

RASPBERRY PI BASED VEHICLE SECURITY SYSTEM AND AUTHENTICATION

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Engineering Degree in Electronics and Communication Engineering

By

HARISH.N (38130081)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY (DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

www.sathyabama.ac.in

**DEPARTMENT OF ELECTRONICS AND
COMMUNICATION ENGINEERING**

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of HARISH.N(38130081) who carried out the project entitled “**RASPBERRY PI BASED VEHICLE SECURITY SYSTEM AND AUTHENTICATION**” under our supervision from September 2021 to April 2022.

Internal Guide

Dr. I. Rexiline Sheeba, M.E., Ph.D.

Associate Professor,

Department of Electronics and Communication Engineering.

Head of the Department

Dr. T. RAVI, M.E., Ph.D.

Department of Electronics and Communication Engineering.

Submitted for Viva voce examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, HARISH.N (38030081) hereby declare that the Project Report entitled **“RASPBERRY PI BASED VEHICLE SECURITY SYSTEM AND AUTHENTICATION”** done by us under the guidance of **Dr. I. Rexiline Sheeba M.E., Ph.D.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

1)

DATE:

PLACE:

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA, M.E., Ph.D. Dean, School of Electrical and Electronics Engineering** and **Dr. T. RAVI, M.E., Ph.D. Head of the Department, Department of Electronics and Communication Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. I. Rexiline Sheeba M.E., Ph.D. Department of Electronics and Communication Engineering** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of the project work.

We wish to express our thanks to all teaching and non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

We express our gratitude to our parents for their constant encouragement and support for the completion of the project.

ABSTRACT

Nowadays, everyone is more concerned about their safety. Theft is becoming a major issue at an alarming rate. It is our responsibility to protect our assets. Because everyone has a vehicle, we developed this strategy to increase vehicle security. Biometric identification is quick, simple, reliable, and accurate. This ensures that the vehicle's security system is functioning properly. Only authorized individuals can start the vehicle since fingerprints of known individuals are registered. The PI camera is permanently installed in the vehicle and activates when the fingerprint is not matched. The image acquired by the camera is delivered to the owner via the applications that have been built. The owner will get to know the person through this program, and he will be able to control the vehicle. The GPS is used to track the vehicle's location. The owner can also see the location utilizing the theft software. And this system is implemented with accident detection and alarm system with that camera takes the Picture for any further clarification. Camera Module capture images when the people Crashing the vehicle.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
1.	INTRODUCTION	
	1.1 PURPOSE OF THE_PROPOSAL	8
	1.2 EMBEDDED SYSTEM	8
	1.3 CLASSIFICATIONS OF EMBDDED SYSTEMS	10
	1.4 BLOCK DIAGRAM OF EMBEDDED SYSTEM	11
	1.5 EMBEDDED SYSTEMS APPLICATIONS	12
2.	LITERTATURE SURVEY	
	2.1 LITERTATURE SURVEY	13
	2.1.1 ACCIDENT MANAGEMENT USING WIRELESS NETWORKS	
	2.1.2MULTI-MODAL ALERT SYSTEM IMPLEMENTATION FOR ITS	
	2.2 OBJECTIVE	14
	2.3 LITERTATURE REVIEW	14
3.	AIM AND SCOPE OF THE SYSYEM	
	3.1 EXISTING METHODOLOGIES	15
	3.2 PROPOSED METHODOLOGIES	15
	3.3 BLOCK DIAGRAM	16
	3.4 HARDWARE REQUIREMENTS	16
	3.5 SOFTWARE REQUIREMENTS	16
4.	HARDWARE DESCRIPTIONS	
	4.1 RASPBERRY PI:	17
	4.1.1 PROCESSER	18
	4.1.2 PERFORMANCE	19
	4.1.3 OVERCLOCKING	20
	4.2 IR SENSOR:	20

CHAPTER NO.	TITLE	PAGE NO.
4.	HARDWARE DESCRIPTIONS	
	4.3 WEB CAMERA	26
	4.3.1 DEFINITION	26
	4.3.2 FUNCTIONS	28
	4.4 VIBRATION SENSOR:	28
	4.5 MEMS SENSOR	32
	4.6 BUZZER	34
	4.7 FINGER PRINT SENSOR:	39
	4.8 MOTOR:	42
	4.9 GSM	46
	4.10 POWER SUPPLY UNIT	50
5.	SOFTWARE DESCRIPTIONS	
	5.1 RASPBIAN OS	54
	5.1.1 FEATURES	55
	5.2 PYTHON	56
	5.3 EXAMPLE OF PYTHON PROGRAMMING	65
	5.4PYTHON IS USED BECAUSE:	67
	5.4.1 ADVANTAGES OR BENEFITS OF PYTHON	69
	5.4.2 PYTHON FEATURES	70
	5.4.3 LANGUAGES INFLUENCED BY PYTHON	72
	5.5 APPLICATIONS:	73
	5.6 FEATURES OF PYTHON	74
6.	CONCLUSION	77
7.	REFERENCE	78

CHAPTER-1

INTRODUCTION

1.1 PURPOSE OF THE PROPOSAL

Recently, traffic on roads has been increasing rapidly as a result of which it is becoming extremely difficult to manage vehicles in terms of traffic management, vehicle theft and accidents. According to Mansell, one in every 42 vehicles in the United States is stolen estimating a loss of over \$9 billion per annum. A number of tools and techniques are used to minimize the probability of such incidents while preserving the safety of the people involved. Vehicle tracking is an old concept and has been implemented across the globe for tracking stolen vehicles or sometimes for personal safety.

A number of methods have been proposed and implemented worldwide to overcome these issues. However, the cost of these solutions varies and is dependent on the technology used. The most common tool used today is Global Positioning System (GPS), which is discussed further down this paper. In order to obtain high reliability and accuracy one cannot rely on GPS because of its dependency on the network availability and same is the case with Global System for Mobile (GSM). As suggested by Mansell, a number of electronic homing devices have also been developed, which are capable of track vehicles but only after it is reported stolen. Hence such systems are useful only in geographical locations where the police receivers are installed. In case of other emergencies such as accidents, breakdowns or medical emergencies, there is also a need of quick tracking of vehicles for a faster response. Most of these solutions rely on computer programs that display the map of active vehicles indicating their location on the map. The location of these vehicles shows up on the map in form of a blinking-colored dot.

The purpose of this work is to present low-cost and accurate vehicle tracking mechanism for security purpose. It opens the door not only for vehicle tracking but also traffic management, parking management, speed limit enforcement.

1.2 EMBEDDED SYSTEM:

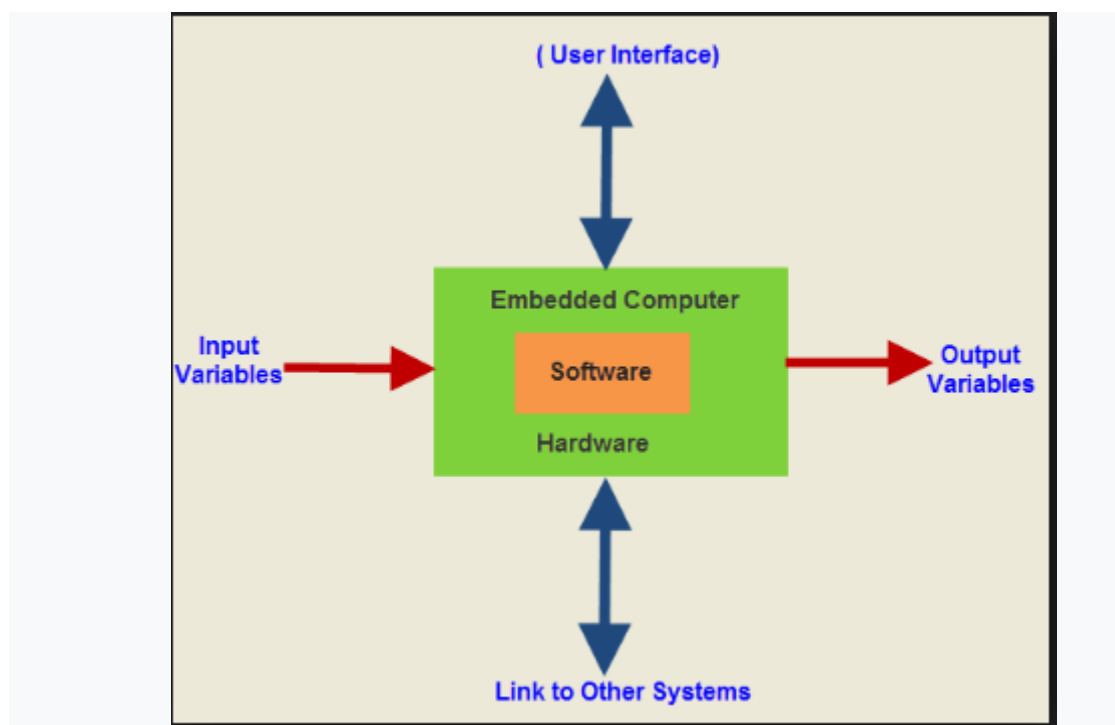
An embedded system is a controller programmed and controlled by a real-time operating system (RTOS) with a dedicated function within a

larger mechanical or electrical system, often with real-time consumption of embedded systems computing constraints.

