TTL serial module (e.g. the Mini-USB Adapter).

- **External Interrupts:** 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.

- **PWM:** 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function. On boards with an ATmega8, PWM output is available only on pins 9, 10, and 11.

- **BT Reset:** 7. (Arduino BT-only) Connected to the reset line of the bluetooth module.

- **SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

- **LED:** 13. On the Diecimila and LilyPad, there is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

4.4.1.2 Analog Pins

In addition to the specific functions listed below, the analog input pins support 10-bit analog- to-digital conversion (ADC) using the analogRead() function. Most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. Analog inputs 6 and 7 (present on the Mini and BT) cannot be used as digital pins.

4.4.1.3 Power Pins

- VIN (sometimes labelled "9V"). The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. Note that different boards accept different input voltages ranges. Also note that the LilyPad has no VIN pin and accepts only a regulated input.

- 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- 3V3. (Diecimila-only) A 3.3 volt supply generated by the on-board FTDI chip.
- GND. Ground pins.

4.4.1.4 Other Pins

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. (Diecimila-only) Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

4.4.2 Shields

Shields are boards to be mounted on top of the Arduino board and that extend the functionality of Arduino to control different devices, acquire data, etc.

- **Xbee Shield** - This shield allows multiple Arduino boards to communicate wirelessly over distances up to 100 feet (indoors) or 300 feet (outdoors) using the Maxstream Xbee Zigbee module.
- **Motor Shield** - This shield allows an Arduino board to control DC motors and read encoders.
- **Ethernet Shield** - This shield allows an Arduino board to connect to the internet.

Fig.4.2: Arduino Shield

4.5 Software

A program for Arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR Studio (older) and Atmel Studio (newer).

4.5.1 IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, and Linux) that is written in the Java programming language. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

4.5.2 Pro IDE

On October 18, 2019, Arduino Pro IDE (alpha preview) was released. The system still uses Arduino CLI (Command Line Interface), but improvements include a more professional development environment, autocompletion support, and Git integration. The application frontend is based on the Eclipse Theia Open Source IDE. The main features available in the alpha release are:

- Modern, fully featured development environment
- Dual Mode, Classic Mode (identical to the Classic Arduino IDE) and Pro Mode (FileSystem view)
- New Board Manager
- New Library Manager

- Board List
- Basic Auto-Completion (Arm targets only)
- Git Integration
- Serial Monitor
- Dark Mode

4.5.3 Sketch

A sketch is a program written with the Arduino IDE. Sketches are saved on the development computer as text files with the file extension .ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension .pde.

A minimal Arduino C/C++ program consists of only two functions:

- **setup() :**This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. It is analogous to the function main() .

- **loop() :** After setup() function is executed repeatedly in loop(). function exits (ends), the the main program. It controls the board until the board is powered off or is reset. It is analogous to the function while(1) .
- Power LED (red) and User LED (green) attached to pin 13 on an Arduino compatible board.

- Most Arduino boards contain a light emitting diode (LED) and a current limiting resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions. A typical program used by beginners, akin to Hello, World!, is "blink", which repeatedly blinks the on-board LED integrated into the Arduino board. This program uses the functions pinMode() , digitalWrite() , and delay() , which are provided by the internal libraries included in the IDE environment.

4.6 Operating System / Threading

There is a Xinu OS port for the atmega328p (Arduino Uno and others with the same chip), which includes most of the basic features. The source code of this version is freely available.

There is also a threading tool, named Protothreads. Protothreads are described as "extremely lightweight stackless threads designed for severely memory constrained systems, such as small embedded systems or wireless sensor network nodes. Protothreads provide linear code execution for event-driven systems implemented in C. Protothreads can be used with or without an underlying operating system."

4.7 Application

- Learning & experimenting
- Prototyping & validation
- Sacrificial concepts (user research)
- temporary installations & demonstrations

Chapter 5

Roadmap of Internet of Things

5.1 Architecture of Internet of Things

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.

So, basic fundamental architecture of IoT i.e., 4 Stage IoT architecture.

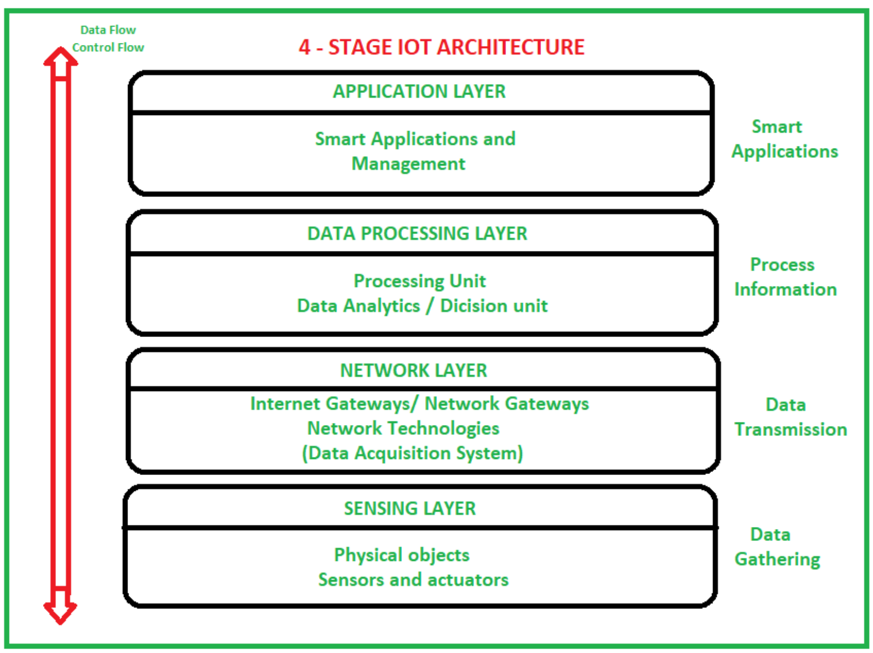


Fig.5.1 Diagram showing 4-Stage IOT Architecture

There are 4 layers are present that can be divided as follows:

Sensing Layer, Network Layer, Data processing Layer, and Application Layer.

These are explained as following below.

- **SensingLayer:**

  Sensors, actuators, devices are present in this Sensing layer. These Sensors or Actuators accepts data (physical/environmental parameters), processes data and emits data over network.

- **Network Layer:**

  Internet/Network gateways, Data Acquisition System (DAS) are present in this layer. DAS performs data aggregation and conversion function (Collecting data and aggregating data then converting analog data of sensors to digital data etc). Advanced gateways which mainly opens up connection between Sensor networks and Internet also performs many basic gateway functionalities like malware protection, and filtering also sometimes decision making based on inputted data and data management services, etc.

- **Data processing Layer –**

  This is processing unit of IoT ecosystem. Here data is analyzed and pre-processed before sending it to data center from where data is accessed by software applications often termed as business applications where data is monitored and managed and further actions are also prepared. So here Edge IT or edge analytics comes into picture.

- **Application Layer –**

  This is last layer of 4 stages of IoT architecture. Data centers or cloud is management stage of data where data is managed and is used by end-user applications like agriculture, health care, aerospace, farming, defense, etc.

The IOT architecture is basically divided into two main parts:

5.1.1 Architecture Reference Model

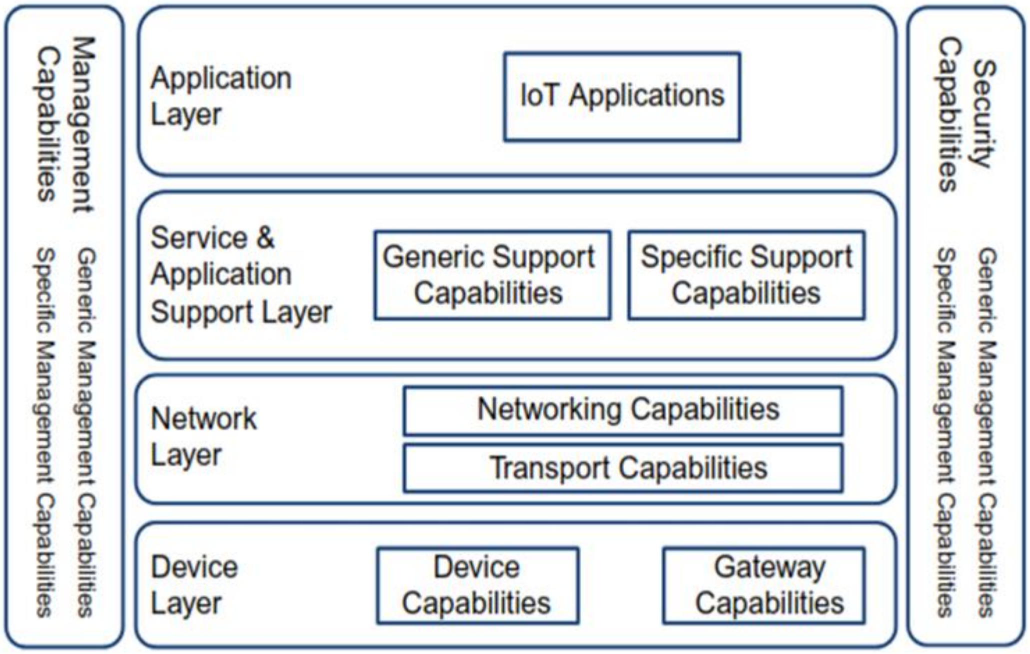


Fig.5.2 Diagram showing Architecture Reference Model

Internet of Things (IoT) includes large number of smart devices connected to a broad internet network with the help of various networking technologies. Mostly these technologies are wireless in manner. This makes the structure more complex and difficult to manage. Therefore, architecture is required.

An architecture is structure for specification of network's physical components and their functional organization and configuration, its operational principles and procedures, as well as data formats used in its operation.

The development of IoT depends on the technologies used, application areas, and business aspects. There are various IoT architectures are available for IoT devices. However, the "5 Layer Architecture is considered as the best-proposed architecture of IoT."

5.1.2 5 Layer Architecture of IoT

When project work is done with various cutting edge technologies and broad application area, 5 layer architecture is considered as best. 5 Layer model can be considered as an extension to the basic architecture of IoT because it has two additional layers to the basic model.

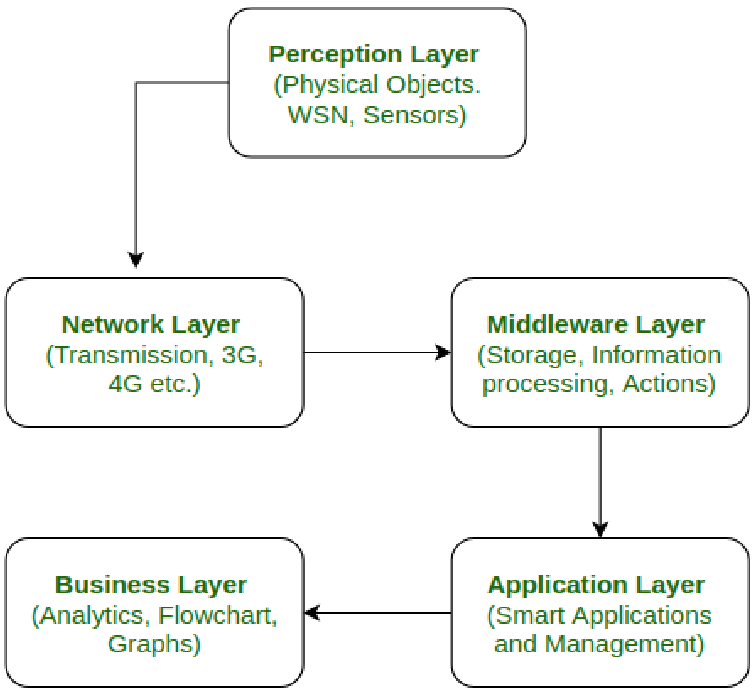


Fig.5.3 Diagram showing 5 Layer Architecture of Internet of Things

- **Perception Layer :**
  This is the first layer of IoT architecture. In the perception layer, number of sensors and actuators are used to gather useful information like temperature, moisture content, intruder detection, sounds, etc. The main function of this layer is to get information from surroundings and to pass data to another layer so that some actions can be done based on that information.

- **Network Layer :**

  As the name suggests, it is the connecting layer between perception and middleware layer. It gets data from perception layer and passes data to middleware layer using networking technologies like 3G, 4G, UTMS, WiFI, infrared, etc. This is also called communication layer because it is responsible for communication between perception and middleware layer. All the transfer of data done securely keeping the obtained data confidential.

- **Middleware Layer :**

  Middleware Layer has some advanced features like storage, computation, processing, action taking capabilities. It stores all data-set and based on the device address and name it gives appropriate data to that device. It can also take decisions based on calculations done on data-set obtained from sensors.

- **Application Layer :**

  The application layer manages all application process based on information obtained from middleware layer. This application involves sending emails, activating alarm, security system, turn on or off a device, smartwatch, smart agriculture, etc.

- **Business Layer :**

  The success of any device does not depend only on technologies used in it but also how it is being delivered to its consumers. Business layer does these tasks for the device. It involves making flowcharts, graphs, analysis of results, and how device can be improved, etc.

5.1.3 Architecture Reference Model

The domain model captures the basic attributes of the main concepts and the relationship between these concepts. A domain model also serves as a tool for human communication between people working in the domain in question and between people who work across different domains.

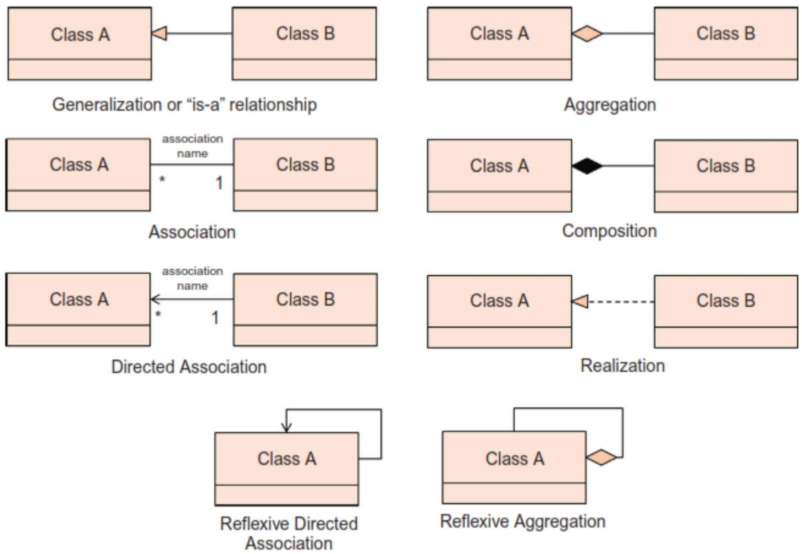


Fig. Diagram showing Architecture Reference Model

The Devices are physical artefacts with which the physical and virtual worlds interact. Devices as mentioned before can also be Physical Entities for certain types of applications, such as management applications when the interesting entities of a system are the Devices themselves and not the surrounding environment. For the IoT Domain Model, three kinds of Device types are the most important:

Sensors:

- These are simple or complex Devices that typically involve a transducer that converts physical properties such as temperature into electrical signals.
- These Devices include the necessary conversion of analog electrical signals into digital signals, e.g. a voltage level to a 16-bit number, processing for simple calculations, potential storage for intermediate results, and potentially communication capabilities to transmit the digital representation of the physical property as well receive commands.
- A video camera can be another example of a complex sensor that could detect and recognise people.

Actuators:

- These are also simple or complex Devices that involve a transducer that converts electrical signals to a change in a physical property (e.g. turn on a switch or move a motor).
- These Devices also include potential communication capabilities, storage of intermediate commands, processing, and conversion of digital signals to analog electrical signals.

Tags:

- Tags in general identify the Physical Entity that they are attached to. In reality, tags can be Devices or Physical Entities but not both, as the domain model shows.
- An example of a Tag as a Device is a Radio Frequency Identification (RFID) tag, while a tag as a Physical Entity is a paper-printed immutable barcode or Quick Response (QR) code.
- Either electronic Devices or a paper-printed entity tag contains a unique identification that can be read by optical means (bar codes or QR codes) or radio signals (RFID tags).
- The reader Device operating on a tag is typically a sensor, and sometimes a sensor and an actuator combined in the case of writable RFID tags.

5.1.4 Information Model

Virtual Entity in the IoT Domain Model is the "Thing" in the Internet of Things, the IoT information model captures the details of a Virtual Entity- centric model. Similar to the IoT Domain Model, the IoT Information Model is presented using Unified Modelling Language (UML) diagrams.

5.1.5 Functional model

The IoT Functional Model aims at describing mainly the Functional Groups (FG) and their interaction with the ARM, while the Functional View of a Reference Architecture describes the functional components of an FG, interfaces, and interactions between the components.

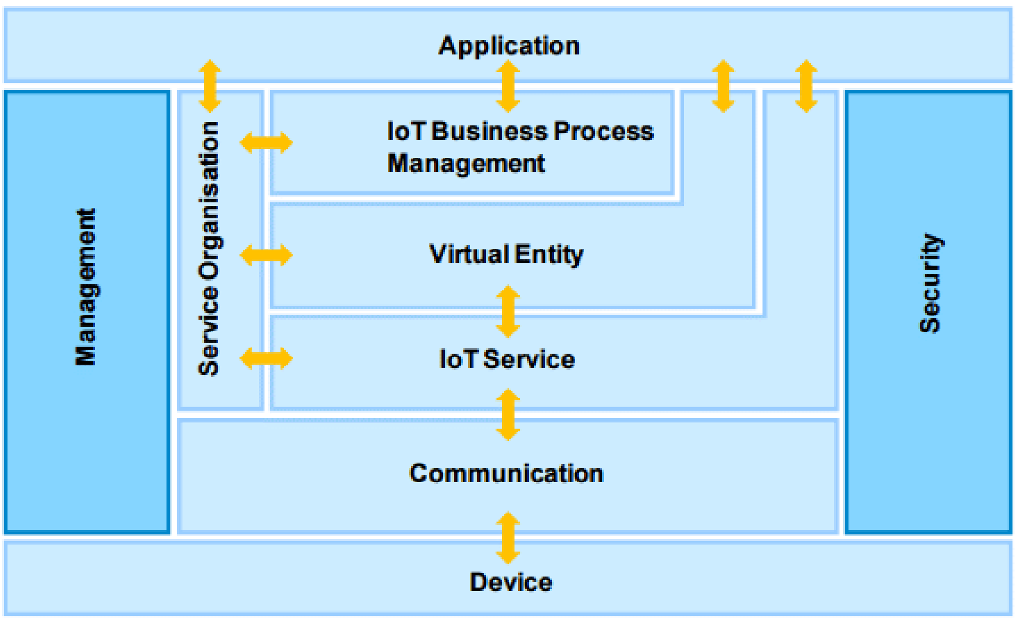


Fig.5.4 Diagram showing Architecture Reference Model

The Functional View is typically derived from the Functional Model in conjunction with high-level requirements.

5.1.6 Device functional group

The Device FG contains all the possible functionality hosted by the physical Devices that are used for increment the Physical Entities. This Device functionality includes sensing, actuation, processing, storage, and identification components, the sophistication of which depends on the Device capabilities.

5.1.7 Communication functional group

The Communication FG abstracts all the possible communication mechanisms used by the relevant Devices in an actual system in order to transfer information to the digital world components or other Devices.

5.1.8 IoT Service functional group

The IoT Service FG corresponds mainly to the Service class from the IoT Domain Model, and contains single IoT Services exposed by Resources hosted on Devices or in the Network (e.g. processing or storage Resources).

5.1.9 Virtual Entity functional group

The Virtual Entity FG corresponds to the Virtual Entity class in the IoT Domain Model, and contains the necessary functionality to manage associations between Virtual Entities with themselves as well as associations between Virtual Entities and related IoT Services, i.e. the Association objects for the IoT Information Model. Associations between Virtual Entities can be static or dynamic depending on the mobility of the Physical Entities related to the corresponding Virtual Entities.

5.1.10 IoT Service Organization functional group

The purpose of the IoT Service Organisation FG is to host all functional components that support the composition and orchestration of IoT and Virtual Entity services. Moreover, this FG acts as a service hub between several other functional groups such as the IoT Process

Management FG when, for example, service requests from Applications or the IoT Process Management are directed to the Resources implementing the necessary Services.

IoT Process Management functional group

The IoT Process Management FG is a collection of functionalities that allows smooth integration of IoT-related services (IoT Services, Virtual Entity Services, Composed Services) with the Enterprise (Business) Processes.

Management functional group

The Management FG includes the necessary functions for enabling fault and performance monitoring of the system, configuration for enabling the system to be flexible to changing User demands, and accounting for enabling subsequent billing for the usage of the system. Support functions such as management of ownership, administrative domain, rules and rights of functional components, and information stores are also included in the Management FG.

Security functional group

The Security FG contains the functional components that ensure the secure operation of the system as well as the management of privacy. The Security FG contains components for Authentication of Users (Applications, Humans), Authorisation of access to Services by Users, secure communication (ensuring integrity and confidentiality of messages) between entities of the system such as Devices, Services, Applications, and last but not least, assurance of privacy of sensitive information relating to Human Users.

Application functional group

The Application FG is just a placeholder that represents all the needed logic for creating an IoT application. The applications typically contain custom logic tailored to a specific domain such as a Smart Grid

Communication model

Safety

the IoT Reference Model can only provide IoT-related guidelines for ensuring a safe system to the extent possible and controllable by a sys- tem designer.
Eg: smart grid.

Privacy

Because interactions with the physical world may often include humans, protecting the User privacy is of utmost importance for an IoT system. The IoT-A Privacy Model depends on the following functional components: Identity Management, Authentication, Authorisation, and Trust & Reputation

Trust

Generally, an entity is said to ‘trust’ a second entity when the first entity makes the assumption that the second entity will behave exactly as the first entity expects.”

Security

The Security Model for IoT consists of communication security that focuses mostly on the confidentiality and integrity protection of interacting entities and functional components such as Identity Management, Authentication, Authorisation, and Trust & Reputation.

Architecture of home automation system

To design a intelligent home automation system to reduce the human efforts and save time by overcoming the traditional methods of detecting the faults in home appliances and fixing it by using sensors.

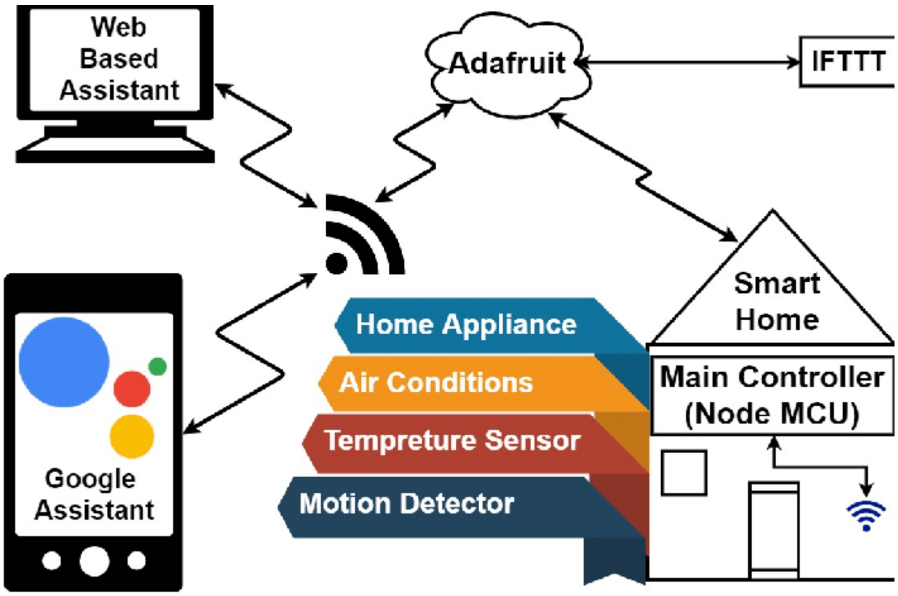


Fig.5.5 Diagram showing Architecture of Home Automation System

The concept of Home Automation aims to bring the control of operating your every day home electrical appliances to the tip of your finger, thus giving user affordable lighting solutions, better energy conservation with optimum use of energy. Apart from just lighting solutions, the concept also further extends to have a overall control over your home security as well as build a centralised home entertainment system and much more. The **Internet of Things** (or commonly referred to as IoT) based Home Automation system, as the name suggests aims to control all the devices of your smart home through internet protocols or cloud based computing.

The IoT based Home Automation system offer a lot of flexibility over the wired systems s it comes with various advantages like ease-of-use, ease-of-installation, avoid complexity of running through wires or loose electrical connections, easy fault detection and triggering and above and all it even offers easy mobility.
the working of different smart devices which together constitute the Home Automation system.

Controller: The Brain of Your System

The main controller or the hub is the most essential part of your Home Automation system irrespective of whether you connect single or multiple sensors in your home. The main controller or the hub is also referred to as gateway and is connected to your home router through the Ethernet cable. All the IoT based sensors transmits or receive commands through the centralised hub. The hub in turn receives the input or communicates the output to cloud network located over the internet.

Due to this kind of architecture, it is possible to communicate with the centralised hub even from remote and distant locations through your smartphone. All you need is just a reliable internet connection at the hub location and the data package to your smartphone that helps you connect to the cloud network.
Most of the smart home controllers available in the market from several manufacturers cater to all three widely used protocols of wireless communication for Home Automation: **ZigBee, Z-Wave and Wi-Fi.**

Smart Devices: The Sensory Organs of Your Home

The IoT based home automation consist of several smart devices for different applications of lighting, security, home entertainment etc. All these devices are integrated over a common network established by gateway and connected in a mesh network. This means that it gives users the flexibility to operate one sensor based followed by the action of the other. For e.g.

you can schedule to trigger the living room lights as soon as the door/windows sensor of your main door triggers after 7pm in the evening.

Thus all the sensors within a common network can perform cross-talk via the main controller unit. As shown in the figure, some of the smart sensors in home automation acts as sensor hubs. These are basically the signal repeaters of signal bouncers which that are located in the midway between the hub installation location and the sensors that are at a distant location. For such long distances, these sensor hubs play an important role to allow easy transmission of signals to sensors that are far away from the main controller but in closer proximity to the sensor hub. The commonly used sensor hubs in IoT based Home Automation system are Smart Plugs.

Wireless Connectivity: How the Internal Communication Occurs

Most of the IoT based Home Automation systems available today work on three widely used wireless communication protocols : Wi-Fi, ZigBee and Z-Wave

The ZigBee and the Z-Wave controllers are assigned a network ID which is distributed over other sensors in the network. The communication amongst devices take place in a mesh topology where there is no fixed path for the signals transmitted from the controller to the sensors and vice versa. Depending on the availability of the shortest path the signal from the controller will travel to the target sensors either directly or through signal hops. If any intermediate sensor in the pathway is busy or occupied the signal will trace another path within the mesh network to reach the final destination. Note that sensors with different Network IDs cannot communicate with each other over common channel.
Wi-Fi: Please explain this section.

Connected with the Cloud: Access Everything on the Go

The Cloud-based-Networking system involves storage and maintenance of data over the Internet location. This gives users the flexibility to have access to the data from any location on the planet.

As a result of this, in IoT based Home Automation systems users over the cloud network can send commands to the hub even from a distant or remote location. The hub will further send the signal for the intended sensors to trigger and perform the user-requested action. Once the action is performed, the hub will update the status of the action taken to the cloud network and in this way users can control and monitor every aspect of their smart homes.

Events and Notifications: Get Notified Instantly

Real-time monitoring and notifications is one of the key features of IoT based Home Automation systems. Since the hub is connected over the cloud network through the Internet, you can schedule various events as per your routine activities or daily schedules. The cloud network can receive and store all the user inputs and transfer them to the hub as per the scheduled events.

Once the hub transfer the desired signals to the target sensor and the desired action takes places, it will quickly upload the new status over the cloud notifying user instantaneously. For e.g. the motion sensor will instantaneously notify the user wither through emails, SMS, calls or App notifications when it detects any unwanted motion or intrusion. After receiving such notification, the user can quickly turn on the IP based home security smart camera can check the status of your home even from remote location.

IFTTT Integration: Put Internet to Work for You

It is not practically possible to trigger every action one by one in your day long busy schedule. This is where you can put the Internet to work for you. The IF This Then That (IFTTT) Integration helps you in this condition.

This system is based on Internet of things which aims at connecting maximum devices through internet.We are surrounded by Plants and tress it provide us with oxygen, food e.t.c. In this Smart garden monitoring system water is provided to the plant with the help of pump motor. In our garden we have many types of herbs, shurbs and climbers.But in our busy schedule we don't have time to take care of these plants which provides beauty to the nature and foods for the needy so this system with the help of soil and moisture sensor checks the fertility of the soil and provide it with water through the pump. In this system NodeMCU will be programmed by connecting it through 3.3V USB cable. It will sense the data sent by soil moisture sensor and humidity sensor.

PUMP CONTROLLER This controller is suitable for any type of motor - Single or Three Phase. Switches ON the pump when the water in the overhead tank goes below the pre-decided minimum level. Switches OFF the pump when the water level in the overhead tank reaches the maximum level therefore prevents overflow. Shall again switch ON the pump when there is sufficient water in the underground tank. Therefore no need to switch ON or switch OFF the pump manually.

SOFTWARE DISCRIPTION This paper is based on Internet of Things. It work on the mobile application and on the web server by uploading the data on the arduiono application available on PC. In mobile application BLYNK we upload the wifi hotspot name and the password in our program based on embedded C programming language. In web server on a particular IP address or the web page data is uploaded on the server and through the sensors data is uploaded on the web server. GSM connectivity is also programmed in ESP 8266 by using the GSM library available in the Arduino application. It is shown in the figure how the button is selected in the application and how the temperature, moisture and humidity is uploaded on the web. Rx and Tx of the ESP 8266 is connected through the pins and it is defined in the programming.

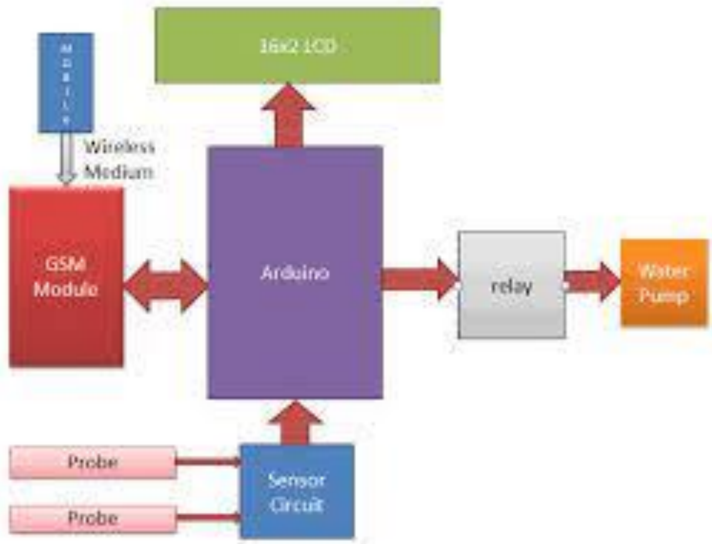


Fig.5.6 Diagram showing IFTTT Integration

Architecture of Embedded System

Typical embedded system mainly has two parts i.e., embedded hardware and embedded software.

Embedded hardwares are based around microprocessors and microcontrollers, also include memory, bus, Input / Output, Controller, where as embedded software includes embedded operating systems, different applications and device drivers. Basically these two types of architecture i.e., Havard architecture and Von Neumann architecture are used in embedded systems.

Architecture of the Embedded System includes Sensor, Analog to Digital Converter, Memory, Processor, Digital to Analog Converter, and Actuators etc.

The below figure illustrates the overview of basic architecture of embedded systems :

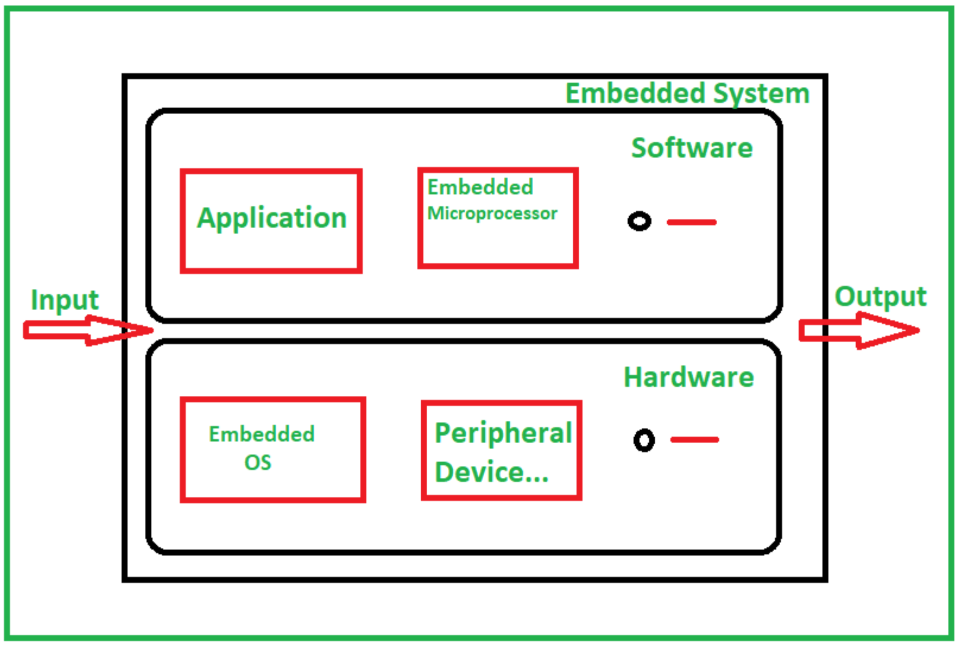


Fig.5.7 Diagram showing Architecture of Embedded System

Chapter 6
Protocols and Models

6.1 Protocols in IOT

- Home automation technology witnessed a huge leap forward.
- Products like Amazon Echo and Google Home were finally able to break into the mainstream consumer space and start an industry trend.
- Consequently, it is expected to continue this trend with technologies like facial recognition, voice commands and biometrics finding wider application in smart home solutions.
- Leading tech companies are investing huge amounts in R&D to create their own connected home and office ecosystems.
- The ultimate vision is to create completely automated systems which can recognize and respond to select individuals or user groups, within a home or office environment.
- A prototype has been implemented to show the real illustration of the proposed approach.
- An android mobile application has been developed to display the real-time profiles of environmental factors like soil, moisture, and humidity.
- With the help of this system, users will be able to treat their gardens in a better way in terms of plant health and growth.
- This report work replaces the need for gardeners and issues faced during the maintenance of gardens in big cities.
- The purpose of this research is to introduce and prosper the IoT innovation towards smart cities in our society.

Here we review the popular home automation and smart gardening system protocols in use today.

6.1.1 Communication Protocols

- Communication protocol defines the type of medium used for data exchange between two components.
- For this purpose, we use C++, a file format type that has lightweight and flexible, used for serializing and transmitting structured data over a network connection–transmit data between a server and a client.
- This approach is considered as a standard approach in most of the custom built machines and devices as it requires the data transfer in the form of the variable between device and application.
- We use a Wi-Fi Manager and Client Libraries in Arduino IDE to allow the device to connect application via an IP Address and transfer the live detected parameters directly to the app.

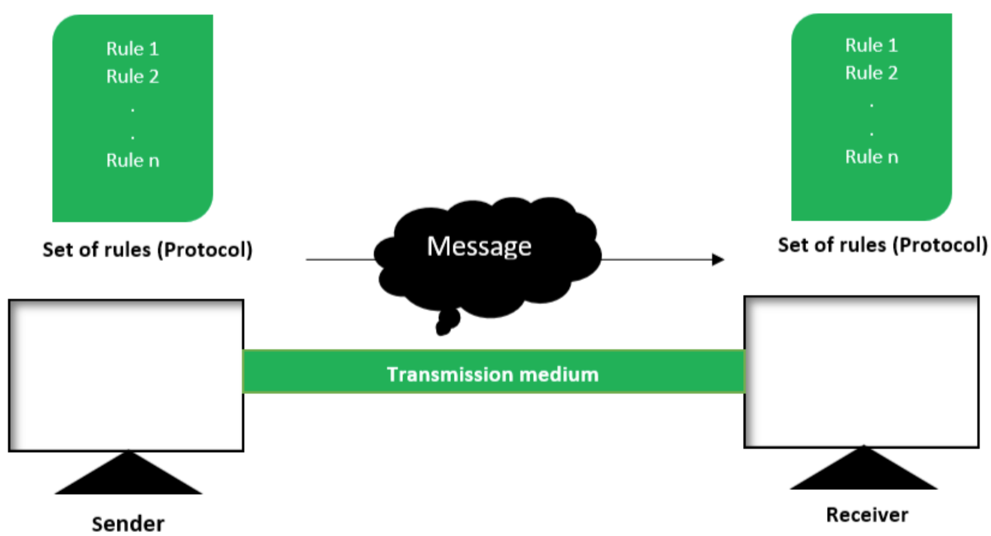


Fig.6.1 Diagram showing Transmission using Protocols

6.1.2 X10 Home Automation

- One of the first home control technologies to be introduced, x10 has been around since 1975.
- X10 utilizes existing power lines for controlling lamps, appliances and monitoring doors and windows. Technically, X10 is a little behind its time, and delays between an action and reaction are perceivable due to its use of RF signals.
- X10 protocols are also susceptible to disruptions related to line noise and require additional installation of filters, couplers and repeaters to minimize such instances.
- An upgrade called the Universal Powerline Bus (UPB) was introduced in 1999, which addresses most of the X10's shortcomings.
- UPB uses a more complex setup involving unique device IDs, shared network access and individual network passwords to provide a more realistic and customizable smart home implementation.

Fig.6.2 Diagram showing X10 Home Automation

6.1.3 Zigbee Home Automation

- A leading name in smart home technology, the Zigbee protocol is renowned for its convenient operation and interoperability with older versions.
- The protocol claims to add almost seven years of battery life to smart security sensors – providing a long-term home security solution.
- Since it uses standardized pairing requirements, Zigbee is also compatible with most modern devices and appliances, which reduces the overall cost of development for product owners and developers.
- Zigbee home automation also comes with DIY configurations, allowing greater customizability for both product engineers and end-users.
- Remote access with internet monitored control of a smart home.
- Smartphone app support for modulating smart home appliances and devices.
- Smart power management and control for supported products.
- Supports installation of additional certified security devices and protocols for enhanced home security.
- Allows lighting products to utilize dynamic lighting controls, often used for creating unique home and office environments.

Fig. Diagram showing Zigbee Home Automation

6.1.4 Insteon Home Automation

- Insteon domotics technology uses both power lines and radio frequency communication for smart home connectivity.
- The protocol requires all messages – received by a compatible device – to be checked for errors, and instills corrections, hence increasing the reliability of functions.
- However, Insteon protocol and its supported products were subject to certain instances of whitehat hacking in the past.
- In multiple cases, hackers were able to access Insteon users' smart homes and personal information. Those protocols have been retired since.

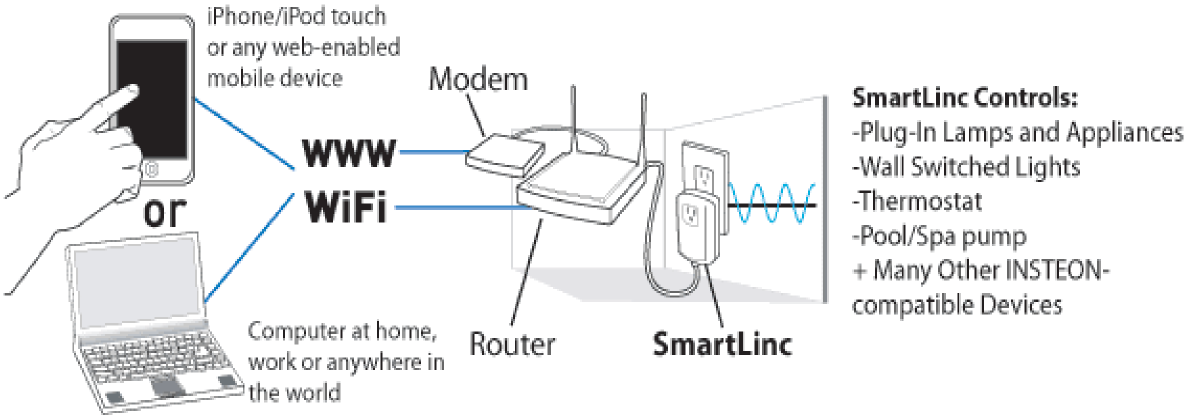


Fig.6.3 Diagram showing Insteon Home Automation

6.1.5 Wi-Fi Home Automation

- Given the penetration of Wi-Fi networks within local area networks, it is one of the most convenient protocol to work with for home automation devs.

- Wi-Fi protocols provide a ready-made infrastructure with an inherent ability to manage high quantities of data. Another advantage of using Wi-Fi protocol for home and office automation is the in-built AES 256-bit encryption.

- However, poor Wi-Fi speed and signal strength may bottleneck performance in larger domotics setups.

- Additionally, most homes and businesses are using 802.11n Wi-Fi standard, which is too power consuming for most IoT applications.

- The Wi-Fi Alliance recently announced a new 802.11ah standard. Termed Wi-Fi HaLow by the Alliance, the protocol can operate over greater distances than the existing 2.4Ghz and 5Ghz bands – purportedly up to 1 kilometers under ideal conditions – and has been especially developed for the implementation of IoT and automation.

- Although the real-world application of this protocol still needs to be tested thoroughly, the propensity of the 900Mhz band to operate amidst various interferences is well known.

- The sub-gigahertz operation of Wi-Fi 802.11ah allows better smart home, connected car and digital healthcare deployment.

- Thus, in theory, products utilizing the HaLow band will have significantly better coverage area. Also, the HaLow band can transmit at a minimum frequency of 150Kbps over channels up to 2Ghz.

- This means connected IoT devices can quickly resume their passive state after waking up to receive instructions – conserving more power – and hence overcoming one of the inherent limitations of current Wi-Fi networks.

- The limiting factor of HaLow band however is its transmission speeds, which remain in low tens of megabits per second and may not satisfy product planners who require higher bandwidths.

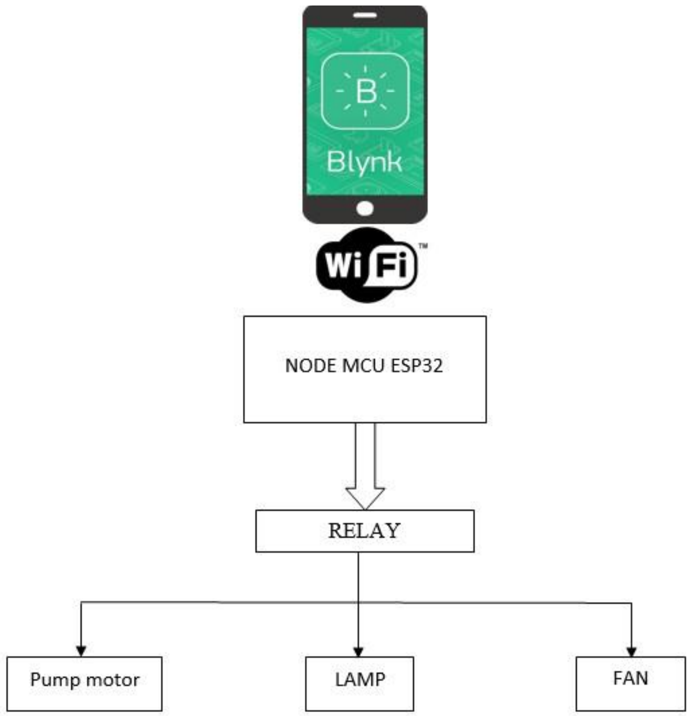


Fig.6.4 Diagram showing Wifi Home Automation

6.1.6 Bluetooth Home Automation

- Bluetooth networks have been occasionally utilized by smart home product developers as a home automation protocol, although the existing technology remains limited by its range and signal quality.
- Still, certain smart home devices can use Bluetooth signals to connect and perform basic tasks (depending on the level of support).
- Security Concerns: The major concern for product owners using Bluetooth networks for home automation is of security. The existing Bluetooth Low Energy (LE) technology is prone to sabotage due to multiple security exploits. Like:

- Passive eavesdropping, which allows a third device to intercept data exchanged between two paired devices. While BLE uses AES 128bit encryption to secure the data transfers, there are still some protocol deficiencies which can be exploited by hackers to intercept and decrypt personal data.
- Man-in-the-middle or MITM attacks, which allow a third-party device to insert itself between two legitimate devices, giving them the illusion that they are interconnected. The interception allows the malicious device to fool Gap Central and Gap Peripheral and tamper with the information being exchanged.
- Identity tracking, where a third party is able to track a specific user by associating a BLE device address with his device. In this instance, BLE does have a mechanism to periodically alter the device address to overcome this shortcoming.

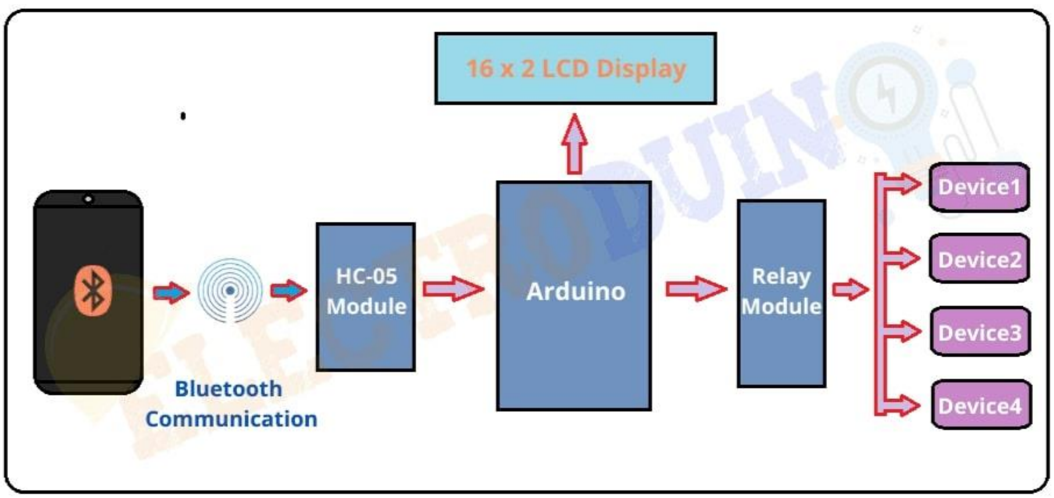


Fig.6.5 Diagram showing Bluetooth Home Automation

6.1.7 6lowPAN Home Automation

- The 6LoWPAN system is used for a variety of applications including wireless sensor networks.
- This form of wireless sensor network sends data as packets and using IPv6 - providing the basis for the name - IPv6 over Low power Wireless Personal Area Networks.
- 6LoWPAN provides a means of carrying packet data in the form of IPv6 over IEEE 802.15.4 and other networks.
- It provides end-to-end IPv6 and as such it is able to provide direct connectivity to a huge variety of networks including direct connectivity to the Internet.
- In this way, 6LoWPAN adopts a different approach to the other low power wireless sensor network solutions.
- 6LoWPAN is an open standard defined by the Internet Engineering Task Force, IETF in their document RFC 6282.
- The IETF is the standards body that defines many of the open standards used in the Internet including HTTP, TCP, UDP and many others.

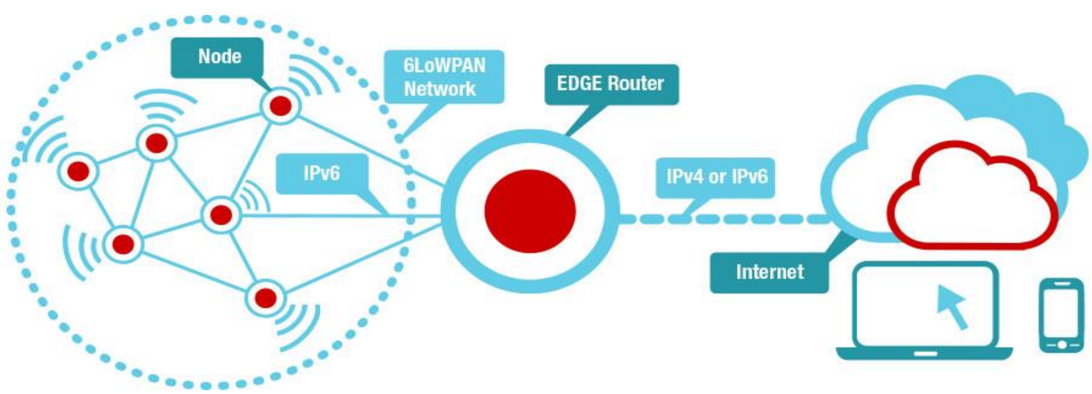


Fig.6.6 Diagram showing 6low PAN Home Automation

6.1.7 MQTT Home Automation

- MQTT (Message Queuing Telemetry Transport) is a lightweight messaging protocol that was developed by IBM and first released in 1999.
- It uses the pub/sub pattern and translates messages between devices, servers, and applications.
- The MQTT protocol was initially created in order to link sensors on oil pipelines with communications satellites, with an emphasis on minimal battery loss and bandwidth consumption.
- The connected devices in the MQTT protocol are known as “clients,” which communicate with a server referred to as the “broker.”
- The broker handles the task of data transmission between clients.
- Whenever a client (known as the “publisher”) wants to distribute information, it will publish to a particular topic, the broker then sends this information to any clients that have subscribed to that topic (known as “subscribers”).
- The publisher does not need any data on the number or the locations of subscribers.
- In turn, subscribers do not need any data about the publisher.
- Any client can be a publisher, subscriber, or both.
- The clients are typically not aware of each other, only of the broker that serves as the intermediary.
- This setup is popularly known as the “pub/sub model.”

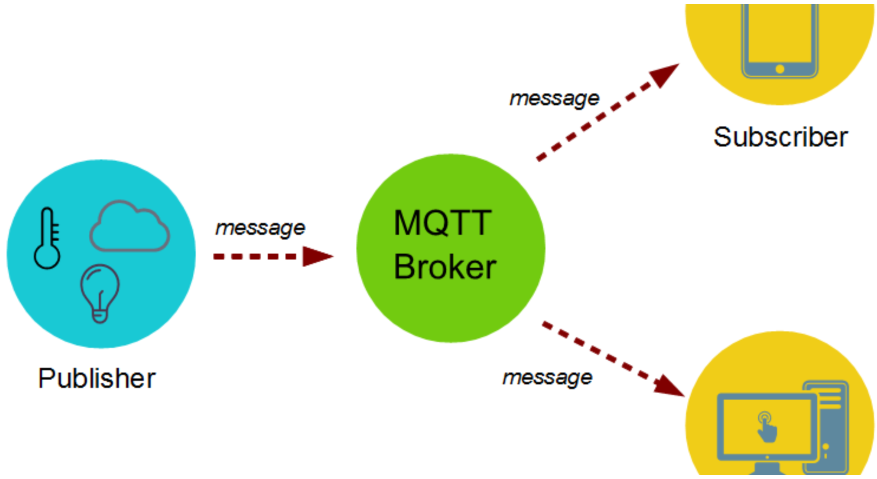


Fig.6.7 Diagram showing MQTTHome Automation

6.2 Models in IOT

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 :

6.2.1 Request & Response Model :

This model follows a client-server architecture.

- The client, when required, requests the information from the server. This request is usually in the encoded format.
- This model is stateless since the data between the requests is not retained and each request is independently handled.

- The server Categories the request, and fetches the data from the database and its resource representation. This data is converted to response and is transferred in an encoded format to the client. The client, in turn, receives the response.

- On the other hand — In Request-Response communication model client sends a request to the server and the server responds to the request. When the server receives the request it decides how to respond, fetches the data retrieves resources, and prepares the response, and sends it to the client.

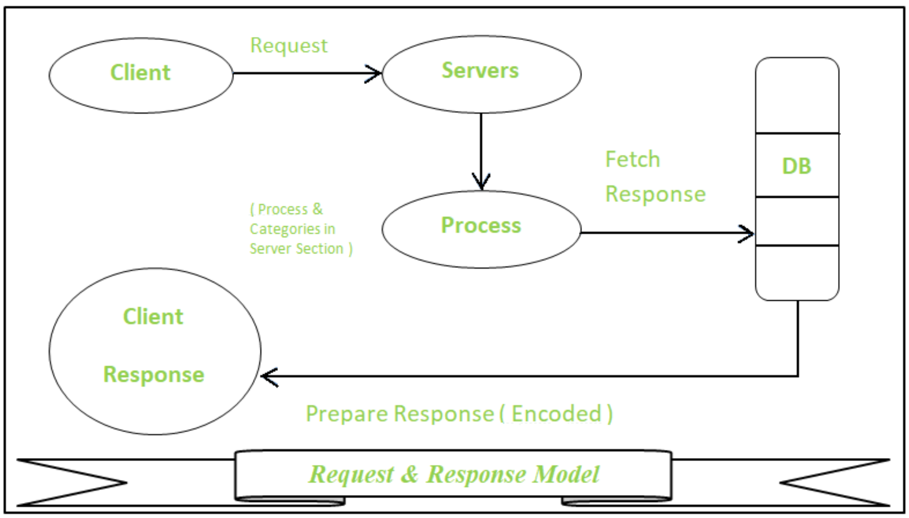


Fig.6.8 Diagram showing Request and Response Model

6.2.2 Publisher-Subscriber Model :

This model comprises three entities: Publishers, Brokers, and Consumers.

- Publishers are the source of data. It sends the data to the topic which are managed by the broker. They are not aware of consumers.

- Consumers subscribe to the topics which are managed by the broker.

- Hence, Brokers responsibility is to accept data from publishers and send it to the appropriate consumers. The broker only has the information regarding the consumer to which a particular topic belongs to which the publisher is unaware of.