It is embedded as part of a complete device often including hardware and mechanical parts. Embedded systems control many devices in common use today. Ninety-eight percent of all microprocessors are manufactured to serve as embedded system component.

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, which make them significantly more difficult to program and to interact with.

However, by building intelligence mechanisms on top of the hardware, taking advantage of possible existing sensors and the existence of a network of embedded units, one can both optimally manage available resources at the unit and network levels as well as provide augmented functions, well beyond those available. For example, intelligent techniques can be designed to manage power.



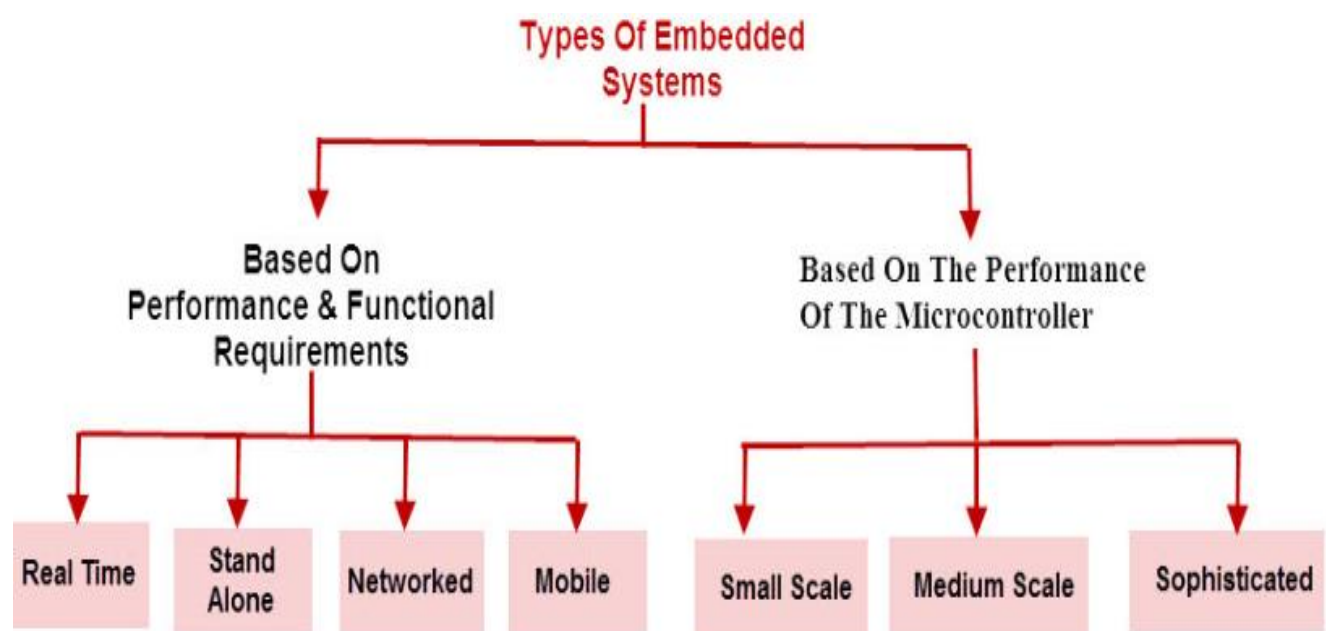
Embedded Computer Sub-Assembly for Electronic Voting Machine

Embedded systems are commonly found in consumer, industrial, automotive, medical, 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, 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.

1.3 CLASSIFICATIONS OF EMBDDED SYSTEMS:



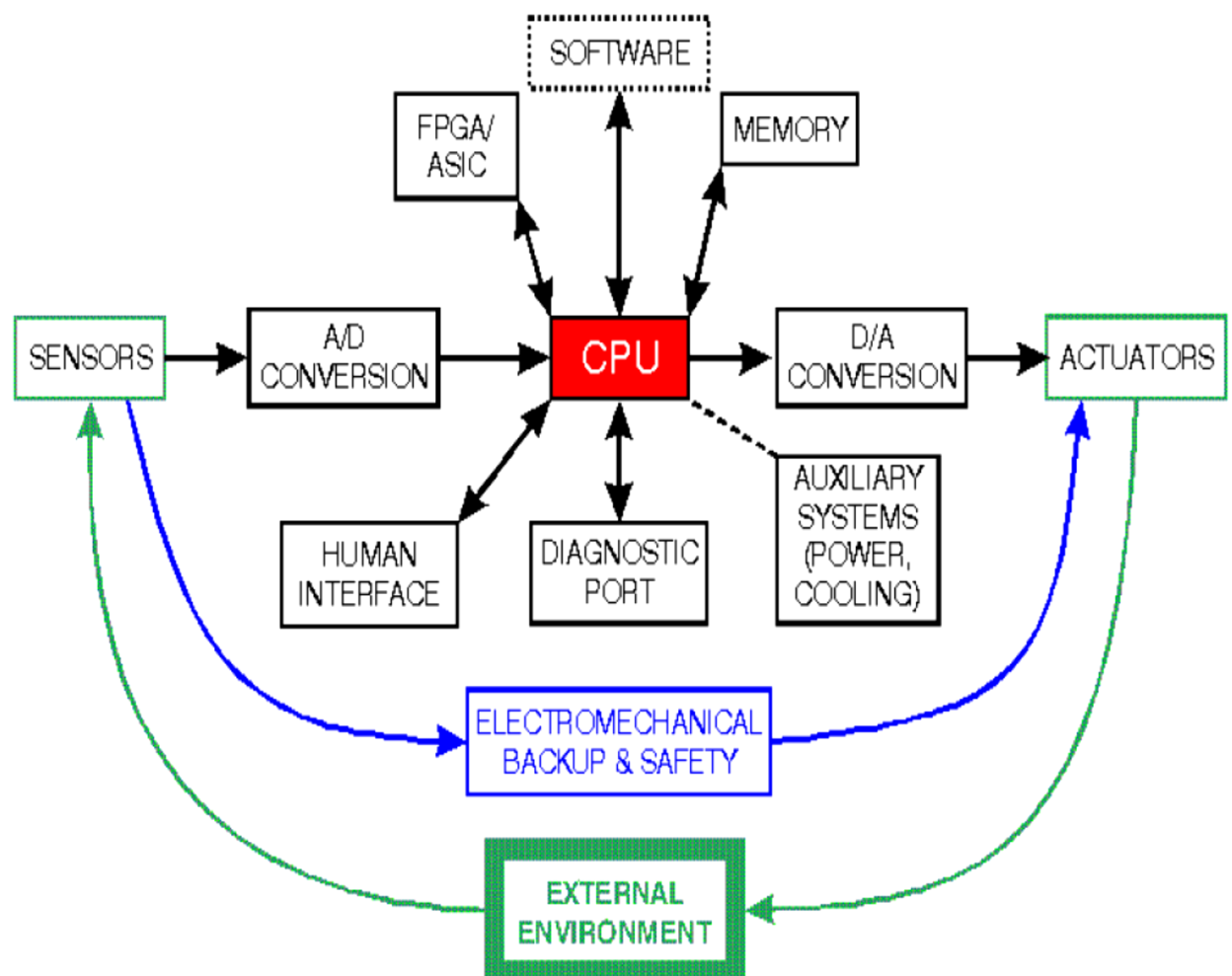
Advanced HVAC systems use networked thermostats to more accurately and efficiently control temperature that can change by time of day and season. Home automation uses wired- and wireless-networking that can be used to control lights, climate, security, audio/visual, surveillance, etc., all of which use embedded devices for sensing and controlling.

like traffic lights, factory controllers, and largely complex systems like hybrid vehicles, MRI, and avionics Embedded systems range from

portable devices such as digital watches and MP3 players, to large stationary installations. Complexity varies from low, with a single microcontroller.

1.4 BLOCK DIAGRAM OF EMBEDDED SYSTEM:

An embedded system usually contains an embedded processor. Many appliances that have a digital interface microwave, VCRs, cars utilize embedded systems. Some embedded systems include an operating system. Others are very specialized resulting in the entire logic being implemented as a single program. These systems are embedded into some device for some specific purpose other than to provide general purpose computing.



Block diagram of a typical embedded system

1.5 EMBEDDED SYSTEMS APPLICATIONS:

Embedded systems in automobiles include motor control, cruise control, body safety, engine safety, robotics in an assembly line, car multimedia, car entertainment, E-com access, mobiles etc.