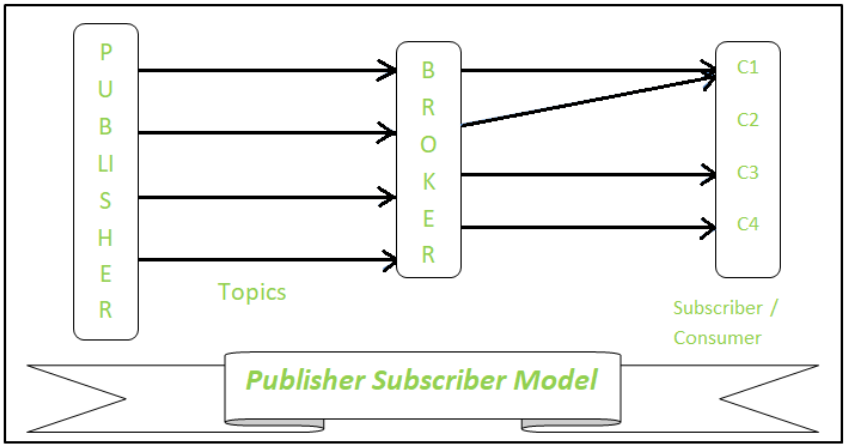


Fig.6.9 Diagram showing Publisher-Subscriber Model

6.2.3 Push- Pull Model :

The push-pull model constitutes data publishers, data consumers, and data queues.

- Publishers and Consumers are not aware of each other.
- Publishers publish the message/data and push it into the queue. The consumers, present on the other side, pull the data out of the queue. Thus, the queue acts as the buffer for the message when the difference occurs in the rate of push or pull of data on the side of a publisher and consumer.

- Queues help in decoupling the messaging between the producer and consumer. Queues also act as a buffer which helps in situations where there is a mismatch between the rate at which the producers push the data and consumers pull the data.

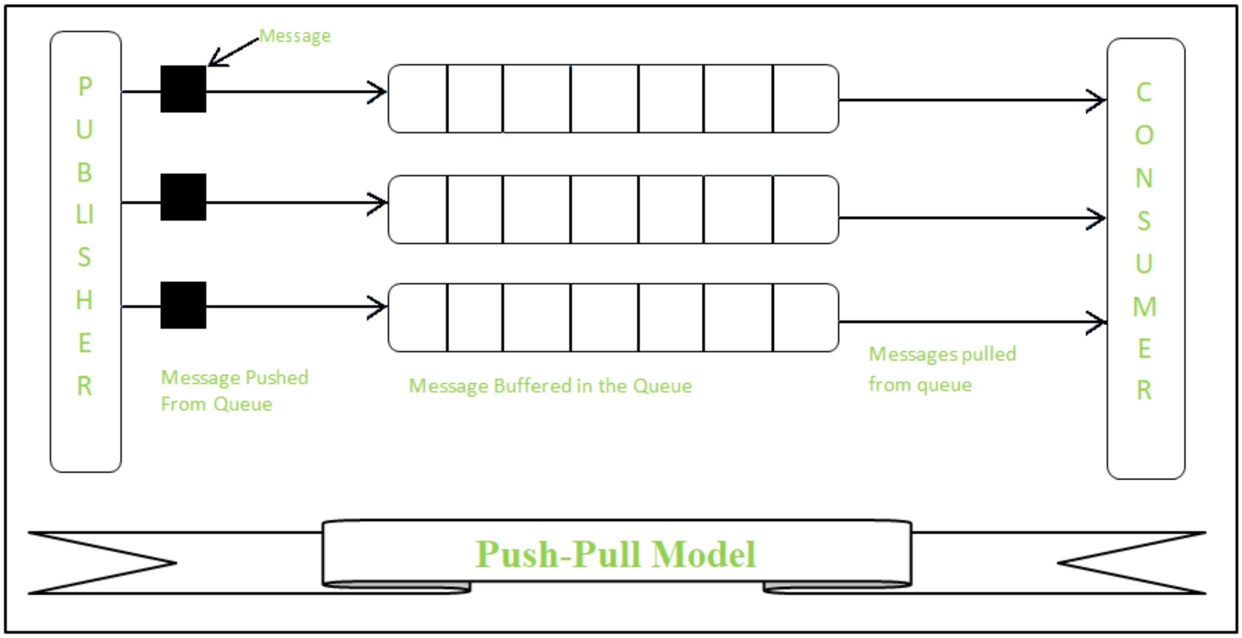


Fig.6.10 Diagram showing Push-Pull Model

6.2.4 Exclusive Pair :

- Exclusive Pair is the bi-directional model, including full-duplex communication among client and server. The connection is constant and remains open till the client sends a request to close the connection.

- The Server has the record of all the connections which has been opened.

- This is a state-full connection model and the server is aware of all open connections.

- WebSocket based communication API is fully based on this model.

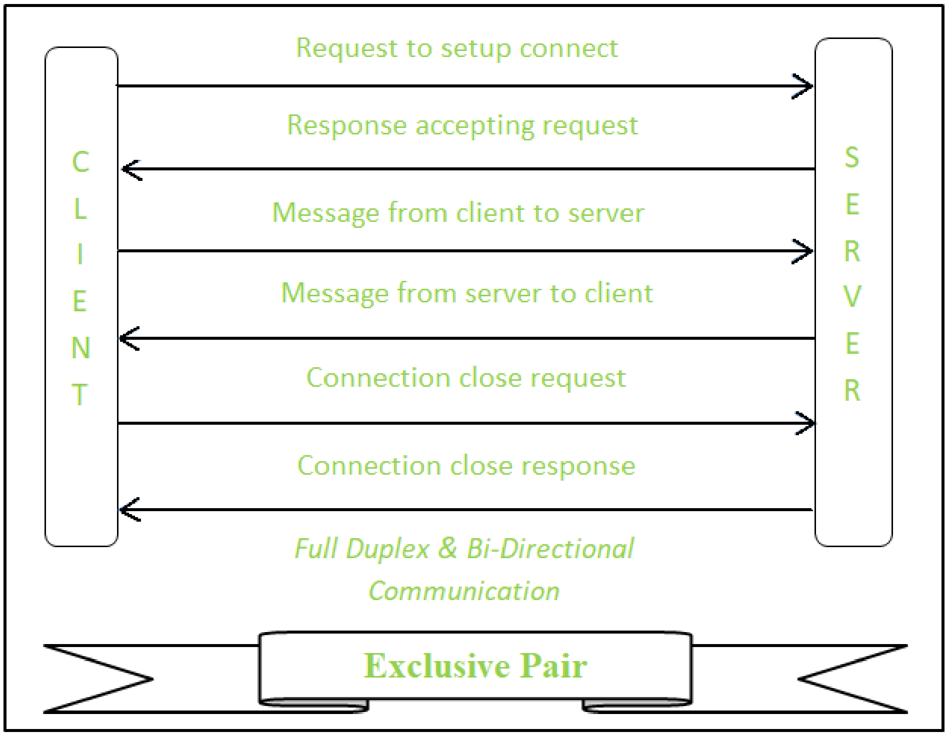


Fig.6.11 Diagram showing Exclusive Pair

6.3 Protocols in Embedded Systems

6.3.1 Inter System Protocol

The inter-system protocol using to communicate the two different devices. Like communication between computer to microcontroller kit. The communication is done through an inter bus system.

The different categories of intersystem protocol mainly include the following.

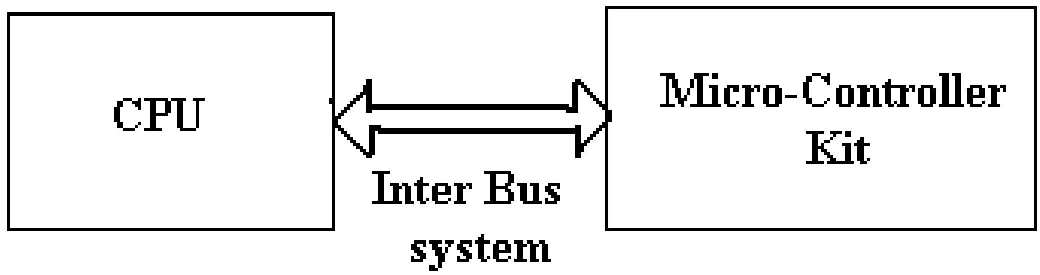


Fig.6.12 Diagram showing Inter System Protocol

- **UART Protocol *:*** UART stands for a universal asynchronous transmitter and receiver. UART Protocols is a serial communication with two wired protocols. The data cable signal lines are labeled as Rx and Tx. Serial communication is commonly used for transmitting and receiving the signal. It is transferred and receives the data serially bit by bit without class pulses. The UART takes bytes of data and sends the individual bits in a sequential manner. Ex: Emails, SMS

- **USART Protocol:** USART stands for a universal synchronous and asynchronous transmitter and receiver. It is a serial communication of a two-wire protocol. The data cable signal lines are labeled as Rx and TX. This protocol is used to transmitting and receiving the data byte by byte along with the clock pulses. It is a full-duplex protocol that means transmitting and receiving data simultaneously to different board rates. Different devices communicate with microcontroller to this protocol. Ex:-Telecommunications.

- **USB Protocol:** USB stands for universal serial bus. Again it is a serial communication of two-wire protocol. The data cable signal lines are labeled D+ and D-. This protocol is used to communicate with the system peripherals.USB protocol is used to send and receive the data serially to the host and peripheral devices.USB communication requires driver software that is based on the functionality of the system.USB devices can transfer data on the bus without any request on the host computer. Ex: Mouse, Keyboard

6.3.2 Intra System Protocol

The Intra system protocol is used to communicate the two devices within the circuit board. While using these intra system protocols, without going to intrasystem protocols we will expand the peripherals of the microcontroller. The circuit complexity and power consumption will be increased by using the intrasystem protocol. Using intra system protocols circuit complexity and power consumption, the cost is decreased and it is very secure to accessing the data.

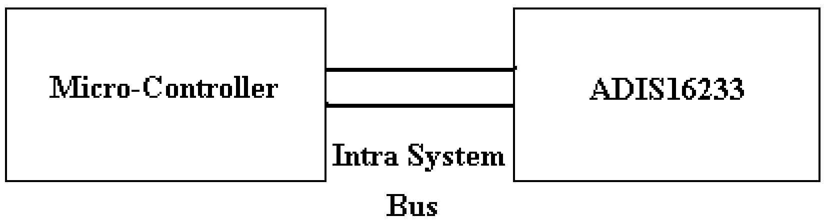


Fig.6.13 Diagram showing Intra System Protocol

- **I2C Protocol:** I2C stands for the inter-integrated circuit and it requires only two wires connecting all peripherals to the microcontroller. I2C requires two wires SDA (serial data line) and SCL (serial clock line) to carry information between devices. It is a master to a slave communication protocol. Each slave has a unique address. The master device sends the address of the target slave device and reads/writes the flag. The address matches any slave device that the device is ON, the remaining slave devices are disabled mode.

- **I2C Pull-up Resistors:** Both SDA and SCL lines are open-drain drivers. It can drive output low canot driver it high. For the lines to be able to go high you must provide pull-up resistors.

- **SPI Protocol:** SPI stands for the serial peripheral interface. It is one of the serial communication protocol developed by Motorola. Sometimes SPI protocol is also called a 4-wire protocol. It requires four wires MOSI, MISO, SS, and SCLK.SPI protocol used to communicate the master and slave devices. The master first configures the clock using frequency.The master then selects the particular slave device for communication by pulling the chip select button. That particular device is selected and starts the communication between the master and that particular slave. The master selects only one slave at a time. It is a full-duplex communication protocol. Not limited to 8-bit words in the case of bit transferring.

- **CAN Protocol:** CAN stands for the controller area network. It is a serial communication protocol. It requires two wires CAN High (H+) and CAN low (H-). It was developed by the Robert bosh company in 1985 for in-vehicle networks. It is based on a message-oriented transmission protocol.

6.4 Models in Embedded Systems

6.4.1 Analytic Models

Analysis involves the use of mathematical approaches to create high-level abstractions of system properties, most notably performance. Analytic models are typically succinct mathematical equations that may be evaluated for any set of conditions to predict system properties. Different analytic models are typically required to express different categories of system properties.

Analytic techniques include:

- Plumbing diagram that examine steady-state capacities and flow rates (e.g., bus bandwidth vs. average message traffic)

- Queuing models that estimate backlogs and latencies (e.g., a model to determine expected queue length).

- Probabilistic models that compute the likelihood of events (e.g., computing probability of message loss due to collision on a CSMA network, occurrence rate of double-bit errors given a single-bit DRAM error rate).
- Empirical models that extrapolate past system characteristics to similar new systems (e.g., failure rate rules-of-thumb)

Analytic models typically have the advantage of being readily grasped. In many cases, analytic models can be constructed in a few hours. They are often relatively cheap to evaluate, and can give a reasonable estimate of system characteristics quickly. Also, analytic models can provide insight into the dynamics of the system to help guide design activities.

On the other hand, analytic models can only be created by people having keen insight into the system being designed. There is always a risk that the system properties being analyzed are not the ones that will ultimately dominate the system's characteristics. Also, in some cases too many simplifying assumptions must be made in order to create tractable models.

So, while analysis can provide quick answers during the exploration phase of a design, it is possible that the answers are not good approximations to reality. It is also possible that while analysis answers questions correctly, it may not provide insight into whether the right questions have been asked.

6.4.2 Simulation Models

Simulation involves the use of executable computer programs to demonstrate emergent system behavior. Building an executable model at even a high level of abstraction forces the designer to think through issues that otherwise might be swept under the rug with a non-executable specification technique. More than one simulation technique and corresponding model are often desirable for any particular system, depending on the aspects that must be studied.

Simulation techniques include:

- Discrete Event simulations for queuing simulations and communication protocol performance, including execution of queue- based models and Petri nets
- Cycle-based simulations for processors, memories, and network performance
- Continuous time simulators for non-digital processes
- Coupled analytic models in a spread-sheet type environment to explore their coupled behavior by "what-if" evaluation trials.

Simulations require a "workload", or stimulus at an appropriately high level of abstraction. Simulations may be fed by:

- **Random** inputs, with varying probability distributions that are used to provide system stimuli according to predefined criteria.
- **Abstract** workloads, in which characteristics of the workload are abstracted to, for example, a set of periodic and aperiodic events (often with "noise" in the form of probabilistic timing jitter and drift).
- **Traces**, in which stimuli are provided via time-stamped data files from the output of other simulations, instrumented prototype, or production systems.

Simulations provide an important intermediate capability between analytic models and actual prototypes. By building a model of the system and executing it, designers can see what behavior emerges. With appropriate instrumentation and attention, a simulation can reveal unexpected interactions and performance bottlenecks that are missed by analysis. In particular, simulations are valuable for studying "fine-grain", detailed interactions that deal with specific sequences of events rather than the broad-brush steady-state approach typical of analytic methods.

Simulations can also be superior to prototypes in many cases. It is relatively simple to create arbitrary initial conditions (controllability) and detailed monitoring devices (observability) in a simulation. Controllability is important to investigate conditions that are unlikely to happen in practice, or are too expensive to create in the laboratory more than once. With the complete controllability offered by digital computer simulations, it is generally easy to repeat experiments in order to evaluate potential design changes.

Simulations also offer superior observability, since any state within the model is available as a value in some memory location. An important implication of complete observability is that it is usually straightforward to freeze operation of a system and capture the complete state when an infrequent bug occurs.

So, simulation provides an intermediate step between quick tradeoff studies performed by analysis and detailed validation provided by prototyping.

Chapter 7
Components

7.1 Introduction

An embedded system is a computer system—a combination of a computer processor, computer memory, and input/output peripheral devices—that has a dedicated function within a larger mechanical or electronic system. It is embedded as part of a complete device often including electrical or electronic hardware and mechanical parts. Because an embedded system typically controls physical operations of the machine that it is embedded within, it often has real-time computing constraints. Embedded systems control many devices in common use today. In 2009 it was estimated that ninety-eight percent of all microprocessors manufactured were used in embedded systems.

Modern embedded systems are often based on microcontrollers (i.e. microprocessors with integrated memory and peripheral interfaces), but ordinary microprocessors (using external chips for memory and peripheral interface circuits) are also common, especially in more complex systems. In either case, the processor(s) used may be types ranging from general purpose to those specialized in a certain class of computations, or even custom designed for the application at hand. A common standard class of dedicated processors is the digital signal processor (DSP).

Since the embedded system is dedicated to specific tasks, design engineers can optimize it to reduce the size and cost of the product and increase the reliability and performance. Some embedded systems are mass-produced, benefiting from economies of scale.

Embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations like traffic light controllers, programmable logic controllers, and large complex systems like hybrid vehicles, medical imaging systems, and avionics. Complexity varies from low, with a single microcontroller chip, to very high with multiple units, peripherals and networks mounted inside a large equipment rack.

7.1.1 Sensors/Devices

First, sensors or devices help in collecting very minute data from the surrounding environment. All of this collected data can have various degrees of complexities ranging from a simple temperature monitoring sensor or a complex full video feed.

A device can have multiple sensors that can bundle together to do more than just sense things. For example, our phone is a device that has multiple sensors such as GPS, accelerometer, camera but our phone does not simply sense things.

The most rudimentary step will always remain to pick and collect data from the surrounding environment be it a standalone sensor or multiple devices.

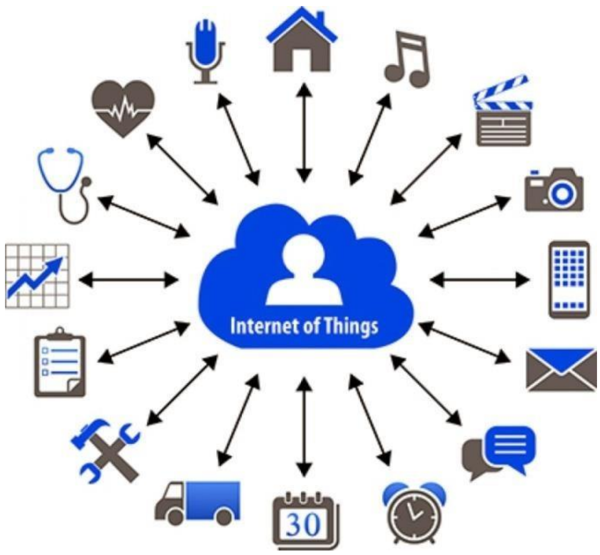


Fig.7.1: Sensors/Devices

7.1.2 Connectivity

Next, that collected data is sent to a cloud infrastructure but it needs a medium for transport.

The sensors can be connected to the cloud through various mediums of communication and transports such as cellular networks, satellite networks, Wi-Fi, Bluetooth, wide-area networks (WAN), low power wide area network and many more.

Every option we choose has some specifications and trade-offs between power consumption, range, and bandwidth. So, choosing the best connectivity option in the IOT system is important.

Fig.7.2: Connectivity

7.1.3 Data Processing

Once the data is collected and it gets to the cloud, the software performs processing on the acquired data.

This can range from something very simple, such as checking that the temperature reading on devices such as AC or heaters is within an acceptable range. It can sometimes also be very complex, such as identifying objects (such as intruders in your house) using computer vision on video.

But there might be a situation when a user interaction is required, example- what if when the temperature is too high or if there is an intruder in your house? That's where the user comes into the picture.

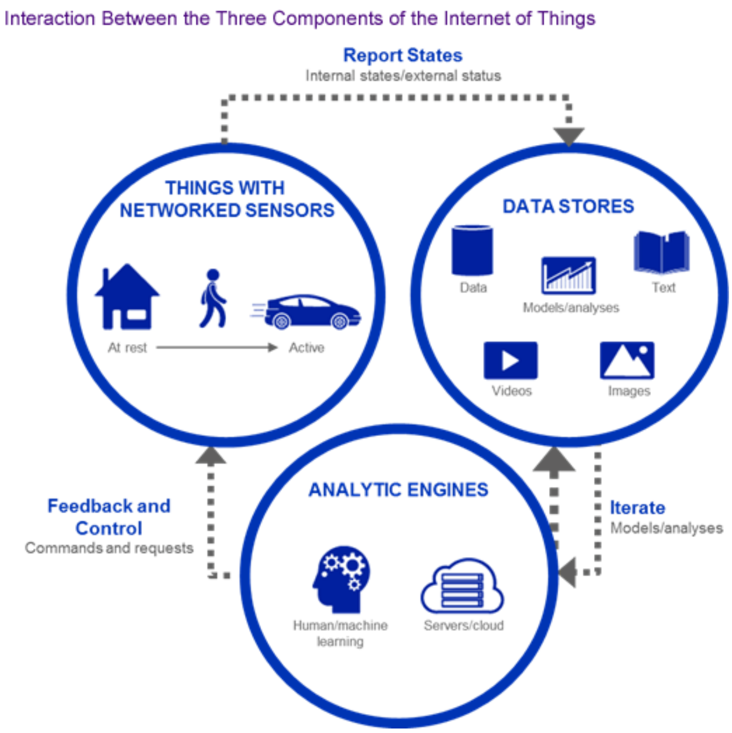


Fig.7.3: Data Processing

7.1.4 User Interface

Next, the information made available to the end-user in some way. This can achieve by triggering alarms on their phones or notifying through texts or emails.

Also, a user sometimes might also have an interface through which they can actively check in on their IOT system. For example, a user has a camera installed in his house, he might want to check the video recordings and all the feeds through a web server.

7.2 Applications

Embedded systems are commonly found in consumer, industrial, automotive, home appliances, medical, and telecommunication, commercial and military applications.

Telecommunications systems employ numerous embedded systems from telephone switches for the network to cell phones at the end user. Computer networking uses dedicated routers and network bridges to route data.

Consumer electronics include MP3 players, television sets, mobile phones, video game consoles, digital cameras, GPS receivers, and printers. Household appliances, such as microwave ovens, washing machines and dishwashers, include embedded systems to provide flexibility, efficiency and features. Advanced HVAC systems use networked thermostats to more accurately and efficiently control temperature that can change by time of day and season. Home automation uses wired- and wireless-networking that can be used to control lights, climate, security, audio/visual, surveillance, etc., all of which use embedded devices for sensing and controlling.

Transportation systems from flight to automobiles increasingly use embedded systems. New airplanes contain advanced avionics such as inertial guidance systems and GPS receivers that also have considerable safety requirements. Various electric motors — brushless DC motors, induction motors and DC motors — use electronic motor controllers. Automobiles, electric vehicles, and hybrid vehicles increasingly use embedded systems to maximize efficiency and reduce pollution. Other automotive safety systems using embedded systems include anti-lock braking system (ABS), Electronic Stability Control (ESC/ESP), traction control (TCS) and automatic four-wheel drive.

Medical equipment uses embedded systems for monitoring, and various medical imaging (PET, SPECT, CT, and MRI) for non-invasive internal inspections. Embedded systems within medical equipment are often powered by industrial computers.

Embedded systems are used for safety-critical systems. Unless connected to wired or wireless networks via on-chip 3G cellular or other methods for IoT monitoring and control purposes, these systems can be isolated from hacking and thus be more secure. For fire safety, the systems can be designed to have a greater ability to handle higher temperatures and continue to operate. In dealing with security, the embedded systems can be self-sufficient and be able to deal with cut electrical and communication systems.

Miniature wireless devices called motes are networked wireless sensors. Wireless sensor networking makes use of miniaturization made possible by advanced IC design to couple full wireless subsystems to sophisticated sensors, enabling people and companies to measure a myriad of things in the physical world and act on this information through monitoring and control systems. These motes are completely self-contained and will typically run off a battery source for years before the batteries need to be changed or charged.

7.3 Characteristics

Embedded systems are designed to do some specific task, rather than be a general-purpose computer for multiple tasks. Some also have real-time performance constraints that must be met, for reasons such as safety and usability; others may have low or no performance requirements, allowing the system hardware to be simplified to reduce costs.

Embedded systems are not always standalone devices. Many embedded systems consist of small parts within a larger device that serves a more general purpose. For example, the Gibson Robot Guitar features an embedded system for tuning the strings, but the overall purpose of the Robot Guitar is, of course, to play music. Similarly, an embedded system in an automobile provides a specific function as a subsystem of the car itself.

The program instructions written for embedded systems are referred to as firmware, and are stored in read-only memory or flash memory chips. They run with limited computer hardware resources: little memory, small or non-existent keyboard or screen.

7.4 Processors in Embedded Systems

Examples of properties of typical embedded computers, when compared with general-purpose counterparts, are low power consumption, small size, rugged operating ranges, and low per-unit cost. This comes at the price of limited processing resources.

Numerous microcontrollers have been developed for embedded systems use. General-purpose microprocessors are also used in embedded systems, but generally, require more support circuitry than microcontrollers.

7.4.1 Ready Made Computer Boards

PC/104 and PC/104+ are examples of standards for ready-made computer boards intended for small, low-volume embedded and ruggedized systems, mostly x86-based. These are often physically small compared to a standard PC, although still quite large compared to most simple (8/16-bit) embedded systems. They often use DOS, Linux, NetBSD, or an embedded real-time operating system such as MicroC/OS-II, QNX or VxWorks. Sometimes these boards use non-x86 processors.

In certain applications, where small size or power efficiency are not primary concerns, the components used may be compatible with those used in general-purpose x86 personal computers. Boards such as the VIA EPIA range help to bridge the gap by being PC-compatible but highly integrated, physically smaller or have other attributes making them attractive to embedded engineers. The advantage of this approach is that low-cost commodity components may be used along with the same software development tools used for general software development. Systems built in this way are still regarded as embedded since they are integrated into larger devices and fulfill a single role. Examples of devices that may adopt this approach are ATMs and arcade machines, which contain code specific to the application.

However, most ready-made embedded systems boards are not PC-centered and do not use the ISA or PCI busses. When a system-on-a-chip processor is involved, there may be little benefit to having a standardized bus connecting discrete components, and the environment for both hardware and software tools may be very different.

One common design style uses a small system module, perhaps the size of a business card, holding high density BGA chips such as an ARM-based system-on-a-chip processor and peripherals, external flash memory for storage, and DRAM for runtime memory. The module vendor will usually provide boot software and make sure there is a selection of operating systems, usually including Linux and some real-time choices. These modules can be manufactured in high volume, by organizations familiar with their specialized testing issues, and combined with much lower volume custom mainboards with application-specific external peripherals.

Implementation of embedded systems has advanced so that they can easily be implemented with already-made boards that are based on worldwide accepted platforms. These platforms include, but are not limited to, Arduino and Raspberry Pi.

7.4.2 ASIC and FPGA solutions

A common array for very-high-volume embedded systems is the system on a chip (SoC) that contains a complete system consisting of multiple processors, multipliers, caches and interfaces on a single chip. SoCs can be implemented as an application-specific integrated circuit (ASIC) or using a field-programmable gate array (FPGA).

7.5 Peripherals

Embedded systems talk with the outside world via peripherals, such as:

- Serial Communication Interfaces (SCI): RS-232, RS-422, RS-485, etc.
- Synchronous Serial Communication Interface: I2C, SPI, SSC and ESSI (Enhanced Synchronous Serial Interface)
- Universal Serial Bus (USB)
- Multimedia Cards (SD cards, Compact Flash, etc.)