- Embedded systems in telecommunications include networking, mobile computing, and wireless communications, etc.
- Embedded systems in smart cards include banking, telephone and security systems.
- Embedded Systems in satellites and missiles include defense, communication, and aerospace
- Embedded systems in computer networking & peripherals include image processing, networking systems, printers, network cards, monitors and displays
- Embedded Systems in digital consumer electronics include set-top boxes, DVDs, high-definition TVs and digital cameras

CHAPTER-2

LITERATURE SURVEY

2.1 LITERATURE SURVEY:

2.1.1 ACCIDENT MANAGEMENT USING WIRELESS NETWORKS

In this paper, a new framework for vehicular accident management using wireless networks is proposed and discussed. We present integrated wireless network architecture for next generation all IP wireless networks which supports our framework. In this architecture we show that different wireless Networks can provide Internet access for moving vehicles. A new distributed system is developed to manage vehicle accidents (anywhere, anytime) such that it communicates accident information with traffic police department, hospital, and Insurance Company. The proposed system is implemented using a test bed. Then the results are discussed.

2.1.2 MULTI-MODAL ALERT SYSTEM IMPLEMENTATION FOR ITS

The rapid technological growth is now providing global opportunities to enable intelligent transportation system (ITS) to tackle road accidents which is considered one of the world's largest public injury prevention problem. For this purpose, eCall is an initiative by European Union (EU) with the purpose to bring rapid assistance to an accident location. This paper¹ presents HDy Copilot, an application for automatic accident detection integrated with multimodal alert dissemination, via both eCall and IEEE 802.11p (ITS-G5). The proposed accident detection algorithm receives inputs from the vehicle, via ODB-II, and from the smartphone sensors, namely the accelerometer, the magnetometer and the gyroscope. An Android smartphone is used as human machine interface, so that the driver can configure the application, receive road hazard warnings issued by other vehicles in the vicinity and cancel countdown procedures upon false road vehicle crash detection. The HDy Copilot is developed for Android OS as it provides open-source APIs that allow access to its hardware resources. The application is implemented, tested and connected to an IEEE 802.11p based prototype. The generated results show that the application successfully detects collisions, rollovers, performs the eCall along with sending Minimum Set of Data (MSD) and Decentralized Environmental Notification Message (DENM).

2.2 OBJECTIVE:

A completely unique approach to extend the protection of road travel victimization the ideas of wireless detector network and therefore the Bluetooth protocol has been protected. It mentioned however, vehicles will type mobile ad-hoc network and exchange information perceived by the onboard sensor. Platform of the robot in operation system and software system development atmosphere welltried optimum resolution for public safety just in case of accident. An honest survey of victimization personal itinerant, Microcontroller, Bluetooth and JAVA Technology has been well-tried. It developed integrated system to manage, management associated monitor accessories within the vehicle so as to attain the concept of an intelligence automobile with ability to uses personal mobile hand phone as a far of interface. Sensible phone-based accident detection will scale back overall traffic jam and awareness of emergency responders. This approach conjointly has been projected.

2.3 LITERTATURE REVIEW:

Now-a-days, mobile phone is used almost by all people. With internet usage are also at all. So, these mobile phones also provide communication platform as they are equipped with 2G & 3G network. There are lots of cause of accident of car and they are drunkenness of driver, drowsiness of driver, unconsciousness of driver and many time what happen driver is not responsible for accident but their neighboring car behavior also have made role to enforce accident. There are also some systems have been implemented to avoid the accident but that do not give proper solution to implemented in car to avoid various accidents that they are normally being happen. For example, when driver at speed suppose 80km/h suddenly stop ignition system may leads to changes of dangerous accident. There are several efforts, application: approaches are projected to produce security and safety just in case accident.

CHAPTER-3

AIM AND SCOPE OF THE SYSYEM

3.1 EXISTING METHODOLOGIES:

- In existing system, they used accident notification systems, so we can get the alert messages whenever the accident occurs.
- For the accident prevention they used smart helmet method, so the helmet will alert the driver.
- In existing system, the algorithm-based approaches are used for the accident alert conditions and location access.
- In existing system, the theft control is done by using camera-based approach, but the face recognition has some problems while using with twins or any mask-based conditions.

Disadvantage:

- By using camera-based approach, the privacy is affected.
- Algorithm based approach is not must effective.

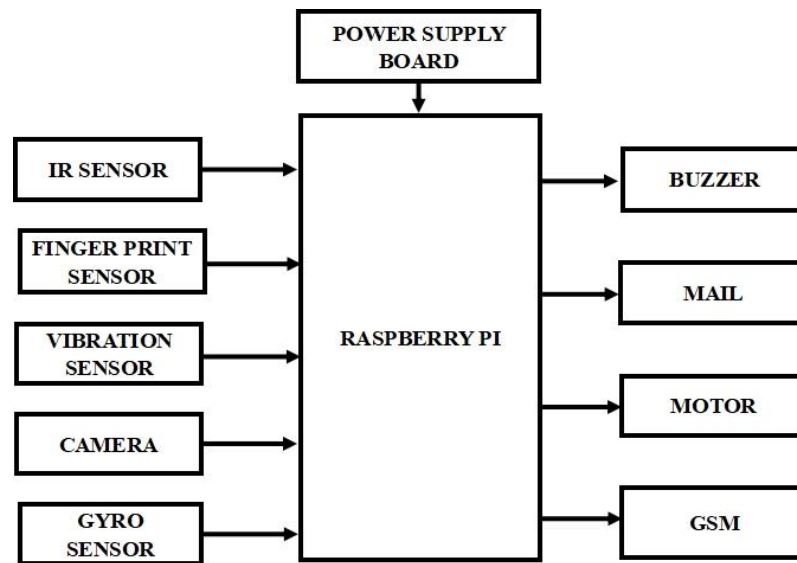
3.2 PROPOSED METHODOLOGIES:

- A Raspberry Pi-based smart car security system for theft control and accident notification as a part of smart city development.
- The finger print module is placed in the car door. Whenever the finger is matched with the finger print module, the door will automatically turn ON, otherwise it will give the buzzer alert and the camera will capture the images of that unauthentication.
- An accident detection system, which also employs vibration sensors and gyro sensor, sends an alert message to a family member via SMS. Then the camera takes some images from outside, and the images are sent by email to the authorised person.
- And in the event of a crash, the vehicle's system has an IR sensor to detect the vehicle that comes closer to our vehicle, and the camera takes images of that occurrence. so that we can know the vehicle number of the crashed person.

Advantage:

- By using biometric, the vehicle is more secured.
- Message can be immediately sending the respective person, so that the death can be reduced.

3.3 BLOCK DIAGRAM:



3.4 HARDWARE REQUIREMENTS:

- RASPBERRY PI
- IR SENSOR
- CAMERA
- FINGERPRINT SENSOR
- VIBRATION SENSOR
- MOTOR
- BUZZER
- GSM
- GYRO SENSOR (MEMS SENSOR)
- POWER SUPPLY

3.5 SOFTWARE REQUIREMENTS:

- PYTHON
- RASPBIAN OS
- OPEN CV

CHAPTER-4

HARDWARE DESCRIPTIONS

4.1 RASPBERRY PI:

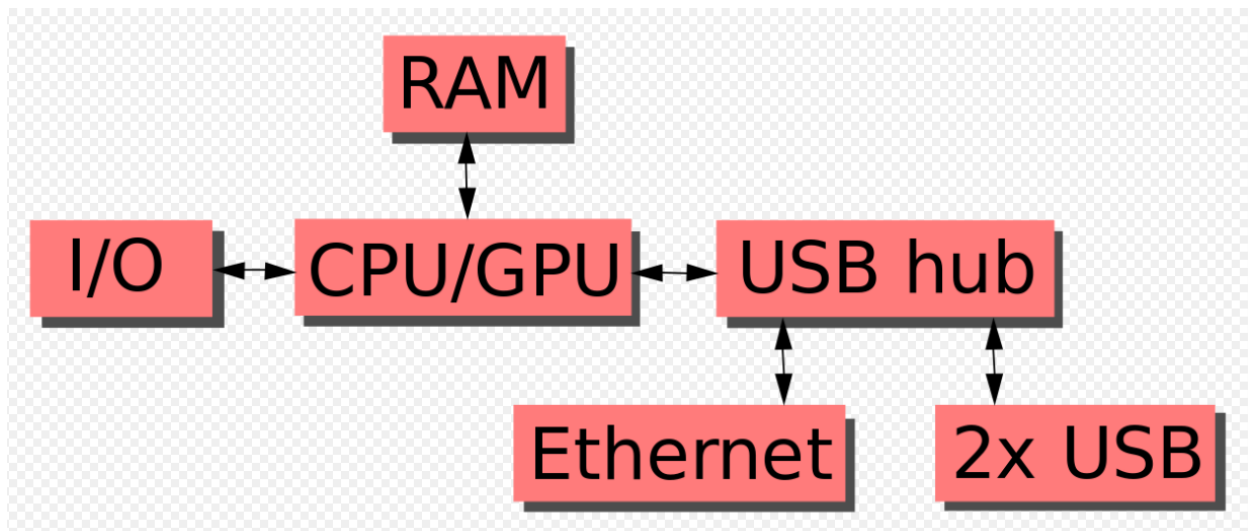
The **Raspberry Pi** is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote teaching of basic computer science in schools and in developing countries. The original model became far more popular than anticipated, selling outside its target market for uses such as robotics. It does not include peripherals (such as keyboards and mice) and cases. However, some accessories have been included in several official and unofficial bundles.