- Networks: Ethernet, LonWorks, etc.
- Fieldbuses: CAN-Bus, LIN-Bus, PROFIBUS, etc.
- Timers: PLL(s), Capture/Compare and Time Processing Units
- Discrete IO: aka General-Purpose Input/Output (GPIO)
- Analog to Digital/Digital to Analog (ADC/DAC)
- Debugging: JTAG, ISP, BDM Port, BITP, and DB9 ports.

7.6 Tools

As with other software, embedded system designers use compilers, assemblers, and debuggers to develop embedded system software. However, they may also use some more specific tools:

- In circuit debuggers or emulators (see next section).
- Utilities to add a checksum or CRC to a program, so the embedded system can check if the program is valid.
- For systems using digital signal processing, developers may use a math workbench to simulate the mathematics.
- System-level modelling and simulation tools help designers to construct simulation models of a system with hardware components such as processors, memories, DMA, interfaces, buses and software behaviour flow as a state diagram or flow diagram using configurable library blocks. Simulation is conducted to select the right components by performing power vs. performance trade-off, reliability analysis and bottleneck analysis. Typical reports that help a designer to make architecture decisions includes application latency, device throughput, device utilization, power consumption of the full system as well as device-level power consumption.
- A model-based development tool creates and simulates graphical data flow and UML state chart diagrams of components like digital filters, motor controllers, communication protocol decoding and multi-rate tasks.
- Custom compilers and linkers may be used to optimize specialized hardware.
- An embedded system may have its own special language or design tool or add enhancements to an existing language such as Forth or Basic.
- Another alternative is to add a real-time operating system or embedded operating system.

- Modelling and code generating tools often based on state machines.
- Software tools can come from several sources:
- Software companies that specialize in the embedded market
- Ported from the GNU software development tools.
- Sometimes, development tools for a personal computer can be used if the embedded processor is a close relative to a common PC processor.

As the complexity of embedded systems grows, higher-level tools and operating systems are migrating into machinery where it makes sense. For example, cellphones, personal digital assistants and other consumer computers often need significant software that is purchased or provided by a person other than the manufacturer of the electronics. In these systems, an open programming environment such as Linux, NetBSD, OSGi or Embedded Java is required so that the third-party software provider can sell to a large market.

Embedded systems are commonly found in consumer, cooking, industrial, automotive, and medical applications. Some examples of embedded systems are MP3 players, mobile phones, video game consoles, digital cameras, DVD players, and GPS. Household appliances, such as microwave ovens, washing machines and dishwashers, include embedded systems to provide flexibility and efficiency.

7.7 Process and People

Let's start defining what a process is: A series of actions or steps taken in order to achieve a particular end. A process can be associated to thing or to a business. In the case of a thing can be the process of a person running. The person will have as a goal to run specific distance in a specific time. Using an IoT device (e.g. fitbit tracker) the person can track in a daily basis distance, time, calories, water consumption, etc. until the goal is accomplished. The person can use all the information gathered by the device to monitor his or her progress during the process and make the necessary adjustment in order to reach the goal. In the past people were relying on manual tracking which was open to human errors.

In the case of a business process, it can be a manufacturing process for example service maintenance. In a lot of companies, manufacturing managers have specific dates defined to do preventive maintenance to a machine. But depending on the use of the machine, sometimes it is too late to perform a preventive maintenance, forcing the manager to invest more time and money in a major maintenance. With a sensor in the machine to check the condition of the different parts, now the machine can provide real time information to the manager and with this information he can determine if it is necessary to execute the preventive maintenance before the original set date and this way save time and money because major maintenance will not be necessary.

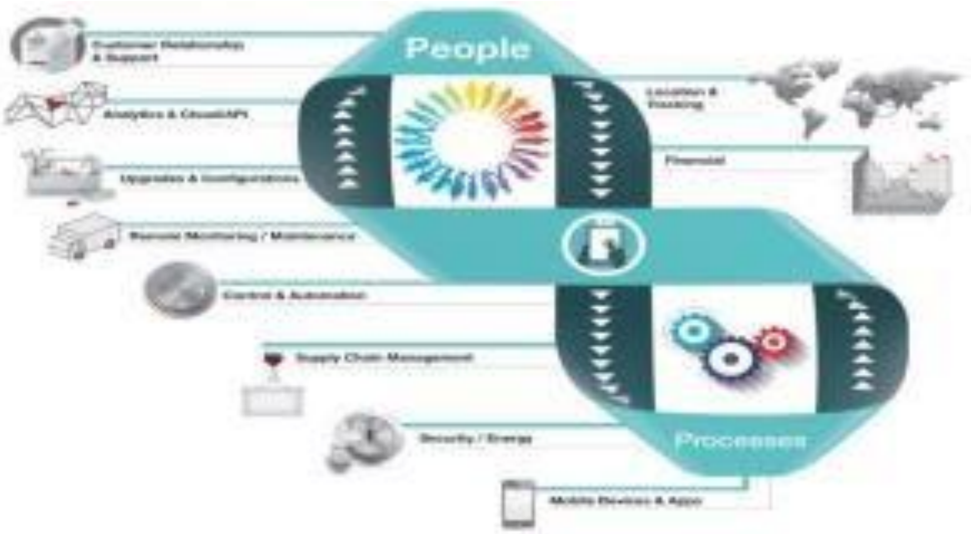


Fig.7.4: People and Process in IOT

Chapter 8

THE CONTROLLER

8.1 Controller

- Your IoT device may be smaller than a coin or larger than a refrigerator.
- It may perform a simple sensing function and send raw data back to a control center
- It may combine data from many sensors, perform local data analysis, and then take action. Additionally, your device could be remote and standalone or be co-located within a larger system.
- Regardless of the function, environment, or location, your IoT device requires two components, a brain and connectivity.
- The “brain” provides local control (or decision-making). Your device’s function will determine the size and capabilities of the brain component.
- Connectivity is needed to communicate with external control. The environment and location of your device will determine how it connects.
- The Internet of Things (IoT) is attracting a huge amount of attention these days, although much of it seems directed at simplistic use cases or the shortcomings of today’s solutions.
- This is because the industry is missing consensus on the overall IoT architecture, standards, and reference models. While all of the leading vendors are working on their strategies for networked

- "Things," few have evolved their thinking to a point where they are willing to publicly discuss key constructs of IoT architecture.

- The unique challenge of IoT is the high volume of sensory data and the complex challenges this creates in managing and leveraging this data.

- This challenge extends to enterprises of all sizes that are interested in connecting more effectively with customers and driving higher levels of efficiency across the value chain.

- Consumers will largely be insulated from the IoT data management and analysis challenges by applications that hide this complexity. Nevertheless, IoT begins with front-end sensor data and ends with back-end systems using analytics and decision-making capability to drive intelligent actions.

- What's missing from the IoT discussion is how to best manage the movement of this data from the front-end to the back-end. This is where the concept of a middle tier supporting controllers and gateways fits in.

- Controllers and gateways are middle-tier components that primarily collect information from front-end devices. However, the role of controllers (consumer focused) and gateways (commercial focus) in consuming this information varies widely.

- In some cases a gateway will primarily be focused on data collection, data aggregation, and then transmission of the data to a back-end system.

- The periodicity associated with this data collection, aggregation, and transmission will vary from real-time streaming and queuing to batch processing. In other situations, the gateway or

- controller will also analyze and act on the data based on a predefined policy in order to automatically interact with front-end devices.

- The location of controllers and gateways will also vary based on the intelligence and constraints of the network connection to the front-end devices, environmental factors, and security concerns. Common networking between edge devices and controllers/gateways includes BT, CAN, LAN, WAN, WIFI, USB, and Internet.

8.2 The "Brain" of IOT

- Your IoT device will most likely use a microcontroller as its brain.

- Think of a microcontroller as a small computer with a microprocessor core, memory, and input/output (I/O) ports.

- The microprocessor core of your microcontroller is a central processing unit.

- It handles all the number crunching and local data manipulation and decision-making.

- The memory includes Read Only Memory (ROM) and Random Access Memory (RAM).

- ROM stores the microcontroller's software program.

- RAM stores and receives data while also supporting number crunching.

- The final microcontroller component, the I/O ports, may be either digital or analog.

- The input ports collect data from sensors. While the outputs support any necessary actuation or local control in the IoT device.

- Usually, microcontrollers control various devices or subsystems within embedded applications.
- By integrating the microprocessor, memory, and input/outputs, microcontrollers reduce cost and make development easier.
- This makes it more affordable and less complicated to control many IoT devices.

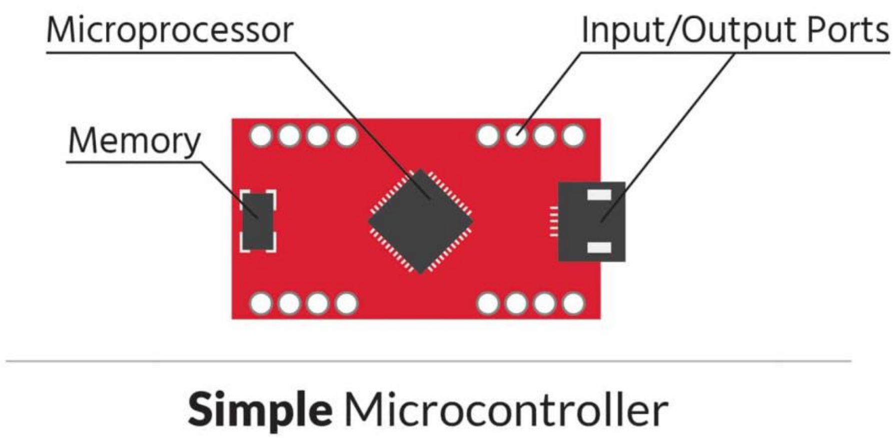


Fig.8.1 Diagram showing Simple Microcontroller

8.3 Adafruit.IO

- Adafruit.io is a cloud service - that just means we run it for you and you don't have to manage it.
- We can connect to it over the Internet.
- It's meant primarily for storing and then retrieving data but it can do a lot more than just that.

- Here at Adafruit, we manufacture, support and sell all of these amazing sensors, LEDs, and robotics.
- So, naturally, we wanted a good way to interact with them over the Internet.
- We've covered a bunch of great services for data logging and communicating with your controller over the web, but we wanted a service that fit the needs of the Prototyping & Maker communities AND was designed for engineers of all skill levels.
- So we decided to build our own system - that is how Adafruit IO got started.
- Like many of the services we've introduced, we maintain both MQTT and REST APIs, which is how you will be communicating with Adafruit IO over the Internet.
- But you don't have to be an expert programme.
- We've built robust client libraries with lots of examples, so you can probably start with some ready-to-go code.
- Once you've got a device connected, you can control and monitor using configurable dashboards.
- The web dashboards come with a dozen widgets that allow for easy 2-way interaction with your devices. You'll get buttons, gauges, maps, sliders.
- Outside of the dashboard, you can create triggers to, say, email you when the water sensor value goes above 9000.
- It is available and free for anyone with an Adafruit account.
- For the vast number of people starting out with IoT, the basic account will do everything they need.
- Adafruit IO plus members get some nifty extras and more bandwidth.

8.4 Dashboard

- A dashboard is a tool used for information management and business intelligence.

- Much like the dashboard of a car, data dashboards organize, store, and display important information from multiple data sources into one, easy-to-access place.

- Using data visualization, dashboards uniquely communicate metrics visually to help users understand complex relationships in their data.

- In a data dashboard, it's easier to draw parallels between different but related metrics, identify trends, and head off potential challenges hidden in an organization's data.

- With the help of smartphones, tablets, and other mobile technology, dashboards are also used to convey relevant information to audiences at any time and in any place.

- The best dashboards are customized, secured, and shared with their intended end-users.

Fig.8.2 Diagram showing Dashboards

- An executive dashboard gathers and holds information that top-level stakeholders need to run a company, business, or organization.

- Executive dashboards function much like business dashboards, except the information in them should cater specifically to the needs and expectations of executives.

- Executives only have so much bandwidth to gather and understand information, which means they need access to the information they need when they need it. Some key benefits include:

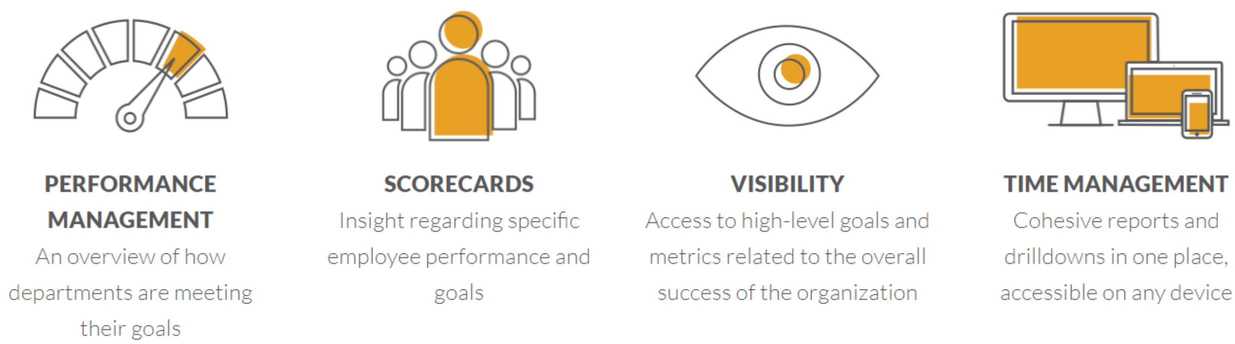


Fig.8.3 Diagram showing Key Benefits

- PIs (Key Performance Indicators) are the heart and soul of your organization's performance.
- They are the stepping stones that will guide your business to long-term success, so tracking and comparing them in one place is vital.
- KPIs should be measurable, tangible metrics that let each employee, team, and department understand how their performance influences the success of the organization and your KPI dashboard is where these metrics are stored.

- A successful KPI dashboard should set tangible goals and targets for each department, facilitate accountability within each department and provide real-time updates on goals and progress.
- Much like KPI dashboards, project dashboards track tangible goals; however, the “goal” of a project dashboard isn’t about hitting a sales quota or increasing marketing revenue by a certain margin.
- Instead, project dashboards track specific metrics related to the progress and complete of a project.

8.5 Steps to design controller

As you know that when we make an IOT project or any embedded project then we have to display the output on a screen. For this, we design an application and it takes a long time so you can prefer to use node-red.

It is an online server application where you made a presentable dashboard in less time to display the output of a project and it is helpful for hackathons where you design an impressive and presentable dashboard of your project to represent the project in front of your judges/mentor.

Before creating your own dashboard, do the following:

- **Step 1: Structure your Solution** – to build an IoT application you need to create a structure encompassing: devices, variables, dashboards, and alerts.

- **Step 2: Select a Device** – devices are the individual hardware selected to sense data in a particular environment. It is very important that the device is selected based on the environment and the requirements you're looking for it to complete. Also please consider your connection protocol of choice, be it WiFi, Bluetooth, Sigfox, LoRa, Ethernet, etc.

- **Step 3: Sign Up or Log In at the site to build controller-** The first thing you need to do is, in case you have not registered yet, create a new account and Sign Up to our platform. Otherwise, if you already have an account, you just need to Log In to move to your IoT Dashboard. Primarily you'll notice that when you first access to your IoT Dashboard, a new window appears to introduce yourself to the platform and specifically, to the IoT Dashboard.

- **Step 4:Starting with IoT Dashboard-** The IoT Dashboard is the place at the site where all the data generated by your things will be displayed. Depending on which kind of information you want to see you may prefer one widget or another. Before start building your IoT Dashboard you should have at least one thing connected using our platform. If you want to start connecting things fast, you should not miss this quick guide of how to connect your first thing in 1 minute. Why? Because you need data to display! Once you have done that, we can move to the option "Edit Dashboard", on the lower right section of your screen. The icon is yellow and this tool is the one we are going to use most during this post, so keep an eye on it!

- **Step 5: Create a Widget-** "Edit Dashboard" there is a a blue "Add Widget" icon. So now we are going to click on it and a pop-up you will have to work with will appear. At this point, you can choose the kind of widget you want your data to be displayed on your IoT Dashboard. You have different options, such as: Map, Doughnut, Lines, Bars, Logs, Value, Pie and Polar Area. Depending on the option you choose in "Values Range" you'll find one or the other. Finally, you can add your new widget to your IoT Dashboard.

- **Step 6: Customize your IoT Dashboard-** One of the coolest things you can do with your IoT Dashboard is to add, remove and replace your widgets easily. To do it you just need to click on "Edit Dashboard" again and just touch one of the widgets. Now you can change its place just moving it, change its size, and also the kind of widget you want it to be! Once you finish editing your IoT Dashboard do not forget to save it!

- **Step 7: Create an Image-**You may think that's all we can offer, but right now we do offer more! Now you can add images to your IoT Dashboard. Go to Edit Dashboard button and Add Widget. Once you are in the pop-up window, in the "Data Source" field you have to choose the option "External Image". Here you will be able to add images from your computer or directly from an URL.

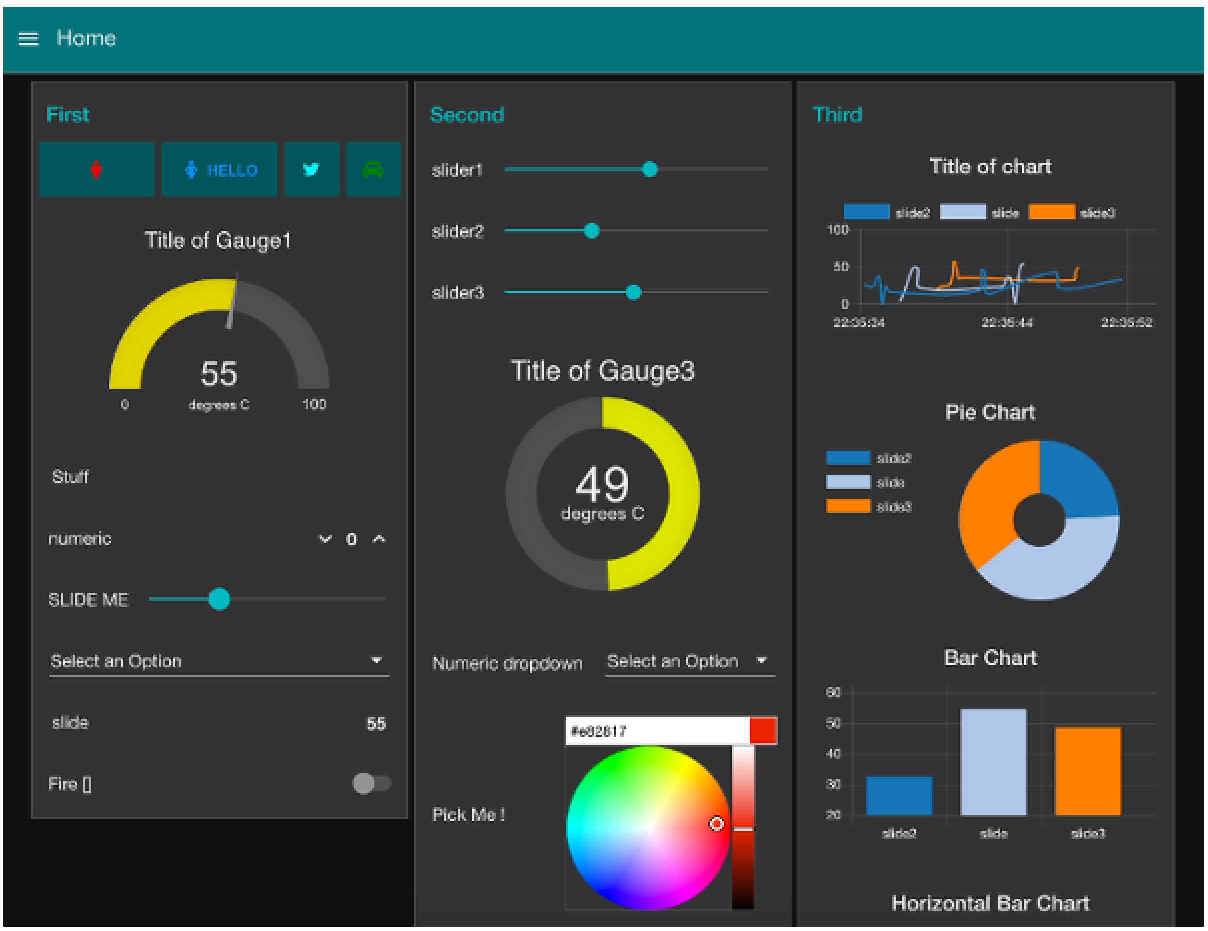


Fig.8.4 Diagram showing Controller Design

8.6 Libraries of Adafruit

Before we get started using Adafruit IO with your Arduino, we'll need to select a library. We provide and support both the Adafruit IO and Adafruit MQTT libraries listed below, but we try starting with the Adafruit IO Arduino Library below:
The library supports the following network platforms and hardware:

8.6.1 Adafruit Air Lift

Give plain microcontroller project a *lift* with the Adafruit AirLift - a breakout board that lets you use the powerful ESP32 as a WiFi co-processor. You probably have your favorite microcontroller (like the ATmega328 or ATSAMD51), awesome peripherals and lots of libraries. But it doesn't have WiFi built in! So lets give that chip a best friend, the ESP32. This chip can handle all the heavy lifting of connecting to a WiFi network and transferring data from a site, even if its using the latest TLS/SSL encryption (it has root certificates pre-burned in).

Having WiFi managed by a separate chip means your code is simpler, you don't have to cache socket data, or compile in & debug an SSL library. Send basic but powerful socket-based commands over 8MHz SPI for high speed data transfer. You can use 3V or 5V Arduino, any chip from the ATmega328 or up, although the '328 will not be able to do very complex tasks or buffer a lot of data. It also works great with Circuit Python, a SAMD51/Cortex M4 minimum required since we need a bunch of RAM. All you need is an SPI bus and 2 control pins plus a power supply that can provide up to 250mA during WiFi usage.

8.6.2 Adafruit ESP8266

Feather is the new development board from Adafruit, and like it's namesake it is thin, light, and lets you fly! We designed Feather to be a new standard for portable microcontroller cores.

This is the Adafruit Feather HUZZAH ESP8266 - our take on an 'all-in-one' ESP8226 WiFi development board with built in USB and battery charging. Its an ESP8266 WiFi module with all the extras you need, ready to rock. At the Feather HUZZAH's heart is an ESP8266 WiFi microcontroller clocked at 80 MHz and at 3.3V logic. This microcontroller contains a Tensilica chip core as well as a full WiFi stack. You can progam the microcontroller using the Arduino IDE for an easy-to-run Internet of Things core. We wired up a USB-Serial chip that can upload code at a blistering 921600 baud for fast development time. It also has auto-reset so no noodling with pins and reset button pressings.

8.6.3 Adafruit ESP32

The ESP8266 started a small revolution by bringing WiFi to a small and cheap package that also had enough processing power and enough pins to get small things done. Now get ready to take your bite-sized WiFi capabilities to the next level with the ESP32 Development Board!

The development board breaks out all the module's pins to 0.1″ headers and provides a CP2102 USB-TTL serial adapter, programming and reset buttons, and a power regulator to supply the ESP32 with a stable 3.3 V. Espress if doubled-down on the CPU resources for the ESP32 with a dual core, running at 160MHz and tons more pins and peripherals.

The ESP32 is still targeted to developers. Not all of the peripherals are fully documented with example code, and there are some bugs still being found and fixed. We got many sensors and displays working under Arduino IDE, so you can expect things like I2C and

SPI and analog reads to work. But other elements are still under development. This board is designed for use with the Tensilica tool chain (ESP IDF) so we recommend that for better hardware-support coverage. Color of PCB may vary.

8.6.4 Adafruit ATWINC1500

The Adafruit ATWINC1500 WiFi Breakout uses SPI to communicate, so with about 6 wires, you can get your wired up and ready to go. Right now the Atmel-supplied library works great with Arduino M0's and M4s but does not work on Uno (328P) and may not work on other Arduinos. You can clock it as fast as 12MHz for speedy, reliable packet streaming. And scanning/connecting to networks is very fast, a few seconds.

This module works with 802.11b, g, or n networks & supports WEP, WPA and WPA2 encryption. You can use it in Soft AP mode to create an ad-hoc network. For secure client connections, there is TLS 1.2 support in firmware 19.4.4.

8.6.5 Adafruit WICED

The WICED Feather is based on Broadcom's WICED (Wireless Internet Connectivity for Embedded Devices) platform, and is paired up with a powerful STM32F205 ARM Cortex M3 processor running at 120MHz, with support for TLS 1.2 to access sites and web services safely and securely.

We spent a lot of time adding support for this processor and WiFi chipset to the Arduino IDE you know and love. Programming doesn't rely on any online or third party tools to build, flash or run your code.

5.6.6 Adafruit Ethernet

Wireless is wonderful, but sometimes you want the strong reliability of a wire. If your Feather board is going to be part of a permanent installation, this Ethernet

FeatherWing will let you add quick and easy wired Internet. Just plug in a standard ethernet cable, and run the Ethernet2 library for cross-platform networking. Works with all/any of our Feather boards as an Ethernet Client (server not supported at this time)!

Ethernet is a tried-and-true networking standard. It's supported by every hub and switch, and because there's a physical connection you don't have to noodle around with SSIDs, passwords, authentication schemes or antennas.

8.6.7 Adafruit FONA (Cellular)

Introducing Adafruit FONA MiniGSM, an adorable all-in-one cellular phone module that lets you add voice, text, SMS and data to your project in an adorable little package.

This module measures only 1.75"x1.25" but packs a surprising amount of technology into it's little frame. At the heart is a GSM cellular module (we use the latest SIM800) the size of a postage stamp. This module can do just about everything

- Quad-band 850/900/1800/1900MHz - connect onto any global GSM network with any 2G SIM (in the USA, T-Mobile is suggested)
- Make and receive voice calls using a headset OR an external 8Ω speaker + electret microphone
- Send and receive SMS messages
- Send and receive GPRS data (TCP/IP, HTTP, etc.)
- Scan and receive FM radio broadcasts (yeah, we don't exactly know why this was included but it works really well)
- PWM/Buzzer vibrational motor control
- AT command interface with "auto baud" detection

Chapter 9
Circuit

9.1 Basics of circuit designing in IOT

IoT networks have become exceedingly popular these days. IoT (Internet Of Things) in the simplest words means a network of devices connected to each other to send and receive data. Of course some of these networks may be possible with pre-manufactured devices that do not require any manual circuit building. However, for some applications, you might want to fine tune the devices to fit your requirement or build a small device yourself with components and a breadboard. In such cases read this article to understand some rules and factors that you need to consider before you design a circuit for IoT application.

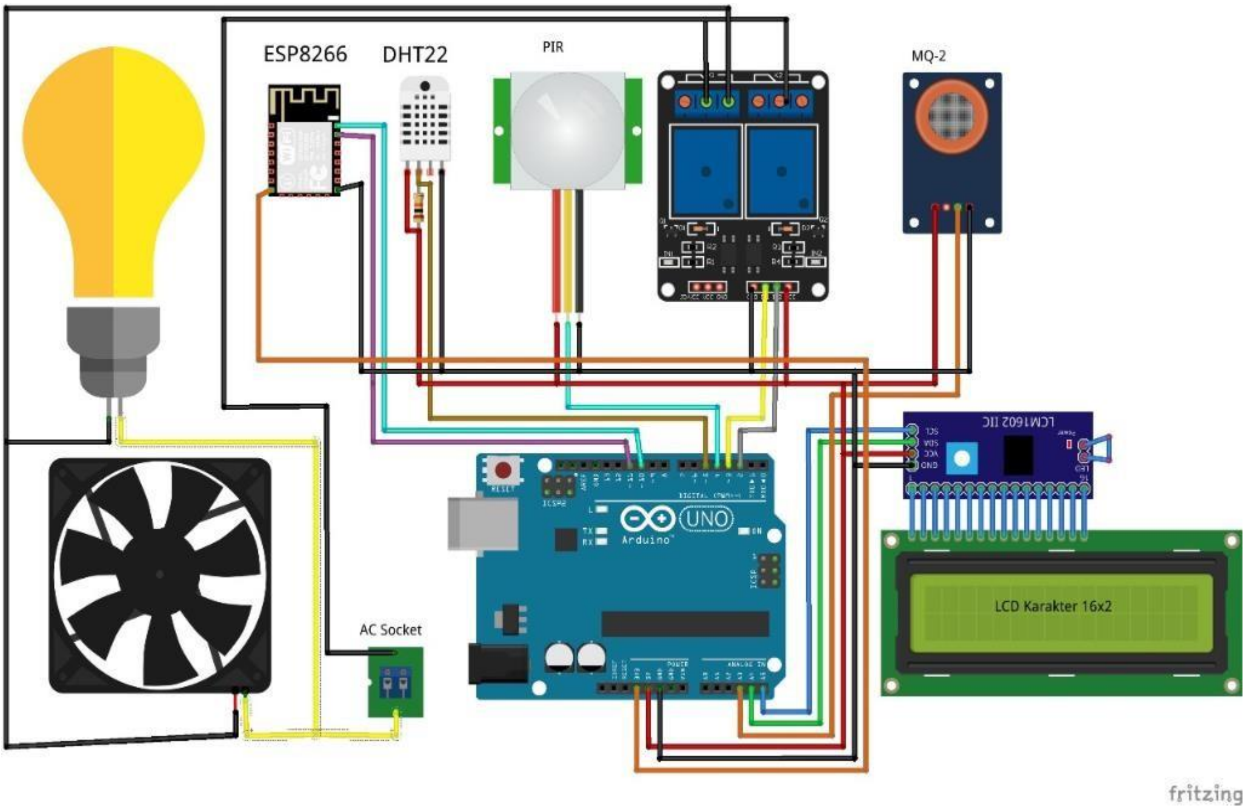


Fig.9.1: Example of Home Automation Circuit Design

9.2 Circuit designing rules

In any circuit design, the basic fundamentals are very important. This knowledge will ensure that you understand the relation between potential and current and are able to pick the correct value of resistance required.

- **Ohms Law:**

As a mathematical expression Ohm's law states that V = I . R In circuit schematic form

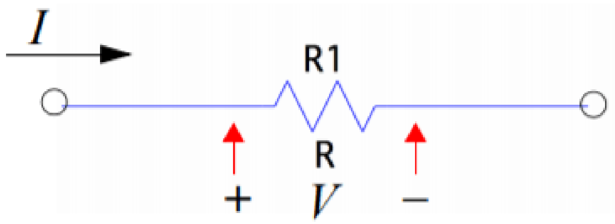


Fig.9.2: Ohms Law

Ohm's law can be written in three equivalent forms,

V = I . R : The voltage across a resistor is the product of the current passing through the resistor times the resistance

- **Kirchhoff's Laws Voltage law**

Sum of voltages around a loop is zero.

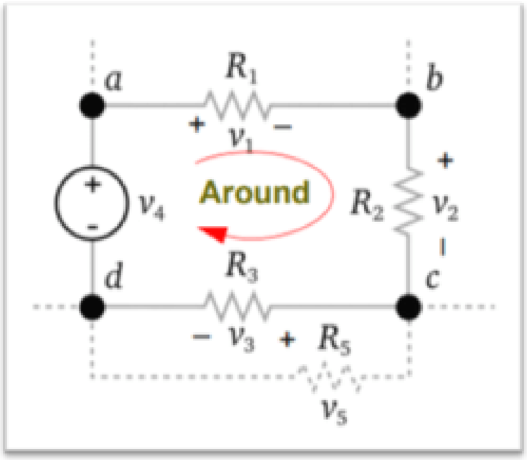


Fig.9.3: Kirchhoff's Voltage Law

Here, v1 + v2 + v3 – v4 = 0

Current law

The current entering a junction is equal to the current leaving a junction.

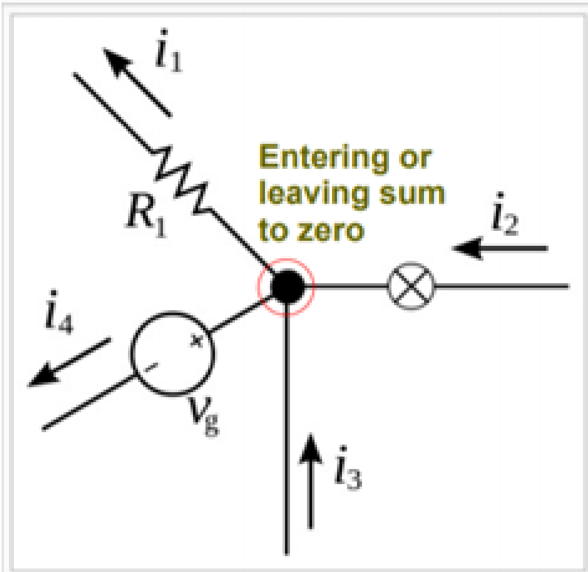


Fig.9.4: Kirchhoff's Current Law

Here, i2 + i3 = i1 + i4

- **Resistance calculation**

We can, in simple terms, say that as voltage drives the current, resistance impedes it. The unit for resistance is ohm, where one ohm is the resistance the allows one amp of current to flow and produce a voltage (potential) drop of one volt.

i) Series

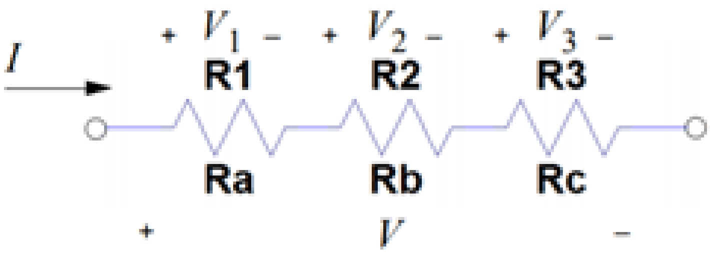


Fig.9.5: Series Resistance

V = V1 +V2+V3 = IR1+IR2+IR3

So, to calculate the total resistance in series,

$$R_{eq} = \frac{V}{I} = \frac{I(R_1 + R_2 + R_3)}{I} = R_1 + R_2 + R_3$$

ii) Parallel

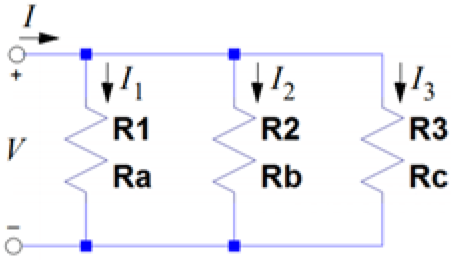


Fig.9.6: Parallel Resistance

I = I1 + I2 + I3

So, to calculate the total resistance in parallel,

$$R_{eq} = \frac{V}{I} = \frac{V}{V\left[\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}\right]} = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}}$$

9.3 Factors affecting IOT Circuit design

Designing a circuit is the Electronics equivalent to creating and bringing a concept to life. To do so, there are some factors that you have to bear in mind. Some of these are generic and others more so for IoT purposes.

9.3.1 Calculating component requirements

Theoretical calculations are needed to select some of the. However in doing so, you must ensure to not only the current or voltage flowing through the component, but also the tolerance and other physical parameters of the component. Tolerance of components, make, quality, and temperature considerably affect the working of the circuit.

9.3.2 Specification of circuit requirements

The first factor that you need to look into is specification of the circuit components. You must use electronic laws and consider your purpose and budget. Specifications include size, shape, weight, current rating and even brands of the components and sensors.

9.3.3 Analog vs Digital

Another area of consideration in the design is analog vs. digital circuits. While designing the circuit, the amount of analog or digital that must be incorporated is not always easy to decide owning to the trade-offs that come with it. These aim at reducing cost, unnecessary functions, development work and circuitry.

9.3.4 Selecting components:

This is a fairly straightforward rule, but let me elaborate why using a capacitor as an illustration. When manufacturing, some capacitors form a film inside. The tolerance for capacitors ranges from -50 per cent to 100 per cent. This means that most capacitors available in the market have a low shelf-life. When a capacitor that has not been used for a while is suddenly a part of an IoT circuit, it may eventually fail. These will then reflect as a ‘failed system’ rather than an unfit component.

9.3.5 Consistency

This is an important long-term factor to consider. IoT applications require circuits to be suitable for usage over a long period of time. The data accuracy in IoT devices is also of importance. Thus one must take into careful consideration the durability of their circuit.

9.3.6 Verify that your circuit works in various conditions

This is especially due to the fact that IoT applications may require the system to be placed remotely. And if your project has the potential to turn into an industrial application, ensure that you keep this in mind

9.3.7 Simplicity

When designing a circuit, try to make it as simple and straightforward as possible. Unnecessarily complex circuits do not show the depth of knowledge and rather portrays the lack of it. When attempting to explain your circuit through a manual, it should be simple and precise. This means that the reader must be able to understand the circuit with ease regardless of his expertise.

9.3.8 Layout of the circuit

This involves determining the physical position and the electrical connection of each component. This might seem difficult and time consuming, but take careful consideration while doing this as it can help your final product turn out as you envision it.

These factors may seem trivial, but may have a profound impact on the overall working of the circuit. Internet of Things (IoT) circuit design has some extra constraints to the circuit design process that we must also keep in mind while designing IoT circuits.

9.4 Challenges in IOT Circuit designing

9.4.1 Space Constraint

In the case of IoT, one of the primary selling points that they offer a plethora of capabilities, but don’t take up much space. In fact any IoT devices are wearable devices. This lack of space makes designing a circuit an IoT application more time-consuming than expected.

9.4.2 Low Power Usage

IoT circuits are expected, due to their small size and varied use, to require only small amounts of power. If your design has high power consumption, it may not be very useful.

9.4.3 New materials to experiment with

IoT circuits are constantly experimenting which materials and components which can lengthen the lifetime and be more consistent in the long run. Circuit designers are expected to spend time researching and finding the most optimum available products.

9.4.4 Designing Products, not just the circuit

When you start to design an IoT circuit, you have to keep in mind the big picture. The circuit is but a small part of a large network that will continue to only grow as time progresses. Your design has to be compatible and be the foundation for a larger purpose and application.

9.5 Introduction to IOT connectivity

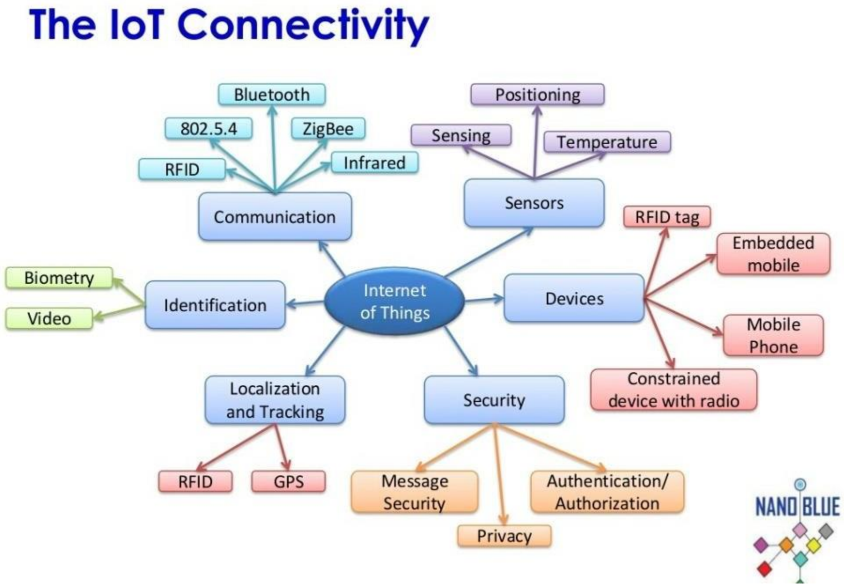


Fig.9.7: IOT Connectivity

When it comes to connecting the Internet of Things, there are a seemingly overwhelming number of options. Cellular, satellite, WiFi, Bluetooth, RFID, NFC, LPWAN, and Ethernet are just some of the possible ways to connect a sensor/device. And within each of these options there can be different providers (e.g. for cellular there's T-Mobile, Verizon, AT&T, Sprint, etc.).

9.6 Trade Off Between Power Consumption, Range, and Bandwidth

The perfect connectivity option would consume extremely little power, have huge range, and would be able to transmit large amounts of data (high bandwidth). Unfortunately, this perfect connectivity doesn't exist.

Each connectivity option represents a tradeoff between power consumption, range, and bandwidth. This allows us to segment the various connectivity options into three major groups, which you'll find below. However, these groups should serve more as a framework for thinking about connectivity than a definitive classification, as there can be connectivity standards that sit more on the borders of these groups.

9.6.1 High Power Consumption, High Range, High Bandwidth

To wirelessly send a lot of data over a great distance, it takes a lot of power. A great example of this is your Smartphone. Your phone can receive and transmit large amounts of data (e.g. video) over great distances, but you need to charge it every 1–2 days. Connectivity options in this group include cellular and satellite.

9.6.2 Low Power Consumption, Low Range, High Bandwidth

To decrease power consumption and still send a lot of data, you have to decrease the range. Connectivity options in this group include Wi-Fi, Bluetooth, and Ethernet.

Ethernet is a hard-wired connection, so the range is short because it's only as far as the wire length. WiFi and Bluetooth are both wireless connections with high bandwidth and lower power consumption than cellular and satellite

9.6.3 Low Power Consumption, High Range, Low Bandwidth

To increase range while maintaining low power consumption, you have to decrease the amount of data that you're sending. Connectivity options in this group are called Low-Power Wide-Area Networks (LPWANs).

LPWANs send small amounts of data which allows them to operate at very low power with ranges in miles rather than feet. For example, a moisture sensor for agricultural purposes doesn't need to send a lot of data, perhaps just a single number (the moisture level) every few hours. You also don't want this sensor to consume a lot of power because it needs to run on battery (plugging it into an outlet in the middle of a field just isn't realistic). And since agriculture covers a wide area, Wi-Fi and Bluetooth lack the range.

Chapter 10
Home Automation

10.1 Introduction

A home automation system is a technological solution that enables automating the bulk of electronic, electrical and technology-based tasks within a home.

It uses a combination of hardware and software technologies that enable control and management over appliances and devices within a home.

Home automation is also known as domotics, and a home with an automation system is also known as a smart home.

Home automation or domotics is building automation for a home, called a smart home or smart house. A home automation system will monitor and/or control home attributes such as lighting, climate, entertainment systems, and appliances. It may also include home security such as access control and alarm systems. When connected with the Internet, home devices are an important constituent of the Internet of Things ("IoT").

A home automation system typically connects controlled devices to a central hub or "gateway". The user interface for control of the system uses either wall-mounted terminals, tablet or desktop computers, a mobile phone application, or a Web interface that may also be accessible off-site through the Internet.

While there are many competing vendors, there are increasing efforts towards open source systems. However, there are issues with the current state of home automation including a lack of standardized security measures and deprecation of older devices without backwards compatibility.

Home automation has high potential for sharing data between family members or trusted individuals for personal security and could lead to energy saving measures with a positive environmental impact in the future.

The home automation market was worth US$5.77 billion in 2013, predicted to reach a market value of US$12.81 billion by the year 2020.

Fig.10.1: Smart Home Automation System

10.2 History

Early home automation began with labour-saving machines. Self-contained electric or gas powered home appliances became viable in the 1900s with the introduction of electric power distribution and led to the introduction of washing machines (1904), water heaters (1889), refrigerators (1913), sewing machines, dishwashers, and clothes dryers.

In 1975, the first general purpose home automation network technology, X10, was developed. It is a communication protocol for electronic devices. It primarily uses electric power transmission wiring for signalling and control, where the signals involve brief radio frequency bursts of digital data, and remains the most widely available. By 1978, X10 products included a 16 channel command console, a lamp module, and an appliance module. Soon after came the wall switch module and the first X10 timer.

By 2012, in the United States, according to ABI Research, 1.5 million home automation systems were installed. Per research firm Statists more than 45 million smart home devices will be installed in U.S. homes by the end of the year 2018.

The word "domotics" (and "domotica" when used as a verb) is a contraction of the Latin word for a home (domus) and the word robotics. The word "smart" in "smart home" refers to the system being aware of the state of its devices, which is done through the information and communication technologies (ICT) protocol and the Internet of Things (IoT).

Throughout the 2000s, smart devices and systems have been evolving at a rapid pace. It’s estimated that by 2012, there were already 1.5 million automated home systems in place. In 2014, Amazon introduced the Amazon Echo (for Prime members), and while it was originally marketed as a voice-controlled music solution, the inclusion of Alexa quickly demonstrated the use of the device as a smart home hub.

10.3 Applications and Technologies

Home automation is prevalent in a variety of different realms, including:

- Heating, ventilation and air conditioning (HVAC): it is possible to have remote control of all home energy monitors over the internet incorporating a simple and friendly user interface.
- Lighting control system: a "smart" network that incorporates communication between various lighting system inputs and outputs, using one or more central computing devices.
- Occupancy-aware control system: it is possible to sense the occupancy of the home using smart meters and environmental sensors like CO_2 sensors, which can be integrated into the building automation system to trigger automatic responses for energy efficiency and building comfort applications.
- Appliance control and integration with the smart grid and a smart meter, taking advantage, for instance, of high solar panel output in the middle of the day to run washing machines.
- Home robots and security: a household security system integrated with a home automation system can provide additional services such as remote surveillance of security cameras over the Internet, or access control and central locking of all perimeter doors and windows.
- Leak detection, smoke and CO detectors
- Indoor positioning systems (IPS).
- Home automation for the elderly and disabled.
- Pet and baby care, for example tracking the pets and babies' movements and controlling pet access rights.

- Air quality control (inside and outside). For example, Air Quality Egg is used by people at home to monitor the air quality and pollution level in the city and create a map of the pollution.
- Smart kitchen, with refrigerator inventory, premade cooking programs, cooking surveillance, etc.
- Voice control devices like Amazon Alexa or Google Home used to control home appliances or systems.

10.4 Implementations

In a review of home automation devices, Consumer Reports found two main concerns for consumers:

- A Wi-Fi network connected to the internet can be vulnerable to hacking.
- Technology is still in its infancy, and consumers could invest in a system that becomes abandonware. In 2014, Google bought the company selling the Revolv Hub home automation system, integrated it with Nest and in 2016 shut down the servers Revolv Hub depended on, rendering the hardware useless.

In 2011, Microsoft Research found that home automation could involve high cost of ownership, inflexibility of interconnected devices, and poor manageability. When designing and creating a home automation system, engineers take into account several factors including scalability, how well the devices can be monitored and controlled, ease of installation and use for the consumer, affordability, speed, security, and ability to diagnose issues. From iControl showed that consumers prioritize ease-of-use over technical innovation, and although consumers recognize that new connected devices have an unparalleled cool factor, they are not quite ready to use them in their own homes yet.

Historically, systems have been sold as complete systems where the consumer relies on one vendor for the entire system including the hardware, the communications protocol, the central hub, and the user interface. However, there are now open hardware and open source software systems which can be used instead of or with proprietary hardware. Many of these systems interface with consumer electronics such as the Arduino or Raspberry Pi, which are easily accessible online and in most electronics stores. In addition, home automation devices are increasingly interfaced with mobile phones through Bluetooth, allowing for increased affordability and customizability for the user.

IoT Software Implementation Challenges:

- Compatibility of Various IoT Systems.
- Authentication and Identification Issues in IoT.

- Integration of IoT Points with IoT Software.
- IoT Data Storage Challenge.
- Connectivity and Power Management IoT Challenges.
- Unstructured Data Processing.
- Incorrect Data Capture Difficulties.

10.5 Criticism and Controversy

Home automation suffers from platform fragmentation and lack of technical standards a situation where the variety of home automation devices, in terms of both hardware variations and differences in the software running on them, makes the task of developing applications that work consistently between different inconsistent technology ecosystems hard. Customers may hesitate to bet their IoT future on proprietary software or hardware devices that use proprietary protocols that may fade or become difficult to customize and interconnect.

The nature of home automation devices can also be a problem for security, data security and data privacy, since patches to bugs found in the core operating system often do not reach users of older and lower-price devices. One set of researchers say that the failure of vendors to support older devices with patches and updates leaves more than 87% of active devices vulnerable.

Concerns have been raised by tenants renting from landlords who decide to upgrade units with smart home technology. These concerns include weak wireless connections that render the door or appliance unusable or impractical; the security of door passcodes kept by the landlord; and the potential invasion of privacy that comes with connecting smart home technologies to home networks.

Researchers have also conducted user studies to determine what the barriers are for consumers when integrating home automation devices or systems into their daily lifestyle. One of the main takeaways was regarding ease of use, as consumers tend to steer towards "plug and play" solutions over more complicated setups. One study found that there were large gaps in the mental-models generated by users regarding how the devices actually work. Specifically, the findings showed that there was a lot of misunderstanding related to where the data collected by smart devices was stored and how it was used. For example, in a smart light setup, one participant thought that her iPad communicated directly with the light, telling it to either turn off or on. In reality, the iPad sends a signal to the cloud system that the company uses (in this case, the Hue Bridge) which then signals directly to the device.

Overall, this field is still evolving, and the nature of each device is constantly changing. While technologists work to create more secure, streamlined, and standardized security protocols, consumers also need to learn more about how these devices work and what the implications of putting them in their homes can be. The growth of this field is limited not only by the technology but also by a user's ability to trust a device and integrate it successfully into his/her daily life.

10.6 Impact

Utilizing home automation could lead to more efficient and intelligent energy saving techniques. By integrating information and communication technologies (ICT) with renewable energy systems such as solar power or wind power, homes can autonomously make decisions about whether to store energy or expend it for a given appliance, leading to overall positive environmental impacts and lower electricity bills for the consumers using the system. In order to do this, researchers propose using data from sensors regarding consumer activity within the home to anticipate the consumer needs and balance that with energy consumption.

Furthermore, home automation has large potential regarding family safety and security. According to a 2015 survey done by iControl, the primary drivers of the demand for smart and connected devices are first "personal and family security", and second "excitement about energy savings". Home automation includes a variety of smart security systems and surveillance setups. This allows consumers to monitor their homes while away, and to give trusted family members access to that information in case anything bad happens.

Fig.10.2: Internet Enabled Cat Feeder

10.7 FUTURE SCOPE OF HOME AUTOMATION

Future scope for the home automation systems involves making homes even smarter. Homes can be interfaced with sensors including motion sensors, light sensors and temperature sensors and provide automated toggling of devices based on conditions. More energy can be conserved by ensuring occupation of the house before turning on devices and checking brightness and turning off lights if not necessary. The system can be integrated closely with home security solutions to allow greater control and safety for homeowners. The next step would be to extend this system to automate a large-scale environment, such as offices and factories. Home Automation offers a global standard for interoperable products. Standardization enables smart homes that can control appliances, lighting, environment, energy management and security as well as the expandability to connect with other networks.

Chapter 11

Smart Gardening system

11.1 What is a Smart Garden?

- In today's busy world, we forget to nourish and water plants that makes our home clean and soothing.
- The internet of things (IOT)is based upon software and hardware .
- it describes the real time experience to world which develops small things with limited storage and processing capacity ,and consequential issues regarding reliability ,performance ,security ,privacy .
- on other side embedded system are designed to do some specific task with software programming and real time performance constraints that must be met ,safety and reliability .
- the software written for embedded system is often called firmware and is stored in read only memory.
- Transmission advice has assume form of an importunate article considering monetary compound moreover a leading factfinding case in a period of extreme ten years.
- There are now extended mobile phone subscriptions than wired-line subscriptions.
- In recent times, one area of fiscal interest has been low-cost, low-power, and short-distance wireless communion recycled for \peculiar wireless networks.
- " Technology amelioration are administer lesser and more practical invention for systemize computational processing, wireless communication, and a host of other functionalities.
- These embedded elucidation devices will be combined into applications pasturing from homeland security to industry automation and monitoring.
- Think about it what if your plants are able to tell you when they are happy and when they require proper attention.
- It would be awesome right! It would be really helpful if we get a notification on our phones about our plant's health and needs.

- Taking account of this we came up with the idea of building a smart garden with IoT plant monitoring system.
- It checks the moisture content of the soil and intensity of sunlight falling on plant at regular time intervals.
- After the data is processed and verified by a logic created using Integromat, accordingly an pop- up is sent about the plant's health.
- It will make our life simple and easy and will also help us to take better care of our plants.
- Smart garden monitoring systems leveraging Internet of Things (IoT) can help transform your gardening process to a smart and dynamic one.
- The system will monitor the light intensity and moisture content of the soil at regular intervals and alert you when the plants need watering.
- And, if you have a smart plant watering system, your plants will be watered even in your absence.
- A vegetable or herb garden that is controlled by computer.
- Although farming and gardening have been enhanced with computers and electronic devices for decades, the smart garden often refers to small, indoor units that by various means determine when to alert the user to add nutrients.
- Smart gardens may also manage their own lighting.
- Overall, the Smart Garden is almost completely a success.
- With more time, the project would have been a complete success.
- As this is a prototype and the app was able to correctly display the correct sensor reads graph when plant had been watered. “Plant is in dryness” as the plant was in room loosing moisture consistently due to hot climate.

- when the thermistor was warmed using two fingers; and “Plant is in sunlight” when the plant was in a lit room
- The sensor had four different thresholds: in sunlight, in shade, in direct light, and light is low (when there isn’t any light).
- Our final project accurately showed each correct readout for each of these light thresholds.
- The moisture sensor had two thresholds and also accurately displayed the correct readouts on the mobile application.
- The temperature sensor had three thresholds: above room temperature, room temperature, and below room temperature.
- The basic requirements for the Smart Garden were completed, except for the hardware issue with the daughter board.
- However, for a prototype, the Smart Garden displayed effective functionality as well as provided a design to build off of for future projects.
- The project consists in develop a garden system environment able to auto monitoring the humidity and water amount in the garden and provide the resources the environment needs.
- The system under consideration is a project prototype developed using an Edison board and permaculture principles, which allow for sustainable long-term garden cultivation with maximum harvest based on the natural capacity of the soil enhanced by well-planned companion planting.

- The system vision is to connect a growing bed via sensors to an Edison board such that we can measure moisture and temperature and log that data.