The organization behind the Raspberry Pi consists of two arms. The first two models were developed by the Raspberry Pi Foundation. After the Pi Model B was released, the Foundation set up Raspberry Pi Trading, with Eben Upton as CEO, to develop the third model, the B+. Raspberry Pi Trading is responsible for developing the technology while the Foundation is an educational charity to promote the teaching of basic computer science in schools and in developing countries.

HARDWARE:

The Raspberry Pi hardware has evolved through several versions that feature variations in memory capacity and peripheral-device support.

This block diagram describes Model B and B+; Model A, A+, and the Pi Zero are similar, but lack the Ethernet and USB hub components. The Ethernet adapter is internally connected to an additional USB port. In Model A, A+, and the Pi Zero, the USB port is connected directly to the system on a chip (SoC). On the Pi 1 Model B+ and later models the USB/Ethernet chip contains a five-port USB hub, of which four ports are available, while the Pi 1 Model B only provides two. On the Pi Zero, the USB port is also connected directly to the SoC, but it uses a micro-USB (OTG) port.



4.1.1 Processor

- The Broadcom BCM2835 SoC used in the first generation Raspberry Pi includes a 700 MHz ARM11 76JZF-S processor, Video Core IV graphics processing unit (GPU), and RAM. It has a level 1 (L1) cache of 16 KB and a level 2 (L2) cache of 128 KB.
- The level 2 cache is used primarily by the GPU. The SoC is stacked underneath the RAM chip, so only its edge is visible. The 1176JZ(F)-S is the same CPU used in the original iPhone,^[22] although at a higher clock rate, and mated with a much faster GPU.
- The earlier V1.1 model of the Raspberry Pi 2 used a Broadcom BCM2836 SoC with a 900 MHz 32-bit, quad-core ARM Cortex-A7 processor, with 256 KB shared L2 cache. The Raspberry Pi 2 V1.2 was upgraded to a Broadcom BCM2837 SoC with a 1.2 GHz 64-bit quad-core ARM Cortex-A53 processor, the same SoC which is used on the Raspberry Pi 3, but underclocked (by default) to the same 900 MHz CPU clock speed as the V1.1. The BCM2836 SoC is no longer in production as of late 2016.



- The Raspberry Pi 3+ uses a Broadcom BCM2837B0 SoC with a 1.4 GHz 64-bit quad-core ARM Cortex-A53 processor, with 512 KB shared L2 cache.

4.1.2 Performance

- While operating at 700 MHz by default, the first-generation Raspberry Pi provided a real-world performance roughly equivalent to 0.041 GFLOPS. On the CPU level the performance is similar to a 300 MHz Pentium II of 1997–99. The GPU provides 1 Gpixel/s or 1.5 Gtexel/s of graphics processing or 24 GFLOPS of general purpose computing performance. The graphical capabilities of the Raspberry Pi are roughly equivalent to the performance of the Xbox of 2001.
- Raspberry Pi 2 V1.1 included a quad-core Cortex-A7 CPU running at 900 MHz and 1 GB RAM. It was described as 4–6 times more powerful than its predecessor. The GPU was identical to the original.^[23] In parallelised benchmarks, the Raspberry Pi 2 V1.1 could be up to 14 times faster than a Raspberry Pi 1 Model B+.
- The Raspberry Pi 3, with a quad-core ARM Cortex-A53 processor, is described as having ten times the performance of a Raspberry Pi 1. This was suggested^l to be highly dependent upon task threading and instruction

set use.^l Benchmarks showed the Raspberry Pi 3 to be approximately 80% faster than the Raspberry Pi 2 in parallelised tasks.

4.1.3 Overclocking

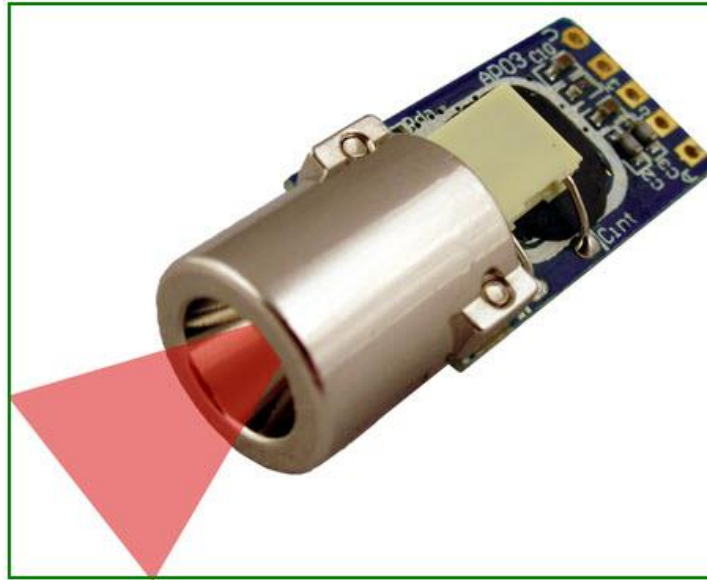
- Most Raspberry Pi systems-on-chip could be overclocked to 800 MHz, and some to 1000 MHz's There are reports the Raspberry Pi 2 can be similarly overclocked, in extreme cases, even to 1500 MHz (discarding all safety features and over-voltage limitations). In the Raspbian Linux distro the overclocking options on boot can be done by a software command running "sudo raspi-config" without voiding the warranty. In those cases the Pi automatically shuts the overclocking down if the chip temperature reaches 85 °C (185 °F), but it is possible to override automatic over-voltage and overclocking settings (voiding the warranty); an appropriately sized heat sink is needed to protect the chip from serious overheating.
- Newer versions of the firmware contain the option to choose between five overclock ("turbo") presets that when used, attempt to maximise the performance of the SoC without impairing the lifetime of the board. This is done by monitoring the core temperature of the chip and the CPU load, and dynamically adjusting clock speeds and the core voltage. When the demand is low on the CPU or it is running too hot the performance is throttled, but if the CPU has much to do and the chip's temperature is acceptable, performance is temporarily increased with clock speeds of up to 1 GHz, depending on the individual board and on which of the turbo settings is used.

4.2 IR SENSOR:

An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measure only infrared radiation, rather than emitting it that is called as a passive IR sensor. Usually in the infrared spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes, that can be detected by an infrared sensor. The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted

by the IR LED. When IR light falls on the photodiode, the resistances and these output voltages, change in proportion to the magnitude of the IR light received.

IR SENSOR

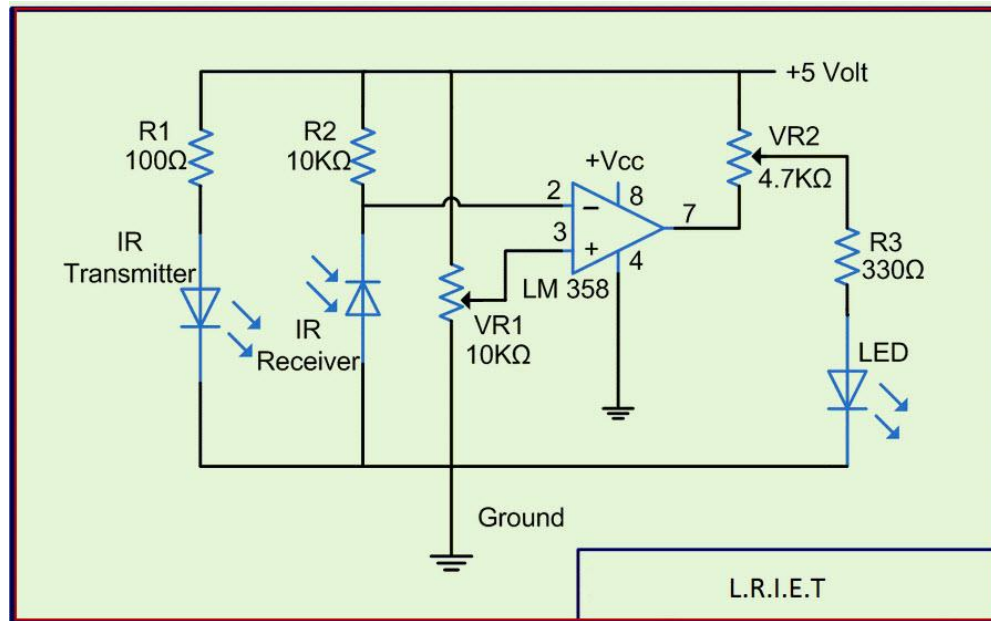


4.2.1 IR Sensor Circuit Diagram and Working Principle

An infrared sensor circuit is one of the basic and popular sensor module in an electronic device. This sensor is analogous to human's visionary senses, which can be used to detect obstacles and it is one of the common applications in real time. This circuit comprises of the following components

- LM358 IC 2 IR transmitter and receiver pair
- Resistors of the range of kilo ohms.
- Variable resistors.
- LED (Light Emitting Diode).