Fig.11.1 Diagram showing Plants in a Garden

- Automation the process of monitoring the garden can transform garden irrigation process from being manual and static to smart and dynamic one.
- υ This leads to higher comfortability, water using efficiency and less human supervision effort.
- This paper proposes a cloud based Internet of Things (IoT) smart garden monitoring and irrigation system using Arduino Uno.
- The watering requirement for a plant can be adjusted by monitoring the soil moisture.
- υ Measuring the soil moisture of the plant gives information if the plant is ideally watered, over watered or under waters.
- Whether you are a lazy or simply uninformed gardener, gardening with technology can help with chores and tasks common in the landscape.

- From smart irrigation systems to self-control lawn mowers, technology has its finger on the gardener's pulse.
- Many of us are familiar with smart plant meters, which monitor the health and moisture levels of houseplants, but the concept doesn't stop there.
- Use our smart gardening guide for tips on technological products that are engineered to create healthier, low maintenance solutions for your yard.

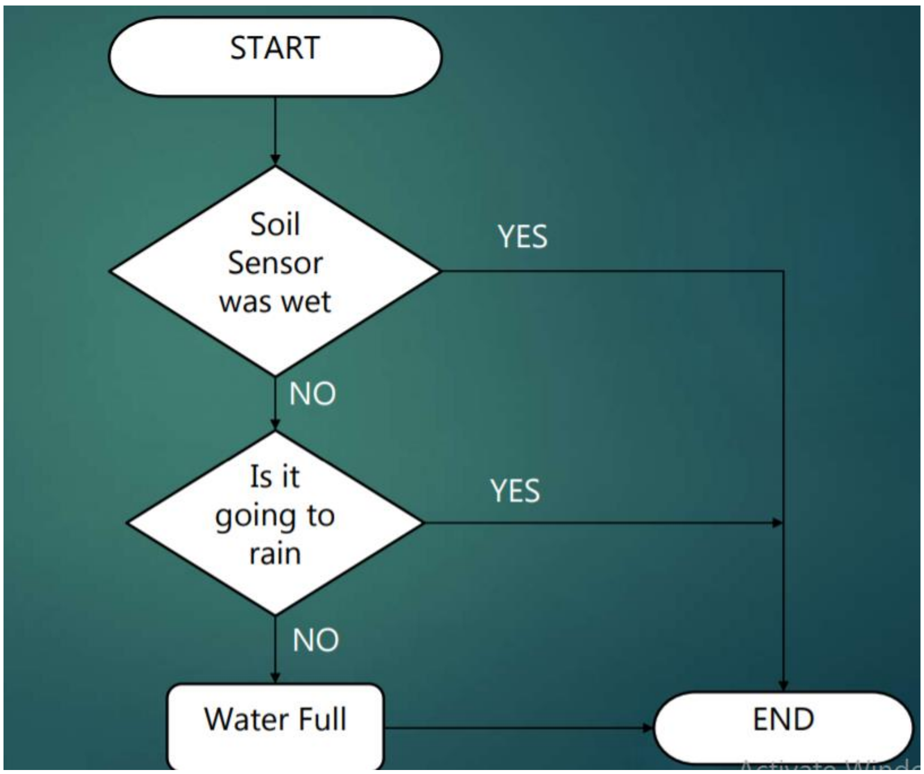


Fig.11.2 Diagram showing Plants in a Garden

11.2 Smart Watering System

Self-watering pots are now making it easy for plant lovers to nurture greens at home. There are various companies which offer self-watering pots in India in different sizes, shapes and colours. Self-watering pots work on a reservoir system using capillary action to moisten the soil. It has a container which one has to fill with the water. There is an overflow hole for the extra water to drain. The soil absorbs water from the bottom, so as long as the reservoir is filled the plants get a consistent level of moisture, delivered to their roots.

Today, one can buy a complete gardening kit, with the growing medium, seeds and self-watering pots. "The Salad Garden kit is one such convenient system, where users can water the plant once a week or in 15 days. The plants automatically take the required water. The science of self-watering depends on the potting mix".

Rapid technological developments have been applied in various sectors of life, one of which is in the agricultural sector. By applying technology in the agricultural sector, it can reduce energy and time wasted due to the application of conventional methods. One of the innovations that are currently trending in technology is the Internet of Things (IoT). Internet of Things (IoT) is a technology that allows physical objects to be connected to the internet in realtime. This research applies the Internet of Things to the garden irrigation system, by remotely controlling water pump and monitoring soil moisture in the garden. Using the application of the Internet of Things the garden owners can measure and detect soil moisture in their plantations. Then efficiently, can manage the use of water used in real-time. The water supply inside the garden is connected to water pump, that will be activated when the soil moisture sensor detects low moisture level, and will be automatically adjust the moisture parameter on its optimum number. In addition, the owner of the garden can monitor the condition of soil moisture through mobile app.

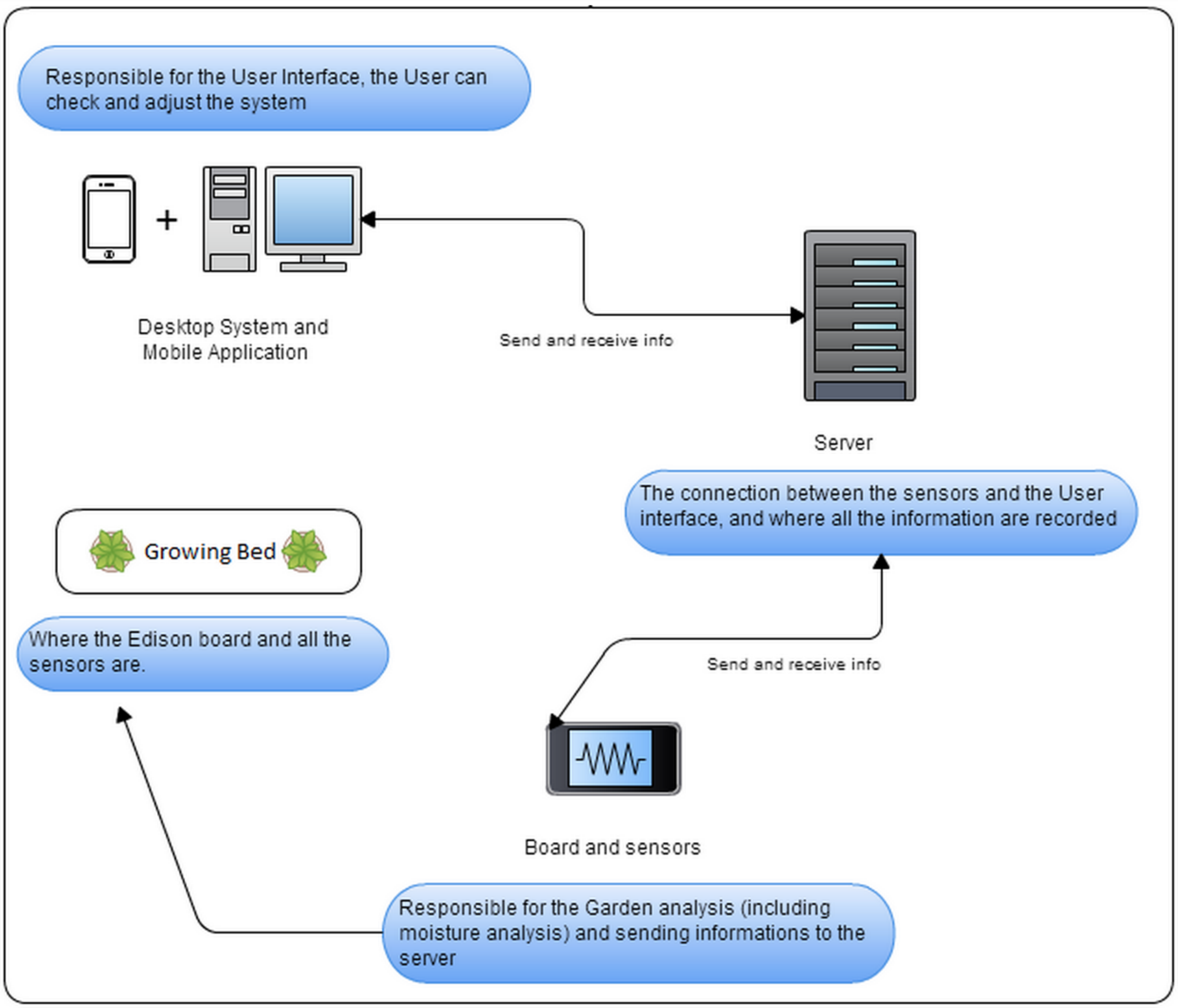


Fig.11.3 Diagram showing Plants in a Garden

11.3 Use of Smart Garden

- **You'll have plants to welcome you home from vacation:**
  With a Smart Garden, we'll never have to come home to an empty house. We'll be excited to see how the capsules you planted have grown while you were away. Your plants will also purify the air!

- **Visual Appeal**:
  To put it simply, a Smart Garden is a visual marvel. It's contemporary, sleek design comes in white, grey, or beige. The Smart Garden's simple yet sophisticated appearance enables it to look fantastic on any kitchen top, table, or windowsill. It stands out among all gifts as the ultimate indoor garden for modern life.

- **Longevity**:
  Smart Gardens are built to last. Your loved one will enjoy being able to grow houseplants well into the new year and beyond. Many Christmas presents these days have a temporary appeal. The gift of a Smart Garden, however, is truly unique and bound to be remembered for years to come.

- **Health Benefits**:
  When giving the gift of a Smart Garden, you are opening the door to health conscious choices. With a Smart Garden, you can grow the healthiest greens indoors. Perhaps your gift will encourage the recipient to enjoy a cleaner, more balanced diet through sustainable methods. Who knows how many will be inspired to follow the same path?

- **Engages People of All Ages**:
  One of the most charming aspects of a Smart Garden is the fact that it's easy to use. In this sense it appeals to all ages and has the potential to bring people together; parents, children, young and

elderly couples, team members within a company, the list is endless. The excitement that comes with growing your own plants is beautifully contagious.

- **What It Says About You**:
  By giving a Smart Garden as a gift, you are showing an appreciation for uniqueness, innovation, sustainable gardening, health and nature. You are also showing your loved one that they deserve the best. After all, we believe growing indoor plants brings an abundance of benefits. Perhaps one of the most underrated benefits is the tranquility that the presence of a Smart Garden brings to any living space.

11.4 Guide to a Smart Garden

Smart Gardening Guide More and more products are being developed to help lower our carbon footprint, simplify chores and help us be wiser consumers. Such technology can enhance plant care, help with landscape design and inform us of the best plants for specific sites. In an imagined future, all the drudgery of gardening will be removed, leaving only the pleasurable aspects of maintaining your home.

- **Smart plant monitors –** There are many plant monitors available to introduce technology to the beginning gardener. Many of these are simply inserted into soil and can take measurements of moisture levels, track light and humidity, and even analyze soil. Many can even determine the nutrients in soil.

- **Smart gardens –** Indoor gardens take the guesswork out of growing your own food or herbs. Most are self-contained systems that provide light, automatic watering, fertilizer and customized heat levels. All you need to do is plant or sow seed and the unit does the rest.

- **Smart sprinklers –** Smart sprinklers do more than just schedule irrigation. They can determine breaks and leaks in the system, save water, adjust to accommodate weather and often can be monitored and changed via your phone or computer.

- **Expandable pots –** A really wonderful new concept is the expandable pot. The containers are said to expand as the plant grows so you don’t need to keep purchasing pots a size up.

- **Gardening apps** – Garden apps can help with design, plant ID, placement of irrigation, solve problem areas and much more. Many, like the GKH Gardening Companion (for Android and iPhone), are available free or you can purchase easy-to-use guides in a variety of formats.

- **Smart mowers –** Mowbot is an automated lawn mower. It operates similarly to robotic vacuums only in a mower. No more sweating in the hot sun trying to get the lawn cut.

- **Robotic weeders –** A product under development is Tertill, a solar powered weeding robot. The idea is that you simply place the product out in a sunny location of the garden and it will weed for you. No more back breaking stooping or use of chemicals.

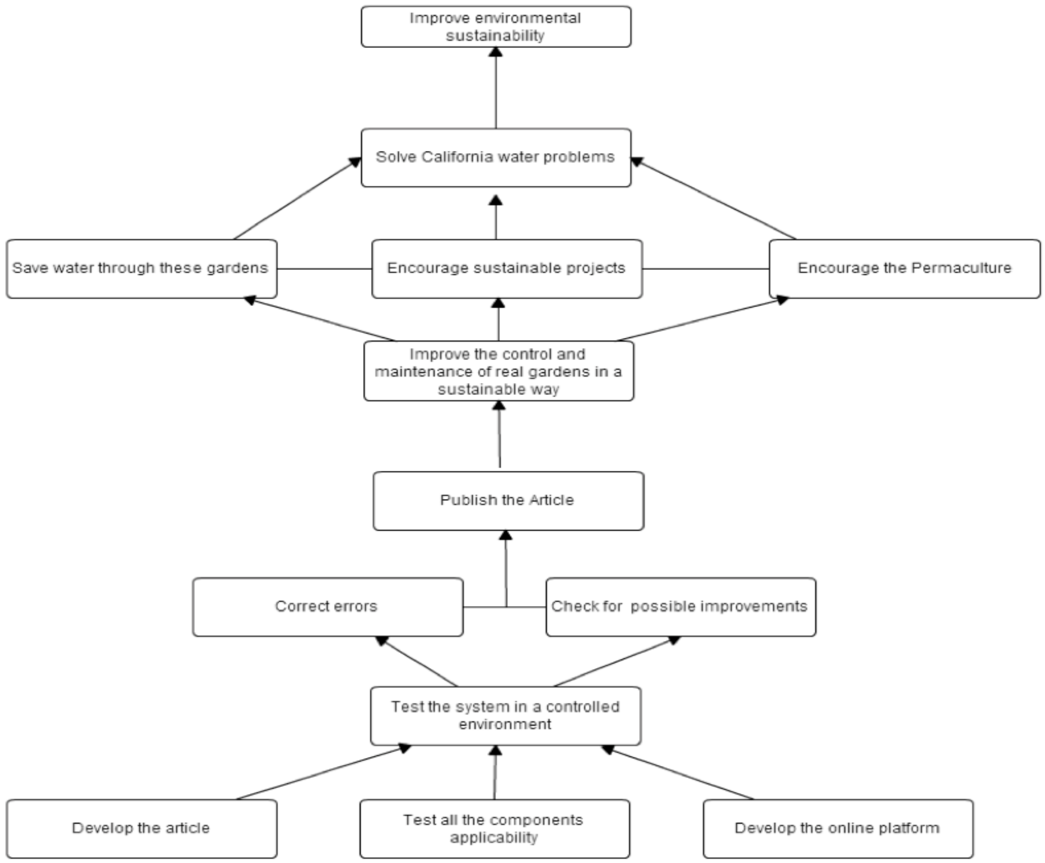


Fig.11.4 Diagram showing Guide to Smart Garden

- After comparative study and considerations of literature survey of other existing systems, there was a proposal of a system The venture "SMART AND GREEN GARDEN SYSTEM" has been effectively composed and tried.
- Coordinating highlights of all the equipment parts utilized have created it.
- Nearness of each module has been contemplated out and set precisely subsequently adding to the best working of the unit.
- Furthermore, utilizing exceptionally propelled IC's and with the assistance of developing innovation the task has been effectively executed.

Chapter 12

Home Automation, Smart and Green Gardening System

12.1 Introduction

- The following is an IOT based project comprising of certain systems embedded together.
- It features automation, real time monitoring, energy management, and many more concepts making it a practical application.
- For a home automation system to function there needs to be a network for communication among devices and a centralized management application, or controller.
- Network controllers have certain requirements: that is Always be on, Stay at the home,

  And be connected to the Internet or an alternative communications pathway.
- The controller needs to be in a device that is always on so the controller is always available to perform its home management tasks.
- There are various devices in the home that could be left on 24/7, such as a desktop computer and routers.
- However, desktop computers use too much energy to be left on 24/7 compared to a router, which is very rarely turned off intentionally.
- Another option might be a laptop computer but laptops on the move are not always connected to the Internet, as for example, during air travel.
- And, for those left at home, having the controller somewhere else may prove to be problematic.
- Automation performs an increasingly vital role in daily experience and global economy.
- Engineers strive to combine automated devices with mathematical and organizational tools to create complex system for a rapidly expanding range of applications and human activities.
- The concept of home automation has been around since the late 1970’s.

- But with the enhancement of technology and smart services, people's expectations have changed a lot during the course of time to perfectly turn the server.
- A home automation system means to grant the end users to manage and handle the electric appliances.
- If we look at different home automation systems over time, they have always tried to provide efficient, convenient, and safe ways for home inhabitants to access their homes.
- Regardless of the change in user's hope, growing technology, or change of time, the appearance of a home automation system has remained the same.
- Plant provides us with almost all the basic needs for survival but we are unable to provide plant with its basic needs like water, non-polluted oxygen and as a result plants are unable to survive.
- In this report on IOT based smart garden monitoring system which sense the requirement of the plant and provide it with water as the soil loses its moisture.
- Different soils have different fertility and moisture level so we have soil and moisture sensor used in this to detect this problem.
- In our country there are six different seasons and each day have different temperature and humidity level so to check the temperature and humidity for the better health and survival of plant temperature and humidity sensor are used which regularly sends data to the server.
- In this way it manages to perform its operations automatically.

12.2 Objective

This project aims at developing a cost effective smart home and gardening system which would cater to the needs of people of the rural as well as the urban society. The system uses non- polluting, digital sensors, natural resources and it is beneficial to the society. Home automation, home control, smart or digital home are just different names for comfort, convenience, security water saving and power saving. These systems are of increasing importance these days. Even though such systems are very expensive in general, they can also be very economical if one designs and construct them for very specific needs. There are many ways to control a smart home system, including wireless communication over internet. This project will help in creating a new generation of smart homes in which user can control the home appliances such as bulbs and also completes the home by providing a miniature smart garden to it. Whenever they want and from wherever they are with the aid of internet. This can be extended to mobiles also in the open source platforms like Android.

The main objective of this project is to develop a home automation system using an Arduino and ESP8266 Chip with Internet being remotely controlled by any Android OS smart phone. As technology is advancing so houses are also getting smarter. Modern houses are gradually shifting from conventional switches to centralized control system, involving remote controlled switches. Presently, conventional wall switches located in different parts of the house makes it difficult for the user to go near them to operate. Even more it becomes more difficult for the elderly or physically handicapped people to do so. Remote controlled home automation system provides a most modern solution with smart phones. In this project, the sensors measure the moisture in the soil and automatically controls the water flow to the garden. People often grow unsuitable plants in their gardens so that they usually die from the lack of enough water, or even too much sunlight. In this project we use the sensors to obtain the values of the parameters and process them using microcontroller and control the survival of plants in the garden. Based on the soil type and plant type, the water requirement for the plant is defined. The user is also alerted periodically about the plants' conditions.

This project revolves around creating a home automation system prototype with the main focus of controlling the home appliances through the internet. The system consists of a central device, a source code and a web application to control the device. Embedded to this home automation is a plant monitoring and smart gardening system. The main purpose of automation is to provide comfort to the people by reducing the manual work and to improve the overall performance of the system without user interaction. The important parameters for the quality and productivity of plant growth are soil and air temperature, humidity, sunlight, soil moisture and pH. Information to the use about the plant health and growth may be provided to the user by continuously monitoring and recording these garden parameters. It provides a better understanding of how each parameter affects the growth of plants.

12.3 Setting up Smart Home

Home automation refers to the ability of your home to make its own decisions depending on environment conditions and give you the option to control it from a remote location. The heart project is the WiFi enabled board that needs no introduction; the ESP8266
based NodeMCU development board. It is an open source platform for developing WiFi based embedded systems and it is based on the popular ESP8266 WiFi Module.

NodeMCU was born out of the desire to overcome the limitations associated with the first versions of the ESP8266 module which was not compatible with breadboards, it was difficult to power and even more difficult to program. The NodeMCU board is easy to use, low cost and that quickly endeared it to the heart of makers and it is one of the most popular boards today.

we will add a 2-channel relay module to the ESP8266 board. The project flow involves the control of NodeMCU’s GPIOs from a webpage on any device connected on the same network as the board. The status of the GPIOs control the coils of the relays and that causes the relay to alternate between normally open (NO) and normally closed (NC) condition depending on the state of the GPIO, thus, effectively turning the connected appliance “ON” or “OFF”.

12.4 Components

12.4.1 ESP8266

- ESP8266 is a low-cost WiFi module that belongs to ESP's family which you can use it to control your electronics projects anywhere in the world. It has an in-built microcontroller and a 1MB flash allowing it to connect to a WiFi.

- The maximum working voltage of the module is 3.3v.

- Power to the ESP8266 NodeMCU is supplied via the on-board MicroB USB connector. Alternatively, if you have a regulated 5V voltage source, the VIN pin can be used to directly supply the ESP8266 and its peripherals.

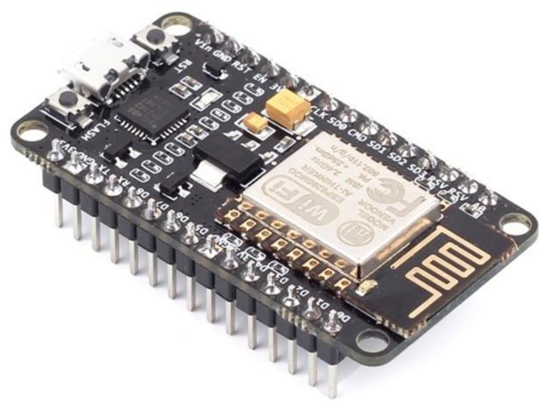


Fig.12.1 Diagram showing ESP8266

12.4.2 Moisture Sensor

- The Soil Moisture Sensor measures soil moisture grace to the changes in electrical conductivity of the earth (soil resistance increases with drought).

- The electrical resistance is measured between the two electrodes of the sensor.

- The soil moisture sensor consists of two probes that are used to measure the volumetric content of water. The two probes allow the current to pass through the soil, which gives the resistance value to measure the moisture value.

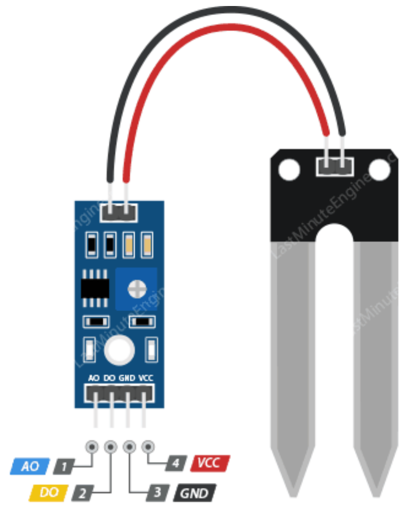


Fig.12.2 Diagram showing Moisture Sensor

12.4.3 Soil Moisture Comparator

It measures the volumetric content of water inside the soil and gives us the moisture level as output. The comparator has both digital and analog outputs and a potentiometer to adjust the threshold level.

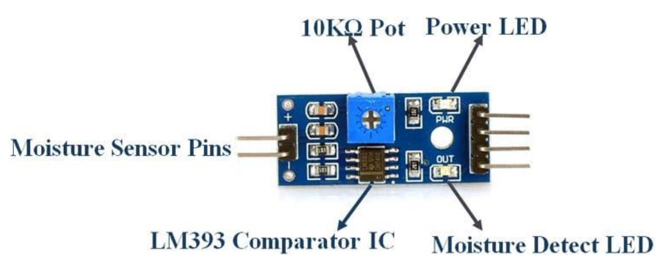


Fig.12.3 Diagram showing Soil Moisture Comparator

12.4.4 Buck Converter

Buck converter is a step-down power supply module. It is used to bring down the power supply to the desired voltage in order to safe guard the circuit.

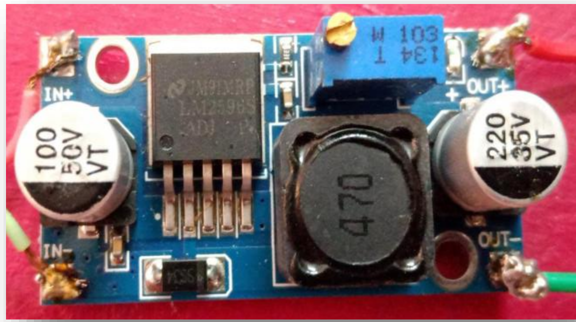


Fig.12.4 Diagram showing Buck Converter

12.4.5 Relay Circuit

Relays are switches that open and close circuits electromechanically or electronically. Relays control one electrical circuit by opening and closing contacts in another circuit. When a relay contact is Normally Closed (NC), there is a closed contact when the relay is not energized.

Fig.12.5 Diagram showing Relay Circuit

12.4.6 Pump & Bulb

Pump: it is used for water supply to the smart garden.

Bulb: it is home appliance that is been controlled through the relay switches.

12.5 ESP8266 Pin Diagram:

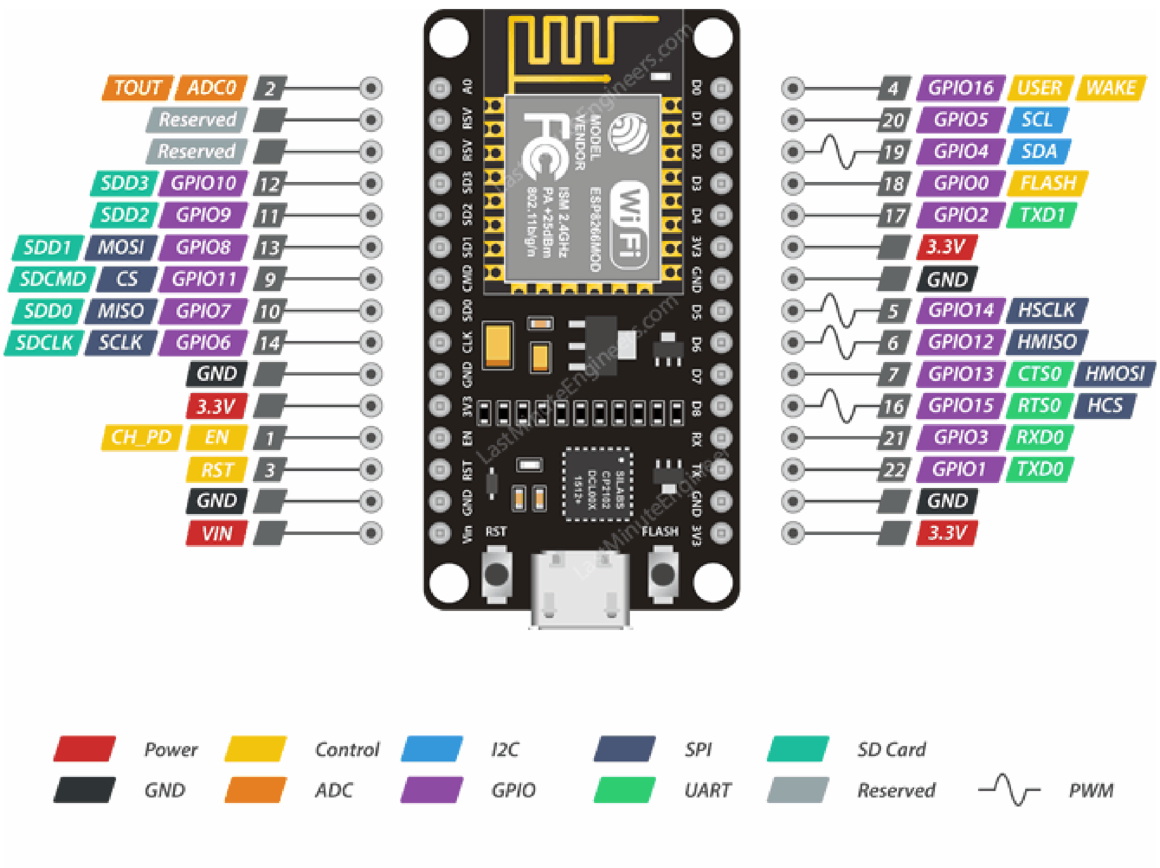


Fig.12.6 Diagram showing ESP8266 Pin Diagram

12.6 Applications of Home Automation

- **Lighting Control:** Leaving the Dark Ages and Stepping Into the Light. It turns out, quite; its capabilities are extensive. You're able to schedule the lights should turn on and off, decide, select the level of light which should be emitted, and choose how particular lights react through controller.
- **HVAC Regulation:** No Longer Burned by Your Heating Bill. As fuel costs rise and the availability and sustainability of our resources becomes a greater concern, heating/cooling our homes efficiently is less a budgetary bonus and more of a necessity.
- **Classic smart home overview:** Smart home is the residential extension of building automation and involves the control and automation of all its embedded technology. It defines a residence that has appliances, lighting, heating, air conditioning, TVs, computers, entertainment systems, big home appliances such as washers/dryers and refrigerators/freezers, security and camera systems capable of communicating with each other and being controlled remotely by a time schedule, phone, mobile or internet.
- **Measuring home conditions:** A typical smart home is equipped with a set of sensors for measuring home conditions, such as: humidity, light and proximity. Each sensor is dedicated to capture one or more measurement. Temperature and humidity may be measured by one sensor, other sensors calculate the light ratio for a given area and the distance from it to each object exposed to it.
- **Managing home appliances:** Creates the cloud service for managing home appliances which will be hosted on a cloud infrastructure. The managing service allows the user, controlling the outputs of smart actuators associated with home appliances, such as such as lamps and fans.
- **Controlling home access:** Home access technologies are commonly used for public access doors. A common system uses a database with the identification attributes of authorized people.

12.7 Setting up Smart Gardening System

Gardening is a very gratifying hobby. However, because of our busy lifestyles, plants in our gardens might not always get the attention they deserve. What if we could remotely monitor their conditions and give them the care they need? There is a way! In this tutorial, we are going to create an IoT-based smart gardening system using the Adafruit IO.

This smart garden project is a plant environment monitoring system that monitors soil moisture level, air temperature, humidity, and water level. Then, depending on the conditions, the system allows users to deliver what is needed remotely through a mobile app. Moreover, this system can be scaled and expanded to automate the entire gardening process!

We are going to measure the humidity levels, soil water moisture content and detecting if there is a need of water in garden. A pump will be driven if water is required by the plants. It can measure humidity levels between 20 to 80% with 5% margin of error. It requires 3 to 5 volts for its operation.

It is:

- remotely controlled water pump
- automatic irrigation if soil moisture is low
- automatic / manual mode
- soil moisture and water pump status display
- log soil moisture historical values

12.8 Moisture Sensor Pin Diagram:

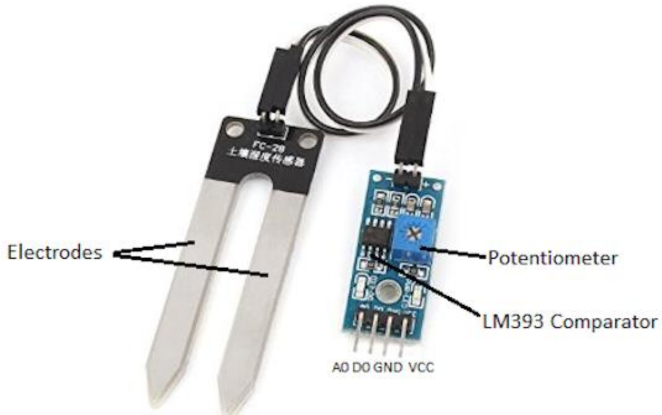


Fig.12.7 Diagram showing Moisture Sensor Pin Diagram

12.9 Applications of Smart Garden

- **Lawn/Gardening management:** If you are a person who remains outside of the house for long hours and you even love to have a garden, but you don't have time to manage it, here is the best solution for you to adopt. Thanks to the IoT developers who have made such powerful sensors that you can install in your garden, and they will come to know when the soil needs water or when the grass needs to be cut.
- **Lawn Irrigation Systems:** The Grass is Always Greener. A lush and healthy lawn is a source of pride for most homeowners, but the weather doesn't always cooperate and provide the adequate elements for a flourishing landscape. For decades we've relied on sprinkler systems to keep our yards at peak presentation average Indian home spends approximately 30% of their daily water usage on lawn and garden maintenance. Nearly half of that amount is wasted due to inefficiency. so this is a great way to initiate smart agriculture.

12.10 Controller

- The controller is the device that receives the sensors' signals, processes them and makes computations on them, and then sends instruction signals to the actuators

- **Dashboard:** The user can display data from a temperature-humidity sensor alongside data from an air quality sensor and add a button to turn on the appliances in the house.

- **Triggers:** Use triggers in Adafruit IO to control and react to your data. Configure triggers to email you when your system goes offline, react to a temperature sensor getting too hot, and publish a message to a new feed.
- It can :

- Display your data in real-time, online
- Make your project internet-connected: Control motors, read sensor data, and more!
- Connect projects to web services like Twitter, RSS feeds, weather services, etc.
- Connect your project to other internet-enabled devices
- The best part? All of the above is do-able for free with Adafruit IO

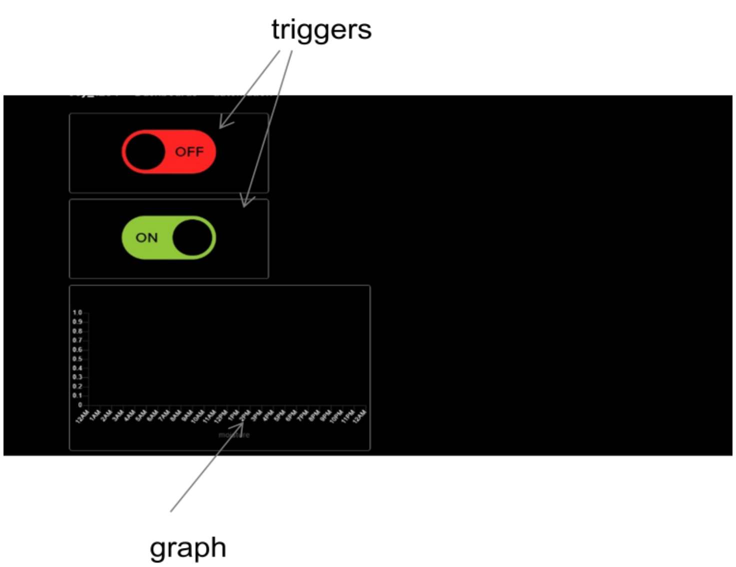


Fig.12.8 Diagram showing Controller Dashboard

12.11 Adafruit

Adafruit.io is a cloud service - that just means it runs for you and you don't have to manage it. You can connect to it over the Internet. It's meant primarily for storing and then retrieving data but it can do a lot more than just that!

It can :

- Display your data in real-time, online
- Make your project internet-connected: Control motors, read sensor data, and more!
- Connect projects to web services like Twitter, RSS feeds, weather services, etc.
- Connect your project to other internet-enabled devices
- The best part? All of the above is do-able for free with Adafruit IO.

Fig.12.9 Diagram showing Adafruit Logo

Implementation of Adafruit.io: In this project, we are going to use the ESP8266 to build the components which are very useful in home automation and smart gardening system: a sensor module, a controller, etc. We can link our project to Adafruit.io in order to make it ubiquitous, which makes it easy for the user to control it from any part of the world at any time.

12.11.1 Steps to create dashboard on Adafruit

- When you login to your io.adafruit.com account, you will be redirected to your list of dashboards. It will look like the page seen below.

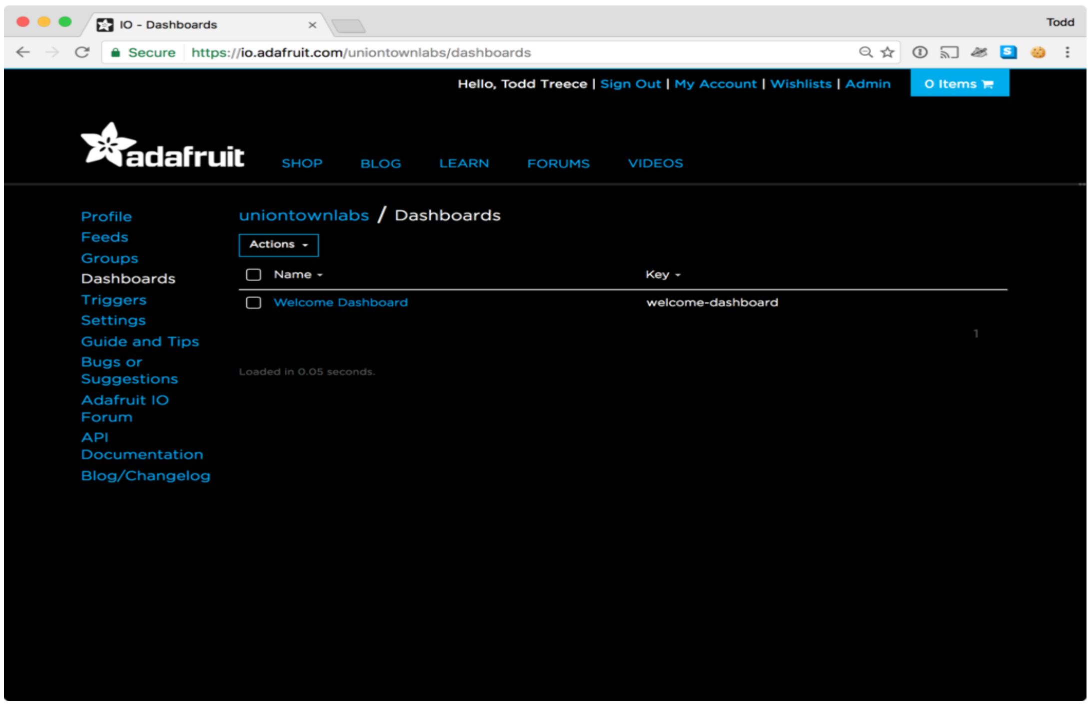


Fig.12.10 Diagram showing Adafruit Interface

- Your list of dashboards will only have the Welcome Dashboard when it is first loaded. You can start the dashboard creation process by clicking the **Actions** menu on the upper left hand side of the screen
- Next, select **Create a New Dashboard** from the dropdown menu.
- You can then enter the name and description of your new dashboard, and click the **Create** button once you are finished.
- Once your dashboard has been created, click on the name of your new dashboard to load it.
- You should now see your new blank dashboard.

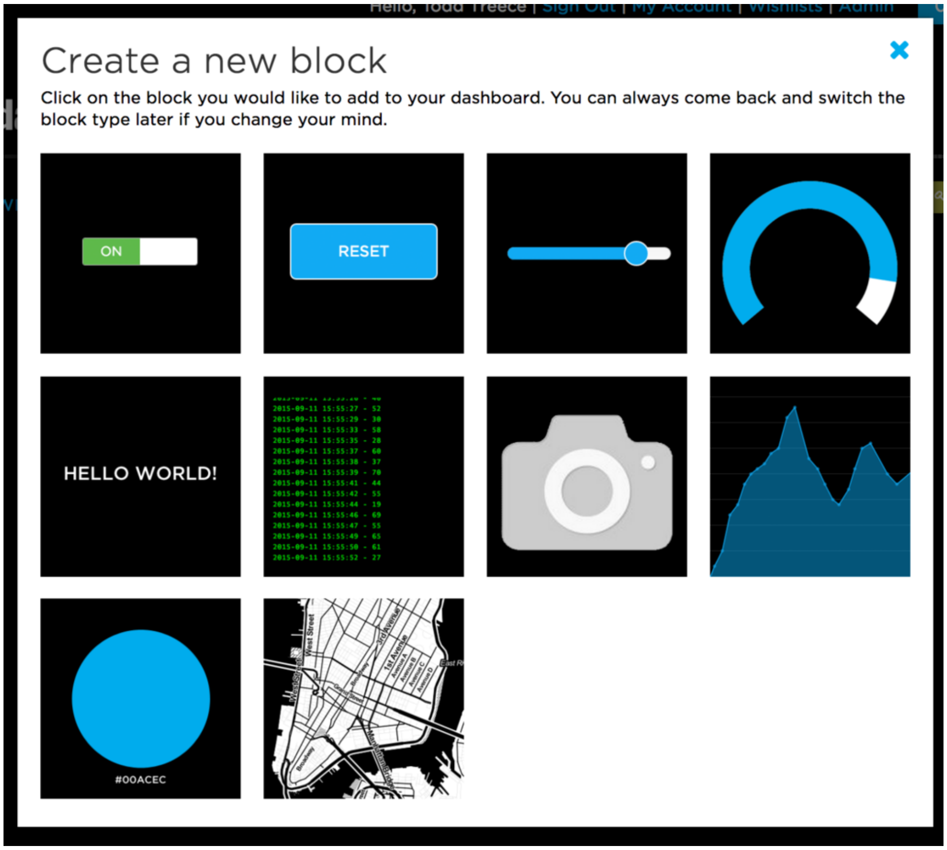


Fig.12.11 Diagram showing Adafruit Blocks

- Blocks are widgets that you can add to your dashboard. There are some blocks that can be used as outputs, and some that can be used as inputs.
- To add a new block, you can click the + (plus) button on the upper right hand side of the dashboard. You will then be presented with a list of block types to choose from, like the one seen below.
- **Toggle Button:** The toggle button will allow you to switch between any two text or numeric values. Unlike the momentary button, the values will stay the same until you click the button again to toggle to the second value.

- **Line Graph:** The line graph allows you to visualize numeric data over time. You can set the time range in hours that the chart will load, and set the labels for each axis of the chart. The chart will update dynamically whenever new values are pushed to the feed.

  Navigating the Create New Block Form

  There are three steps in the create block form:

1. Select block type
2. Select feed(s)
3. Block settings

Once you have selected a new block type by clicking on the icon, you will be presented with a list of feeds. This will allow you to select a feed to attach to your new block. Some blocks, such as the stream and line graph blocks, allow you to attach multiple feeds. Click on the checkbox next to the feed to select it, and click the **Next step** button to continue.

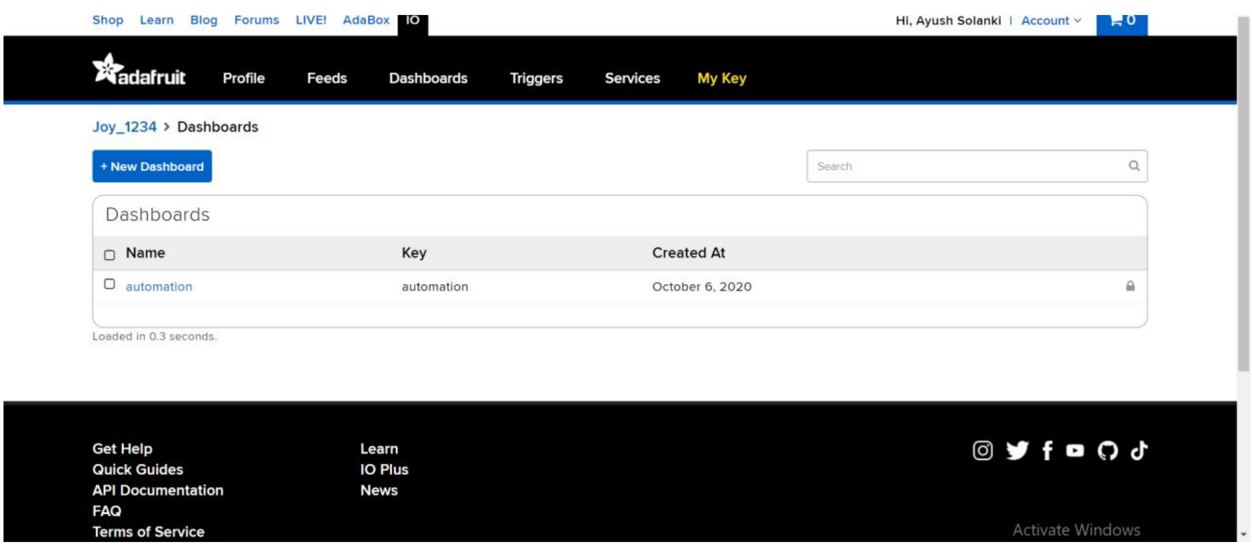


Fig.12.12 Diagram showing Adafruit Dashboards

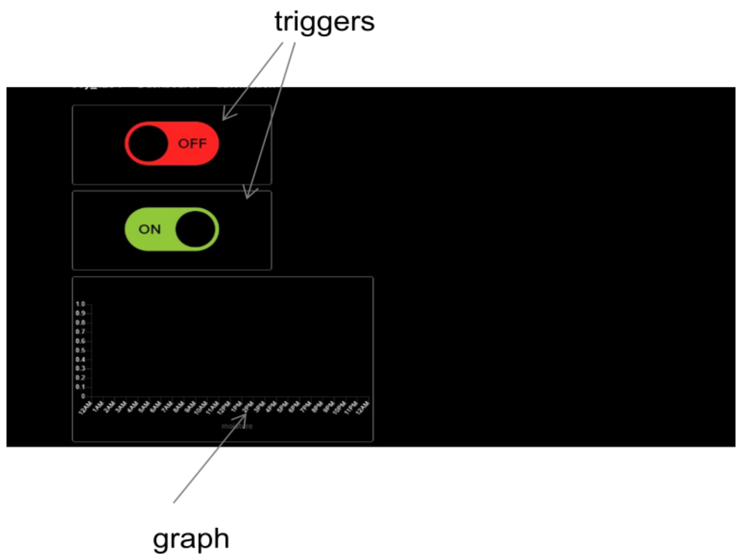


Fig.12.13 Diagram showing Adafruit Final Dashboards (Controller)

12.11.2 Libraries of Adafruit included in Project

- **ESP8266WiFi library:**

The Wi-Fi library for ESP8266 has been developed based on ESP8266 SDK, using the naming conventions and overall functionality philosophy of the Arduino WiFi library. Over time, the wealth of Wi-Fi features ported from ESP8266 SDK to esp8266 / Arduino outgrew Arduino WiFi library and it became apparent that we would need to provide separate documentation on what is new and extra.

This documentation will walk you through several classes, methods and properties of the ESP8266WiFi library. If you are new to C++ and Arduino, don't worry. We will start from general concepts and then move to detailed description of members of each particular class including usage examples.

The scope of functionality offered by the ESP8266WiFi library is quite extensive and therefore this description has been broken up into separate documents marked with : arrow_right :.

- **MQTT Adafruit library:**

MQTT stands for Message Queuing Telemetry Transport. MQTT is a machine to machine internet of things connectivity protocol. It is an extremely lightweight and publish-subscribe messaging transport protocol. This protocol is useful for the connection with the remote location where the bandwidth is a premium. These characteristics make it useful in various situations, including constant environment such as for communication machine to machine and internet of things contexts. It is a publish and subscribe system where we can publish and receive the messages as a client. It makes it easy for communication between multiple devices. It is a simple messaging protocol designed for the constrained devices and with low bandwidth, so it's a perfect solution for the internet of things applications.

Publish-Subscribe Model :

This model involves multiple clients interacting with each other, without having any direct connection established between them. All clients communicate with other clients only via third party known as Broker.

MQTT Client and Broker :

Clients publish messages on different topics to broker. The broker is central server that receives these messages and filters them based on their topics. It then sends these messages to respective clients that have subscribed to those different topics.

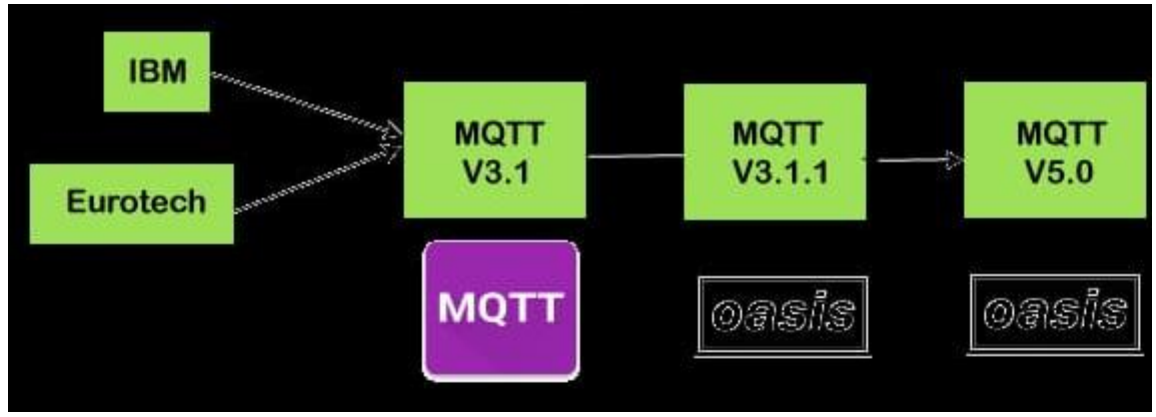


Fig.12.14 Diagram showing Adafruit Final Dashboards (Controller)

12.12 Software: Arduino

The **Arduino Integrated Development Environment (IDE)** is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards.

The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

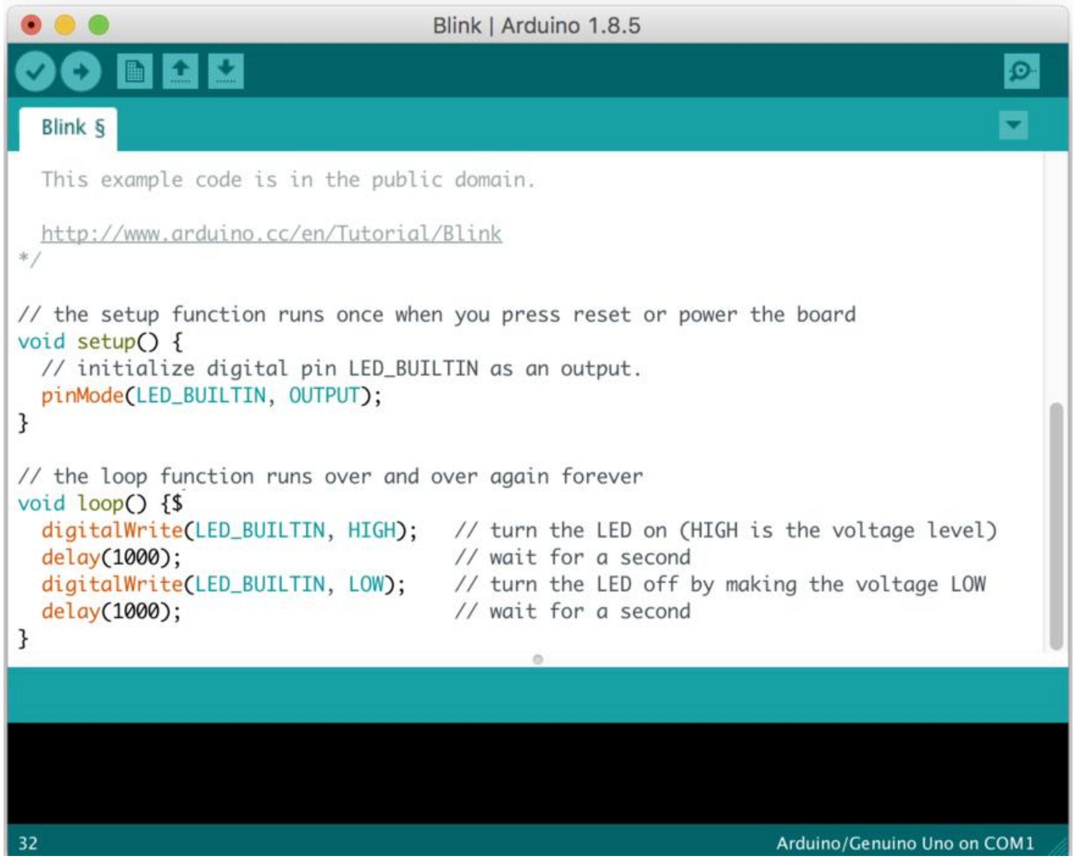


Fig.12.15 Diagram showing Arduino IDE Example

Arduino IDE is a derivative of the Processing IDE, however as of version 2.0, the Processing IDE will be replaced with the Visual Studio Code-based Eclipse IDE framework.

With the rising popularity of Arduino as a software platform, other vendors started to implement custom open source compilers and tools (cores) that can build and upload sketches to other microcontrollers that are not supported by Arduino's official line of microcontrollers.