CIRCUIT DIAGRAM



In this project, the transmitter section includes an IR sensor, which transmits continuous IR rays to be received by an IR receiver module. An IR output terminal of the receiver varies depending upon its receiving of IR rays. Since this variation cannot be analyzed as such, therefore this output can be fed to a comparator circuit. Here an operational amplifier (op-amp) of LM 339 is used as comparator circuit. When the IR receiver does not receive a signal, the potential at the inverting input goes higher than that non-inverting input of the comparator IC (LM339). Thus, the output of the comparator goes low, but the LED does not glow. When the IR receiver module receives signal to the potential at the inverting input goes low. Thus, the output of the comparator (LM 339) goes high and the LED starts glowing. Resistor R1 (100), R2 (10k) and R3 (330) are used to ensure that minimum 10 mA current passes through the IR LED Devices like Photodiode and normal LEDs respectively. Resistor VR2 (preset=5k) is used to adjust the output terminals. Resistor VR1 (preset=10k) is used to set the sensitivity of the circuit Diagram. Read more about IR sensors.

4.2.2 Different Types of IR Sensors and Their Applications

IR sensors are classified into different types depending on the applications. Some of the typical applications of different types of sensors are

The speed sensor is used for synchronizing the speed of multiple motors. The temperature sensor is used for industrial temperature control. PIR sensor is

used for automatic door opening system and Ultrasonic sensor are used for distance measurement.

4.2.3 IR Sensor Applications

IR sensors are used in various Sensor based projects and also in various electronic devices which measures the temperature that are discussed in the below.

Radiation Thermometers

IR sensors are used in radiation thermometers to measure the temperature depend upon the temperature and the material of the object and these thermometers have some of the following features

- Measurement without direct contact with the object
- Faster response
- Easy pattern measurements

Flame Monitors

These types of devices are used for detecting the light emitted from the flames and to monitor how the flames are burning. The Light emitted from flames extend from UV to IR region types. PbS, PbSe, Two-color detector, pyro electric detector are some of the commonly employed detector used in flame monitors.

Moisture Analyzers

Moisture analyzers use wavelengths which are absorbed by the moisture in the IR region. Objects are irradiated with light having these wavelengths (1.1 μm , 1.4 μm , 1.9 μm , and 2.7 μm) and also with reference wavelengths. The Lights reflected from the objects depend upon the moisture content and is detected by analyzer to measure moisture (ratio of reflected light at these wavelengths to the reflected light at reference wavelength). In GaAs PIN photodiodes, Pbs photoconductive detectors are employed in moisture analyzer circuits.

Gas Analyzers

IR sensors are used in gas analyzers which use absorption characteristics of gases in the IR region. Two types of methods are used to measure the density of gas such as dispersive and non-dispersive.

GAS ANALYZER:



Dispersive: An Emitted light is spectroscopically divided and their absorption characteristics are used to analyze the gas ingredients and the sample quantity.

Non dispersive: It is most commonly used method and it uses absorption characteristics without dividing the emitted light. Non dispersive types use discrete optical band pass filters, similar to sunglasses that are used for eye protection to filter out unwanted UV radiation.

This type of configuration is commonly referred to as non-dispersive infrared (NDIR) technology. This type of analyzer is used for carbonated drinks, whereas non dispersive analyzer is used in most of the commercial IR instruments, for an automobile exhaust gas fuel leakage.

IR IMAGING DEVICES:

IR image device is one of the major applications of IR waves, primarily by virtue of its property that is not visible. It is used for thermal imagers, night vision devices, etc.



IR IMAGING DEVICES

For examples Water, rocks, soil, vegetation, an atmosphere, and human tissue all features emit IR radiation. The Thermal infrared detectors measure these radiations in IR range and map the spatial temperature distributions of the object/area on an image. Thermal imagers usually composed of a Sb (indium antimonite), Gd Hg (mercury-doped germanium), Hg Cd Te (mercury-cadmium-telluride) sensors.

An electronic detector is cooled to low temperatures using liquid helium or liquid nitrogen. Then the Cooling the detectors ensures that the radiant energy (photons) recorded by the detectors comes from the terrain and not from the ambient temperature of objects within the scanner itself an IR imaging electronic device.

Thus, this is all about IR sensor circuit with working and applications. These sensors are used in many sensors based electronics projects.

4.3 WEB CAMERA

4.3.1 DEFINITION

The term webcam is a combination of "Web" and "video camera." The purpose of a webcam is, not surprisingly, to broadcast video on the Web. Webcams are

typically small cameras that either attach to a user's monitor or sit on a desk. Most webcams connect to the computer via USB, though some use a Firewire connection. Webcams typically come with software that allows the user to record video or stream the video on the Web. If the user has a website that supports streaming video, other users can watch the video stream from their Web browsers.



WEB CAMERA

Webcams can also be used for video chat sessions with other people. Instead of broadcasting the video on the Web, users can set up a video chat session with one or more friends and have a conversation with live audio and video. For example, Apple's iSight camera, which is built into Apple laptops and iMacs, allows users to video chat using the iChat instant messaging program. Several other chat programs also work with webcams, allowing users to set up video chat sessions with friends.

Since streaming video over the Internet requires a lot of bandwidth, the video stream is typically compressed to reduce the "choppiness" of the video. The maximum resolution of a webcam is also lower than most handheld video cameras, since higher resolutions would be reduced anyway. For this reason, webcams are relatively inexpensive compared to most video cameras. And while they may not be ideal for filming a movie, webcams are great for video chat sessions with friends

4.3.2 FUNCTIONS

A webcam is a small camera attached to a computer. They serve a variety of purposes, from taking still images to sending a live video over conference calls. Many laptops have a webcam built in to the computer screen, but external webcams are also sold. If you purchase an external webcam for use, you may also need to get a microphone if you plan on using the camera for either captured or live video.

Still Images

A webcam can function as a digital camera, snapping still images. In most cases, once a webcam is activated, and the image it is pointed at appears on your computer screen, a picture is taken by hitting a specific key on the keyboard or clicking a button on the screen. Once an image is captured, it is saved to the computer hard drive.

Videos

Webcams can also record short video messages. Similar to taking still images, videos are captured by clicking on a designated button on the computer screen. Unlike still images, videos require the act of starting and then stopping the camera. The start and stop button are usually the same, and have a red blinking circle when the camera is recording. A video is saved to the computer's hard drive.

FEATURES

Microphone

Check if the webcam has a built-in microphone. How strong of a mic you require will depend on the type of videoing you'll be doing. Most video chatting (such as Skype) can be done adequately with a webcam's built-in mic. Webcam microphones have advanced quite a bit in recent years. Check for features like dual microphones and omni-directional mics that can record audio from every direction.

Seriously high-quality recording for webisodes or other higher-tech films will still require an upgrade. For those situations, you may want to invest in an external microphone.

Video Effects

Do you like to use avatars or special backgrounds while recording? Some models come with software that allows you to get silly with your filmmaking.

Lens

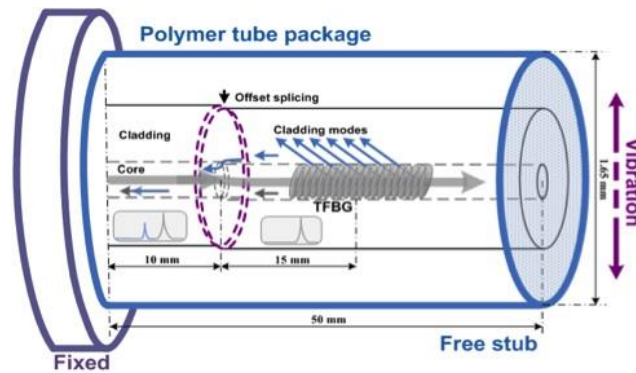
A high-end webcam will have a glass lens while a more moderately priced model will have a plastic lens. As with microphones, whether this difference matters will depend on the type of recording you'll be doing. Most plastic lenses are perfectly adequate for Skype and other video chat software. For more professional video production, consider investing in glass.

4.4 VIBRATION SENSOR:

Vibration sensors are sensors for measuring, displaying, and analyzing linear velocity, displacement and proximity, or acceleration. Vibration — however subtle and unnoticed by human senses — is a telltale sign of machine condition. Abnormal vibration indicative of problems with an industrial machine can be detected early and repaired before the event of machine failure; because such a failure is potentially costly in terms of time, cost, and productivity, vibration measurement allows industrial plants to increase efficiency and save money. Therefore, vibration analysis is used as a tool to determine equipment condition as well as the specific location and type of problems.

The severity of machine vibration is standardized by the International Standards Organization (ISO) in the publication. The standard describes acceptable vibration levels for four different classes of machines. This data, as laid out in the chart below, proves useful as a reference point when analyzing vibration measurements.

- **Vibration sensors** are simply packaged as a raw sensor.
- **Vibration transducers** are slightly more complex sensors which output a voltage or current signal.
- **Vibration transmitters** are sensors packaged with the means (such as integral signal conditioners) to transmit a more complex output.
- **Vibration switches** use an integral sensor to make or break contact when certain vibration levels are detected.

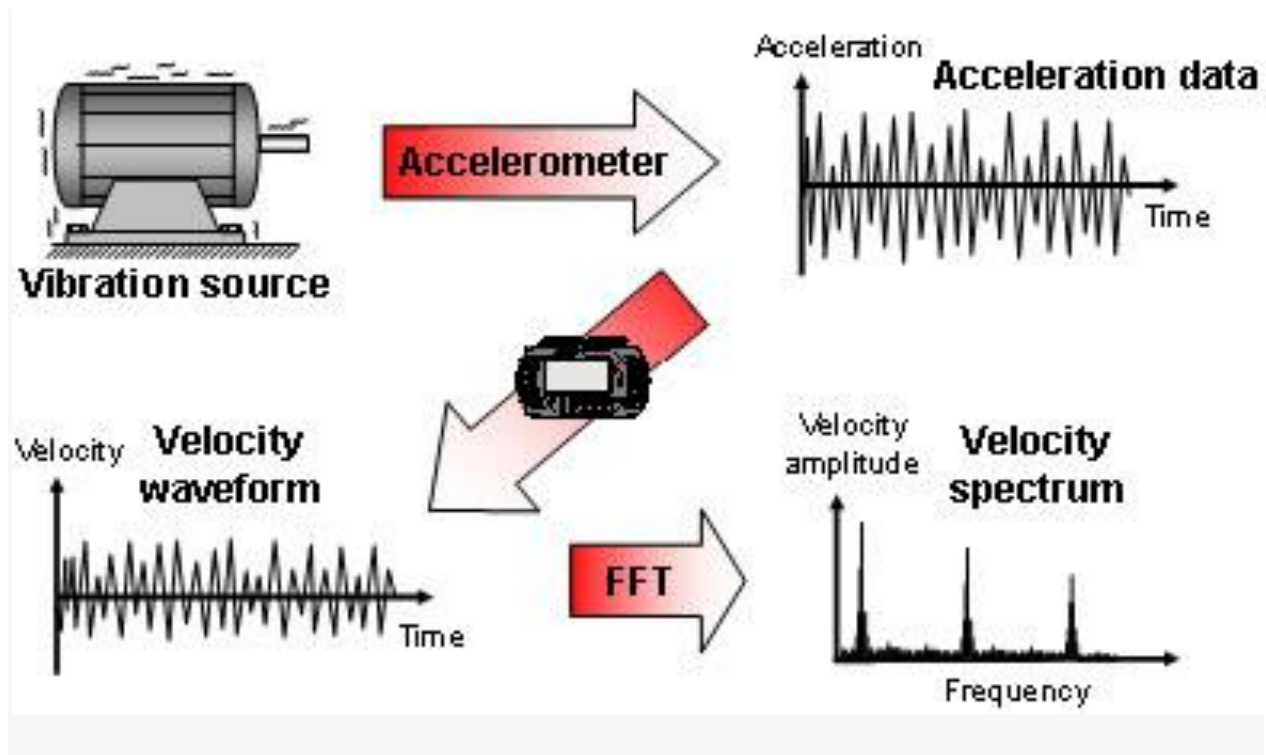


Before taking a vibration measurement, you need to attach a sensor that can detect vibration behavior to the machine that is being measured. Various types of vibration sensors are available, but a type called accelerometer is normally used as it offers advantages over other sensors. An accelerometer is a sensor that produces an electrical signal that is proportional to the acceleration of the vibrating component to which the accelerometer is attached.

What is the acceleration of a vibrating component? It is a measure of how quickly the velocity of the component is changing.

The acceleration signal produced by the accelerometer is passed on to the instrument that in turn converts the signal to a velocity signal. Depending on the user's choice, the signal can be displayed as either a velocity waveform or a velocity spectrum. A velocity spectrum is derived from a velocity waveform by means of a mathematical calculation known as the Fast Fourier Transform or FFT.

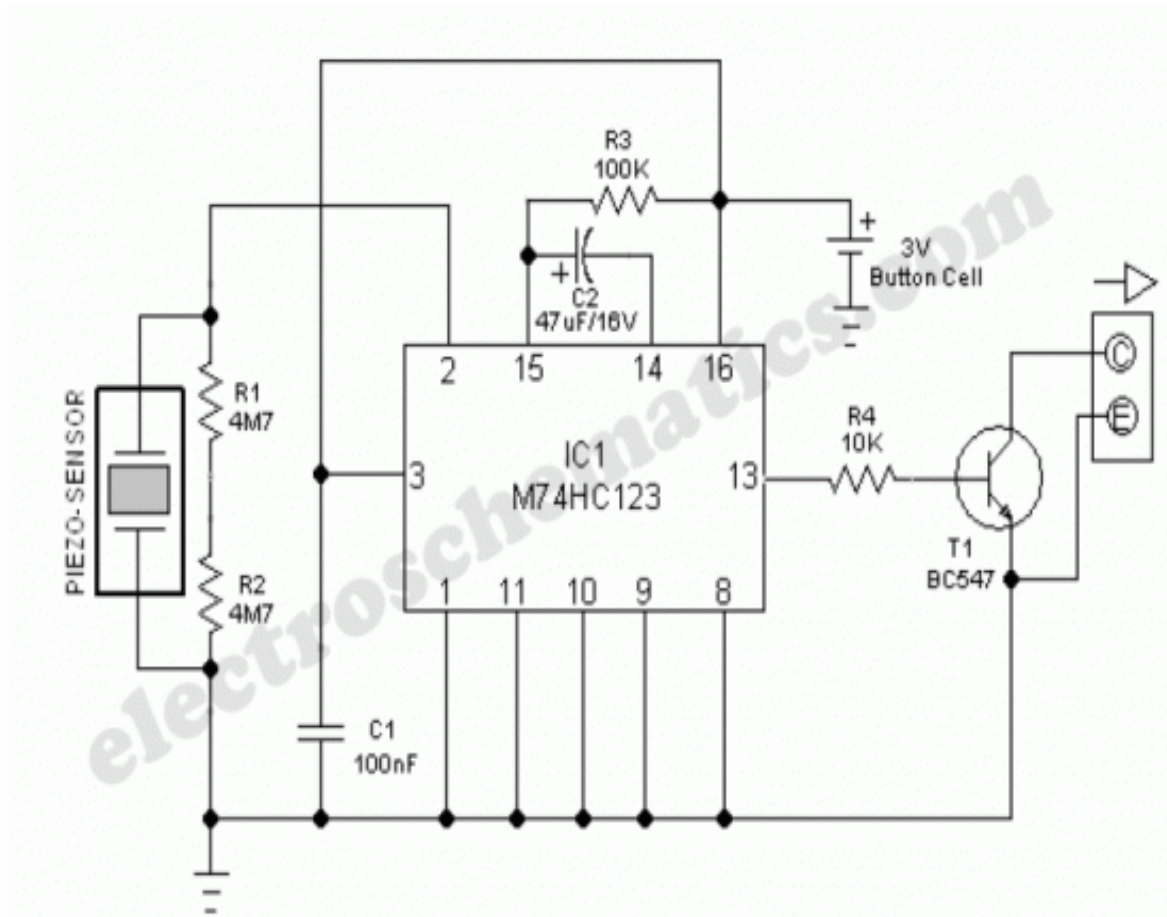
The diagram below is a very simplistic explanation of how vibration data is acquired. You may wish to consult other literature such as the vb-Series Instrument Reference Guide for further information.



With the help of a simple ceramic piezo-electric detector it is possible to assemble an interesting and useful Impact sensor unit, which can be used to detect impact and vibration on doors, showcases, windows etc. The shock sensor (Ceramic piezo-electric detector) uses a “UNIMORPH” diaphragm, which consists of a piezo-electric ceramic disk laminated to a metal disk. The sensor supplies a voltage proportional to the acceleration of the impact or vibration, for example 40mV/G ie output is near 2V for 60G impact.

Here a low voltage, low current Impact sensor unit is realized using a standard ceramic piezo-electric detector which drives a monostable multivibrator (IC1) circuit to activate a npn silicon transistor (T1). Open collector output of this transistor switch can be interfaced to an external alarm/switch circuit for further processing. Since current consumption of the circuit is very low (from 5 to 6 mA only) any common 3V button cell can be used to power the sensor unit. When an impact is sensed, the monostable drives the transistor switch to ON, for a finite duration determined by the in-circuit values of RC timing components R3 and C2.