12.13 Circuit

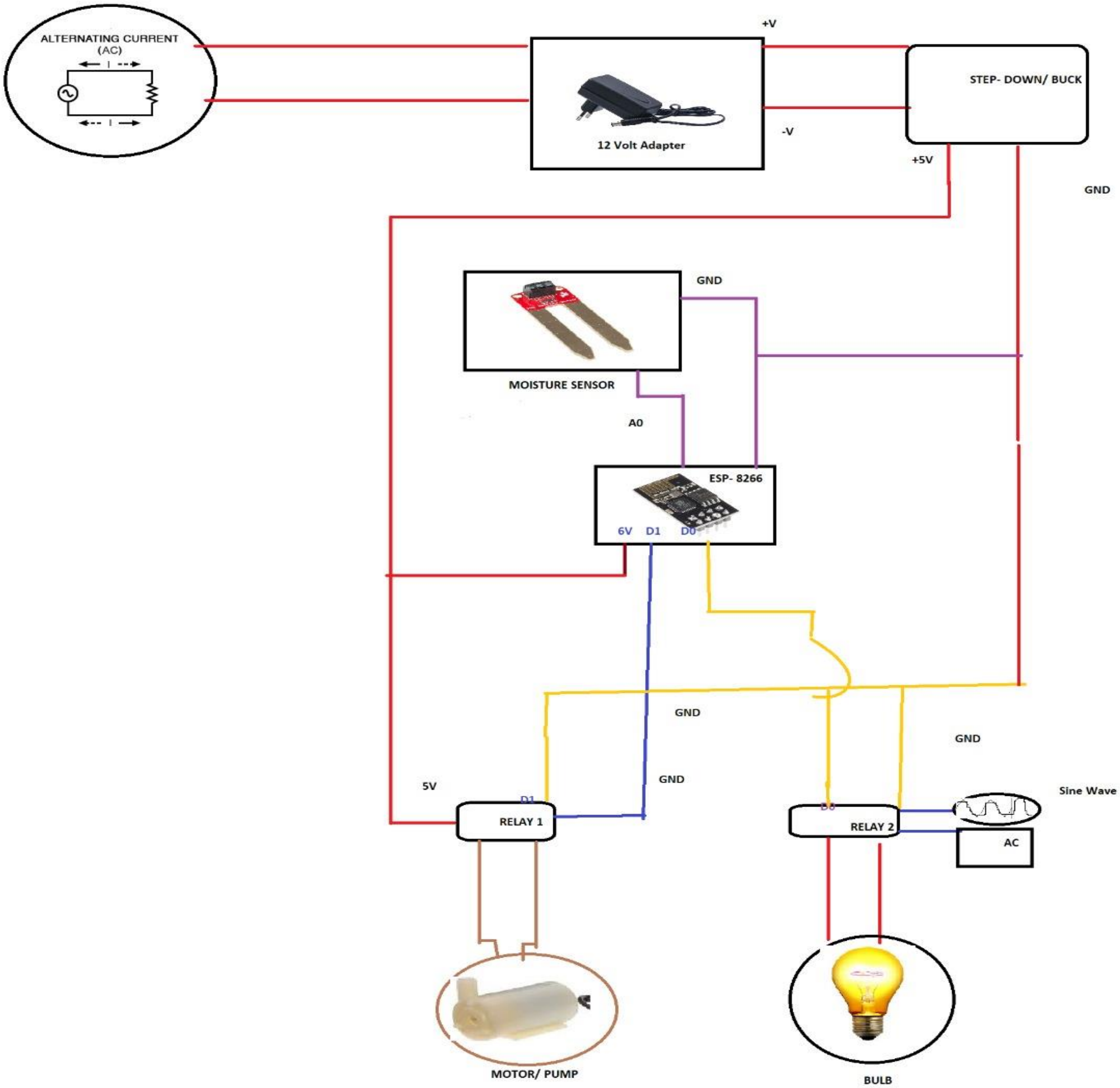


Fig.12.16 Diagram showing Project Circuit

1) **<u>SUPPLY</u> :**

There are 2 A.C. supplies : One is direct to the relay 2 & the other one passes through the adapter and buck converter.

The Supply that passes through buck converter is stepped-down to 5V so as to keep MCU from frying.

2) **<u>ESP8266 MCU</u> :**

This MCU contains pins that are used to connect or pair it with other components. In the figure, the moisture sensor that gives the input to the MCU has 2 pins for supply, one

is connected to ground via buck converter and other is connected to the ESP pin(A0), that gives the positive power supply. Also, ESP is connected to the ground. Now, ESP has 3 more pins (VCC, D0, D1) power supply and selector pins respectively. VCC is connected to the 5V power Supply from Buck Converter and the selector pins D0 and D1 are connected to the Relay 1 and Relay 2 respectively.

3) **<u>MOISTURE SENSOR</u> :**

The moisture sensor after getting supplies is dug inside the soil to sense the humidity or moisture inside the soil to produce input for ESP.

4) **<u>RELAYS</u> :**

The Relays 1 & 2 have have 3 terminals or pins each.

Relay 1: It is connected to GND, 5V suppy, D1 Pin and later the pinout are connected to the motor / pump for water supply.

Relay 2: It is connected to GND, A.C suppy, D0 Pin and later the pinout are connected to the Bulb (Home Appliance) for Home Automation activities.

Relays works as a switching device that enables functioning of components.

12.14 Internet Connectivity

Fig.12.17 Diagram showing Internet Connectivity

The **concept of Home Automation** aims to bring the control of operating your every day home electrical appliances to the tip of your finger, thus giving user affordable lighting solutions, better energy conservation with optimum use of energy. Apart from just lighting solutions, the concept also further extends to have a overall control over your home security as well as build a centralised home entertainment system and much more. The **Internet of Things** (or commonly referred to as IoT) based Home Automation system, as the name suggests aims to control all the devices of your smart home through internet protocols or cloud based computing.

The IoT based Home Automation system offer a lot of flexibility over the wired systems s it comes with various advantages like ease-of-use, ease-of-installation, avoid complexity of running through wires or loose electrical connections, easy fault detection and triggering and above and all it even offers easy mobility.

12.15 Setting up threshold value

The FC-28 soil moisture sensor has four pins:

VCC: Power

A0: Analog Output

D0: Digital Output

GND: Ground

The module also contains a potentiometer, which will set the threshold value. This threshold value will be compared by the LM393 comparator. The output LED will light up and down according to this threshold value.

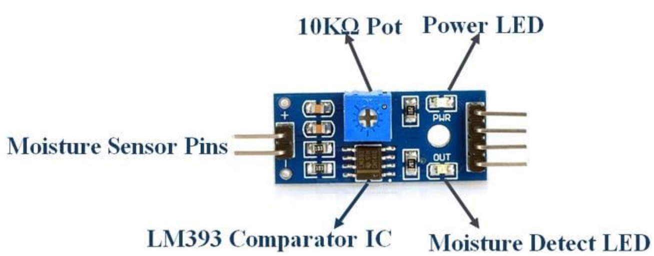


Fig.12.18 Diagram showing Comparator

12.16 Features

- Highly secure system.
- Can be controlled worldwide.
- Can be used by multiple users.
- Can use multiple devices.
- Wireless connectivity.
- Real time sensing.

12.17 Applications of project

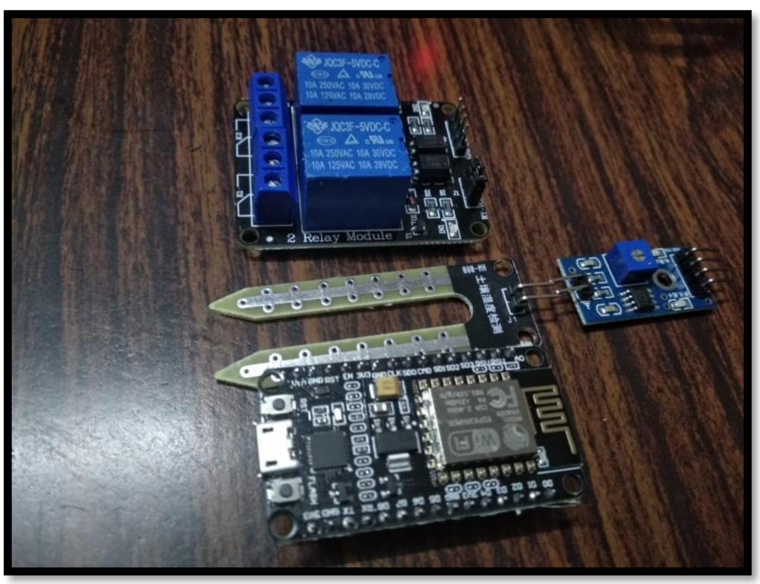
- Lighting control.
- Smart Home Appliances.
- Improved Home safety and security.
- Home air quality and water quality monitoring.
- Smart Switches.
- Smart Energy Meters.

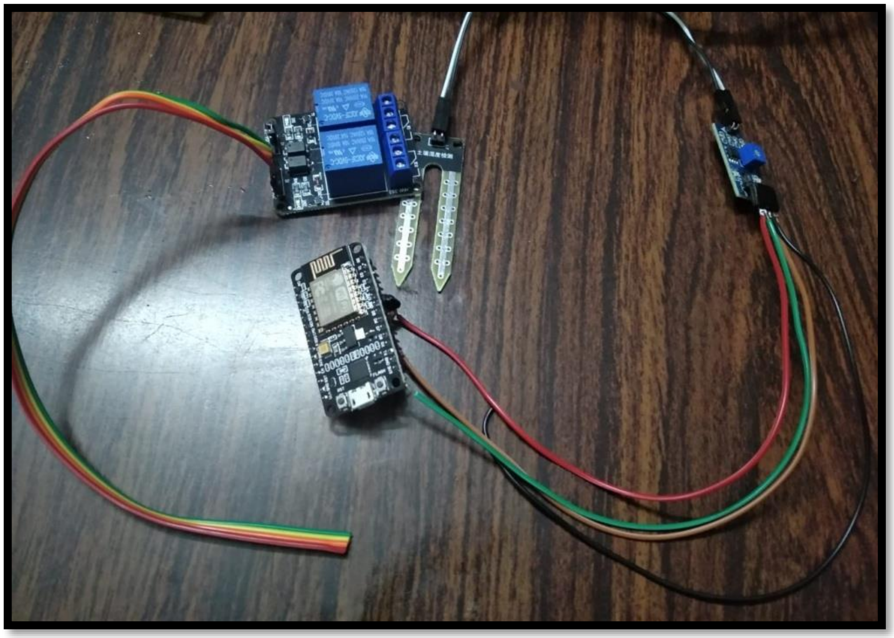
12.18 Why this project is unique?

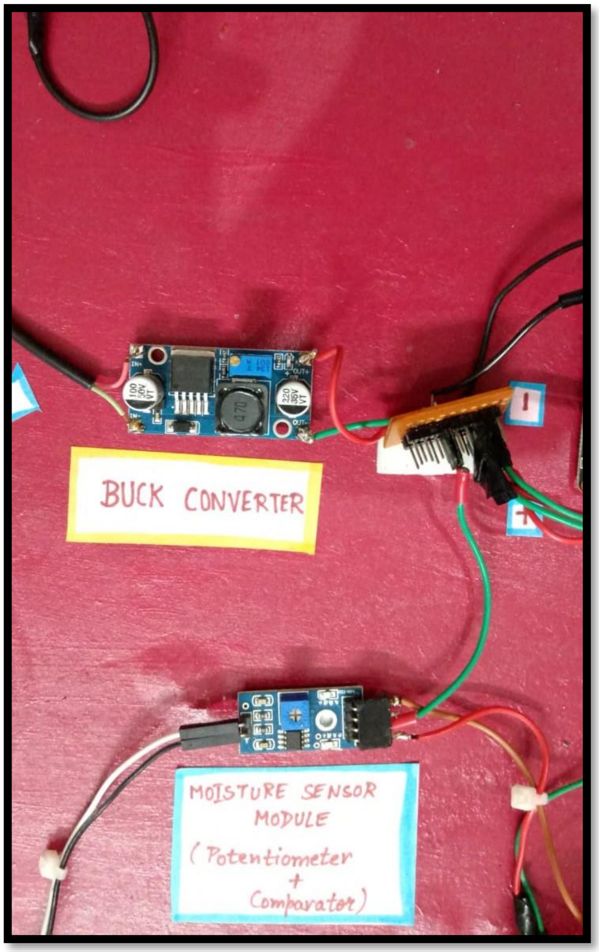
- Increased efficiency, control, and customization
- Integration of Smart home devices
- smart irrigation system extends watering time for plants, and provides ideal growth condition.

- It saves time and timer delay as per the environmental condition can be added for automatic watering.
- This smart irrigation system can be adjusted and modified according to the changing environment.
- It works on a highly secured connection.
- It requires an user ID and a passkey to log in to the system.
- The user ID on which a controller is designed is a pseudo ID, i.e. it does not exist in real world and hence have no data attached to it.
- The code over which our system functions also requires the same ID and passkey for running the system.

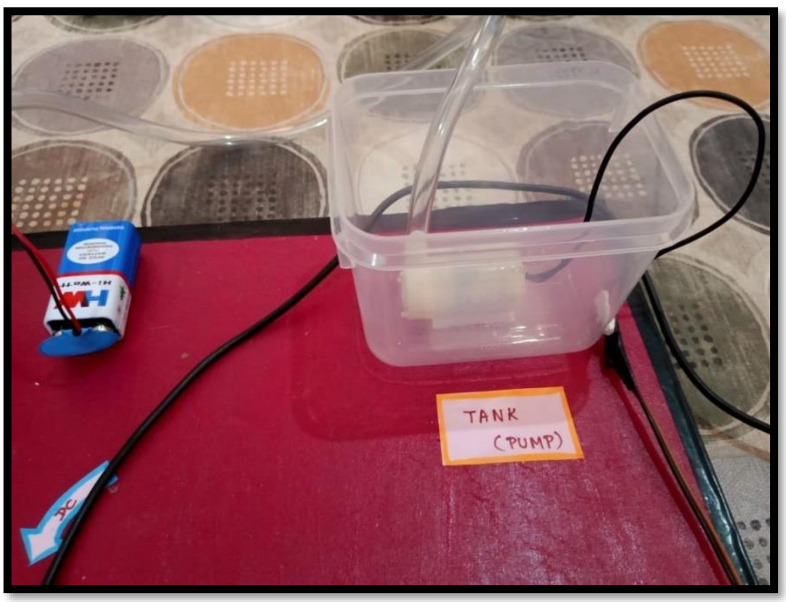
12.19 Snapshots of our Project



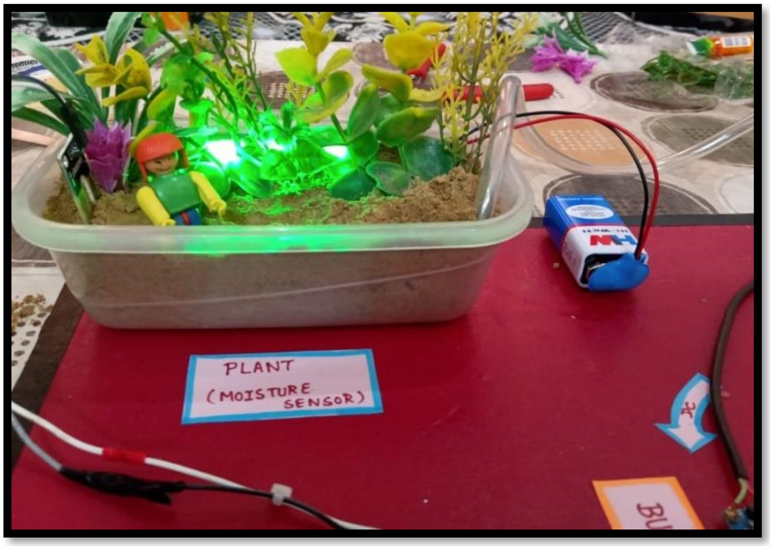


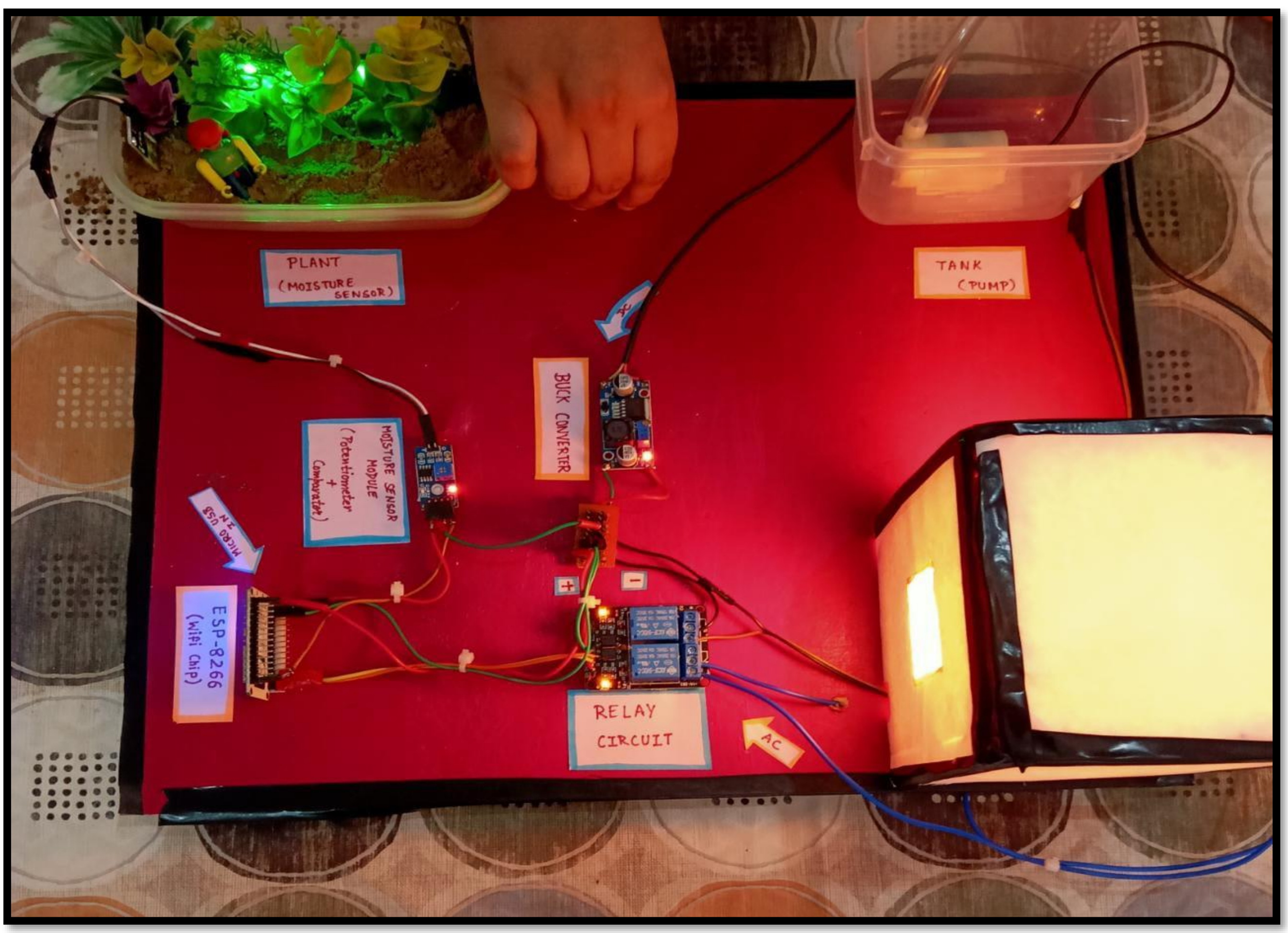

BUCK CONVERTER
MOISTURE SENSOR MODULE
(Potentiometer + Comparator)


Vin GND RST EN 3V3 GND CLK SDO CMD SD1 SD2 SD3 RSV RSV A0
RST
FLASH


TANK
(PUMP)

\


PLANT
(MOISTURE SENSOR)


PLANT
(MOISTURE SENSOR)
TANK
(PUMP)
BUCK CONVERTER
MOISTURE SENSOR MODULE
(Potentiometer + Comparator)
MICRO USB IN
ESP-8266
(wifi Chip)
RELAY
CIRCUIT
AC

Chapter 13

Scope of IOT

13.1 Introduction

Internet of Things has emerged as a leading technology around the world. It has gained a lot of popularity in lesser time. Also, the advancements in Artificial Intelligence and Machine Learning have made the automation of IoT devices easy. Basically, AI and ML programs are combined with IoT devices to give them proper automation. Due to this, IoT has also expanded its area of application in various sectors. Here, in this section, we will discuss the applications and the future scope of IoT in healthcare, automotive, and agriculture industries.

13.2 Healthcare

In this section, we will see the future of IoT in healthcare sector. IoT has proved to be one of the best tools for the healthcare industry. It helps provide advanced healthcare facilities to patients, doctors, and researchers. These facilities include smart diagnosis, wearable devices for tracking health, patient management, and many more. Furthermore, IoT devices have reduced unnecessary strain on the healthcare system.

Fig.13.1: Healthcare in IOT

The healthcare devices can directly send the patients' data health to doctors over a safe network. This allows the doctors to diagnose the patients from remote locations.

Reasons for IoT being beneficial in the healthcare industry are as follows:

13.2.1 Error reduction in treatments

With the help of IoT devices, there is a decrease in manual errors in diagnosing patients. Thus, patients can get proper treatment on time. Also, the 24/7 diagnosis through gadgets gives a more clear idea about patients' health when compared to manual diagnosis.

13.2.2 Decrease in the cost of treatments

Manual diagnosis requires time and the use of various types of costly equipment, along with other hospital charges. Due to this, the overall cost of the treatment increases. We can reduce these costs by using IoT gadgets. Also, the hospital charges and congestion in hospitals can be reduced as patients can be diagnosed from their respective locations.

13.2.3 Availability of specialists in remote locations

Internet of Things solves one of the major problems of the healthcare industry that is the availability of doctors, especially specialists, in remote locations. With the help of IoT devices, the treatment of patients in the physical absence of doctors has become possible. The patients just need to wear the device. Then, the device will send all the real-time data of patients' health to the respective doctors for analyzing the condition. In this way, the scope of Internet of Things is helping the healthcare sector give proper treatment to the needy.

13.3 Agriculture

One of the three basic human needs is food. To fulfill the need for food, we do farming. However, now, as the population of the world is increasing, the agricultural industry is facing many challenges. Also, changes in weather conditions and climate hugely impact the agricultural industry. To meet the rising demand for food, the industry has hence adopted technology to increase productivity. It includes the use of precision farming, agricultural drones, and smart farming application.

All these are built on top of the application of Internet of Things. Now, let us discuss how precision farming, smart farming applications, and agricultural drones help in increasing the productivity of the land.

13.3.1 Precision farming

In agriculture, Information and Communication Technology is a tool used for smart farming. With the help of IoT-based devices, crop fields are observed. The technology uses sensors to calculate the moisture of soil, humidity, and temperature. Also, it uses an automated irrigation system to make efficient use of water. Precision farming helps farmers monitor their fields and boost productivity.

13.3.2 Agricultural drones

Drones used for agriculture and farming are one of the best applications of Internet of Things. They are used to enhance agricultural processes. We use agricultural drones for planting crops, irrigating fields, spraying of pesticides, and monitoring the fields. With the help of

drones, it becomes easier to evaluate the health of crops. This is all possible with the help of smart IoT-based devices that are used to make agricultural drones.

13.3.3 Smart greenhouses

Farmers use greenhouse farming to enhance the productivity of crops. In greenhouse farming, the environmental factors that affect the growth of crops are controlled by manual intervention. However, manually controlling the mechanism for the growth of crops is less productive. The emergence of IoT and technological advancements has led to the creation of IoT-based greenhouses that consist of various devices such as sensors, climate controllers, etc.

These IoT devices help in measuring the various environmental conditions according to the requirements of plants. As all sensors and devices connect over the Internet servers, they provide accurate information on the environmental conditions. Then, the devices activate actuators to control heaters, fans, windows, and lighting of greenhouses to set according to the environment.

This is how the scope of IoT is enhancing the productivity of the agricultural industry.

13.4 Automotive Industry

In the 21st century, the application of IoT is revolutionizing the automotive industry. One of the major applications is the creation of self-driving cars that has changed the trends of the automotive industry. Engineers have created self-driving cars to reduce manual errors and ensure a safe drive. Various companies around the world are creating self-driving vehicles including Google, Tesla, Mercedes-Benz, Volvo, Audi, and many more. These self-driving cars use various technologies such as Data Science, Artificial Intelligence, Deep Learning, and IoT. IoT devices are programmed in such a way that they assist in creating an automated system for self-driving cars.

These IoT devices consist of HD cameras, thermal sensors, smart navigators, speed controller, rain sensors, wireless connectivity, and proximity sensors. While using these cars, you need to enter your location and the destination. Then, the navigator helps locate the destination and tries to find the shortest path. After that, the IoT-based HD cameras help in getting the visuals of the surrounding and send the data to the AI-based systems. These systems analyze and visualize the data of the surrounding and accordingly fix the response of the self-driving cars.

Also, there are IoT-based speed controllers that help regulate the speed of these cars according to the traffic and congestion. This is how the scope of IoT is changing the trends in the automotive industry.

13.5 Jobs in IoT

As India is a developing country, it has a wide IoT scope. According to Naukri.com, the future scope of IoT in India is very high as there are 117,114 job openings for an IoT Developer here. On the other hand, in the United States, the demand for an IoT Developer has jumped over 300 percent.

As we know that today every electronic device comes with an installed IoT system, there is a bright future scope of IoT. Let us look at the below graph to know the stats of jobs that has been available for an IoT Developer over the years.

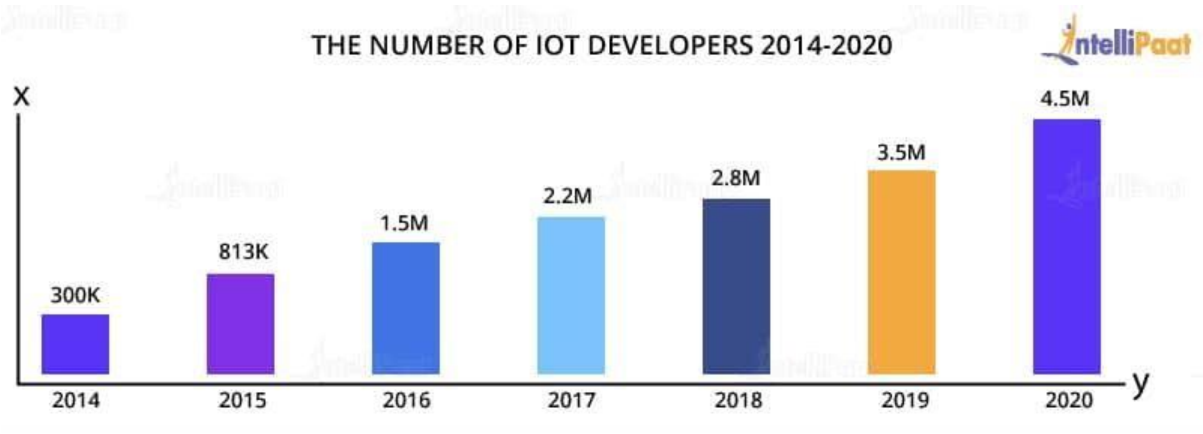


Fig.13.2: Jobs in IoT

Coming to the salary figures, the average salary provided to an IoT Developer in the United States is US$164,417 per year. In India, the average annual salary is ₹850k. This is an attractive package to start your career with.

This is all about the future scope of IoT and how it is changing the world with its applications.

LITERATURE REFERENCES

- Dimitrios Serpanos Marilyn Wolf, "Internet- of- Things (IOT) systems, Architectures, Algorithms, Methodologies", ISBN 978-3-319-69715-4, Springer International Publication, INC. 2018.
- "The Internet of Things" by Samuel Greengard
- "The Fourth Industrial Revolution" by Klaus Schwab
- Arduino – Home [www.arduino.cc]
- https://io.adafruit.com
- Armando Roy Delgado, Rich Picking and Vic Grout,"Remote-Controlled Home Automation Systems with Different Network Technologies", Centre for Applied Internet Research (CAIR), University of Wales, NEWI, Wrexham, UK.
- Margaret Mouse. "Frequency-hopping spread spectrum".
- Kim Baraka, Marc Ghobril, Sami Malek, RouwaidaKanj, AymanKayssi "Low cost Arduino/Android-based Energy-Efficient Home Automation System with Smart Task Scheduling", 2013.
- www.wikkipedia.com
- www.tutorialspoint.com
- Edward Ashford Lee & Sanjit Arunkumar Seshia, "Introduction to Embedded Systems.. A Cyber- physical System Approach", UC Berkely, 2010.