Vibration/Impact Sensor Circuit Schematic



The M74HC123 (IC1) is a high-speed DUAL retriggerable CMOS MONOSTABLE MULTIVIBRATOR (MMV) fabricated with silicon gate CMOS technology, with all inputs protected against static discharge and transient excess voltage. There are two trigger inputs, negative edge and positive edge. Here, only one monostable part with positive edge triggering (pin 2) is used. After triggering, the output maintains the monostable state for the time period determined by the external resistor R3 and capacitor C2.

4.5 MEMS SENSOR

Micro-Electro-Mechanical Systems, or MEMS, is a technology that in its most general form can be defined as miniaturized mechanical and electro-mechanical elements (i.e., devices and structures) that are made using the techniques of

microfabrication. The critical physical dimensions of MEMS devices can vary from well below one micron on the lower end of the dimensional spectrum, all the way to several millimeters. Likewise, the types of MEMS devices can vary from relatively simple structures having no moving elements, to extremely complex electromechanical systems with multiple moving elements under the control of integrated microelectronics. The one main criterion of MEMS is that there are at least some elements having some sort of mechanical functionality whether or not these elements can move. The term used to define MEMS varies in different parts of the world. In the United States they are predominantly called MEMS, while in some other parts of the world they are called “Microsystems Technology” or “micromachined devices”.

While the functional elements of MEMS are miniaturized structures, sensors, actuators, and microelectronics, the most notable (and perhaps most interesting) elements are the microsensors and micro actuators. Microsensors and micro actuators are appropriately categorized as “transducers”, which are defined as devices that convert energy from one form to another. In the case of microsensors, the device typically converts a measured mechanical signal into an electrical signal.

WORKING

MEMS inclinometers and accelerometers are low-cost, high precision inertial sensors that serve a wide variety of industrial applications. That’s all well and good, but what’s in a MEMS sensor anyway? How can a MEMS sensor serve you? What advantages does it have? These are questions that can be answered here.

It is a chip-based technology, known as a Micro Electro-Mechanical System, that is composed of a suspended mass between a pair of capacitive plates.

When tilt is applied to the sensor, the suspended mass creates a difference in electric potential which is measured as a change in capacitance.

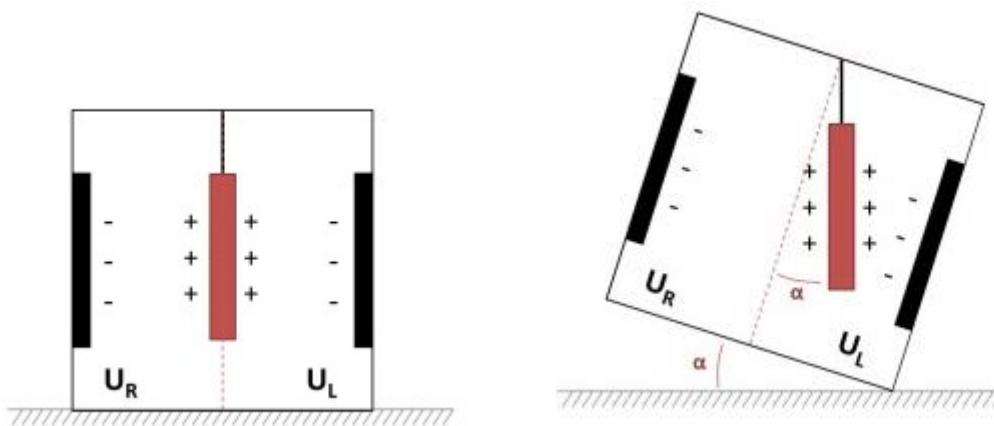
That signal is then amplified to produce a stable output signal in digital, 4-20mA or VDC.

These sensors are fine solutions to applications that do not demand the highest accuracy such as industrial automation, platform leveling, position control and

pitch and roll measurement. Since they are low cost, you can save some dough on your next big project.

The greatest resolution you can get with a MEMS inclinometer is 0.0001° (JDI series). That's a different story from the highly accurate electrolytic line whose most sensitive tiltmeter (A603 tiltmeter) boasts a 2.5 nanoradian resolution. However, MEMS has some advantages of its own.

A MEMS sensor provides the convenient features that you can get with any other sensor line such as analog voltage, current and digital output options. You don't need to concern yourself with space constraints as MEMS utilizes very compact micro machine components so that each sensor can fit into the palm of your hand. They have an IP67 seal and since the operating temperature range is -40° to $+85^\circ\text{C}$, they will withstand some intense conditions. While electrolytic sensors have much higher accuracy, some of them can be sensitive to temperature.



ME

MS Sensor types

Gyro sensors and Accelerometers: These type of MEMS sensors are used for calculation of angular motion or displacement. The main principle behind these sensors is measurement of angular movement with respect to time (i.e. X-Y-Z axial measurements). These sensors also act as temperature sensors. These sensors are used in many different applications such as digital compass, as gravity sensor, GPS tracking etc.

Humidity Sensors: These type of MEMS sensors are widely used in mines, power plants, dockyards and green houses. Humidity sensors work on the principle of absorption of gas in surrounding air. This will bend microscopic cantilever which will cause change in the doped resistance in the absence of gravity. This is sensed

electrically in order to utilize the functionality. Refer humidity sensor for definition of humidity and various types of humidity sensors which include capacitive sensor, electrical conductivity sensor, thermal conductivity sensor, optical hygrometer, oscillating hygrometer etc.

Force Sensors: Refer force sensor which covers spring loaded force sensor with LVDT, force sensor with pressure transducer and strain gauge.

Pressure Sensors: Refer pressure sensor for basic functions and types of pressure sensors e.g. Mercury pressure sensor, Piezoresistive pressure sensor, Capacitive pressure sensor, Optoelectronic pressure sensor etc.

4.6 BUZZER

A **buzzer** or **beeper** is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.



TYPES

Electromechanical

Early devices were based on an electromechanical system identical to an electric bell without the metal gong. Similarly, a relay may be connected to interrupt its

own actuating current, causing the contacts to buzz. Often these units were anchored to a wall or ceiling to use it as a sounding board. The word "buzzer" comes from the rasping noise that electromechanical buzzers made.

Mechanical

A joy buzzer is an example of a purely mechanical buzzer and they require drivers. Other examples of them are doorbells.



Piezoelectric

A piezoelectric element may be driven by an oscillating electronic circuit or other audio signal source, driven with a piezoelectric audio amplifier. Sounds commonly used to indicate that a button has been pressed are a click, a ring or a beep.

A piezoelectric buzzer/beeper also depends on acoustic cavity resonance or Helmholtz resonance to produce an audible beep

APPLICATIONS

While technological advancements have caused buzzers to be impractical and undesirable, there are still instances in which buzzers and similar circuits may be used. Present day applications include:

- Novelty uses
- Judging panels

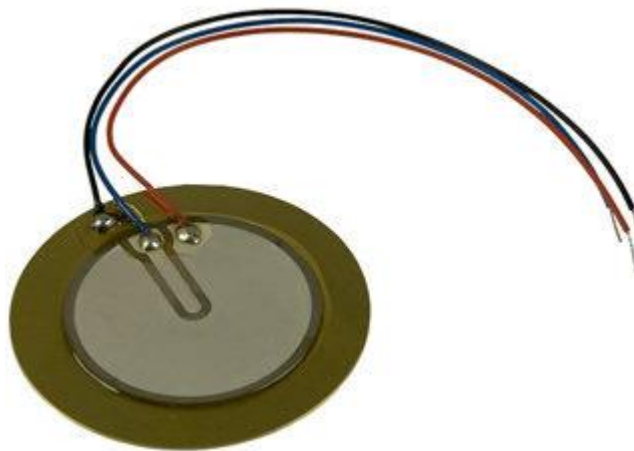
- Educational purposes
- Annunciator panels
- Electronic metronomes
- Game show lock-out device
- Microwave ovens and other household appliances
- Sporting events such as basketball games
- Electrical alarms
- Joy buzzer (mechanical buzzer used for pranks)
-

Simple Piezo Buzzer circuit diagram

Piezo buzzers are one of the most common buzzers available around, they got their name from the piezoelectric material used as the active element.

These buzzers are usually driven at a relatively higher voltage but low current, consumes a little power, but still capable of producing very high sound.

The Piezo element must be a three terminal one, like in the picture.

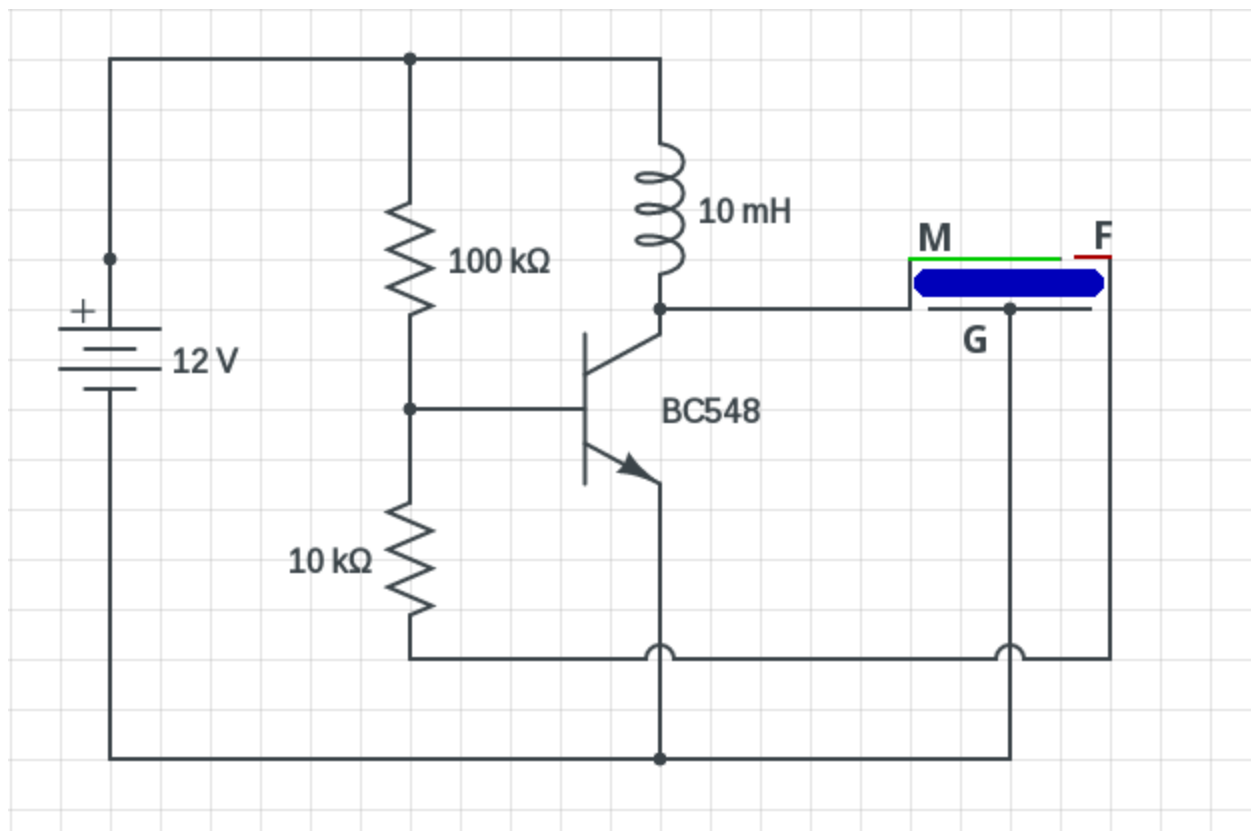


The blue wire is connected to feedback(**F**) terminal, red wire to the main (**M**) terminal and the black wire to the piezo element's ground(**G**) plate.

The inductor coil's value and shape is not crucial. You can use any coil from **1mH** to **10mH** or more, or even no measured value at all. I used a 40 turn coil on a small ferrite toroid in the final design.

Circuit diagram and construction

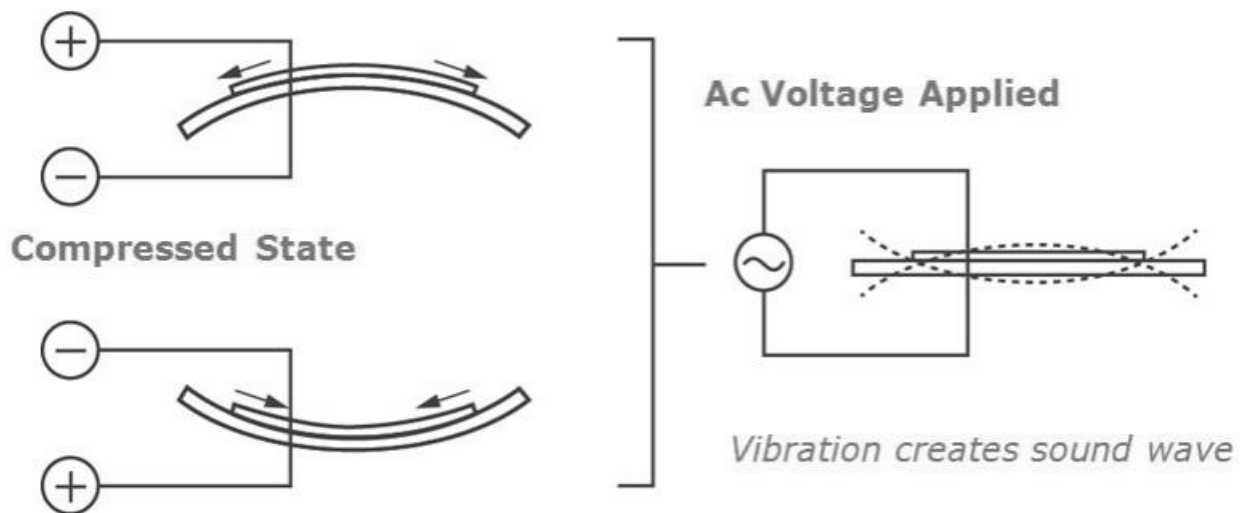
Let's have a look at the circuit diagram,



Note the piezoelectric element's pinout, **M** is the main terminal, **F** is the feedback terminal and **G** is the ground plate.

The circuit is fairly simple, you can use a little piece of strip board to make it. As this piezo buzzer circuit uses very few components, so it also could be constructed by soldering the components to each other.

When A voltage is applied to the electrodes of the piezo element, they produce flex in either way. This flex force bends the ground plate up and down.



The exact opposite thing happens too, when a a piezoelectric element is subjected to varying pressure, it produces voltage.

As you've seen before, self-drive piezo buzzers are constructed with an extra electrically isolated feedback electrode. The voltage created by the flex force is available in the feedback terminal.

The piezo buzzer is placed in a resonant cavity, there is a hole in the opposite side of the resonant cavity from where the buzzing sound comes out.

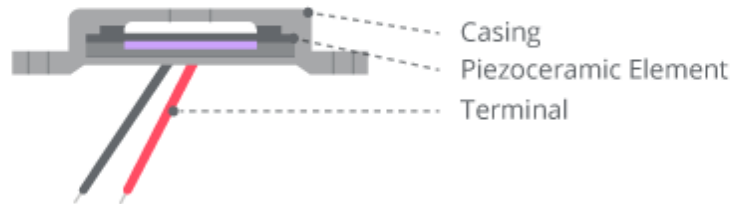
The driver circuit and piezo buzzer co-operates soon between themselves and they start oscillating on the resonant frequency of the piezo buzzer.

Piezo Buzzer Characteristics

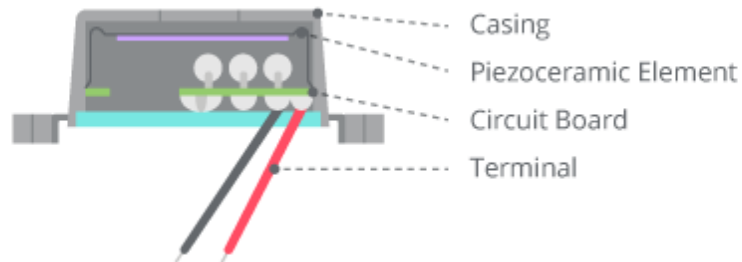
- Wide operating voltage: 3~250 V
- Lower current consumption: less than 30 mA higher rated frequency
- Larger footprint
- Higher sound pressure level

STRUCTURE OF PIEZO BUZZER

Transducer *(without driving circuitry)*



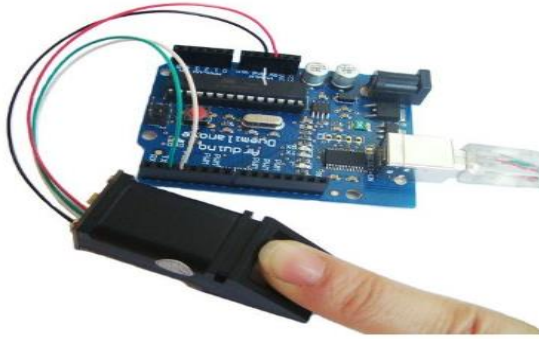
Indicator *(with driving circuitry)*



There are two types of piezo buzzers - transducers and indicators. Transducers consist of a casing, a piezoceramic element and a terminal. In order to operate a transducer, the user must send a square wave signal to the buzzer. Indicators consist of a casing, a piezoceramic element, a circuit board and a terminal. In order to operate an indicator, the user must send the buzzer a specified dc voltage.

4.7 FINGER PRINT SENSOR:

Fingerprint scanners are security systems of biometrics. They are used to unlock doors and in other security applications. During the 2010s fingerprint scanners became commonplace on mobile phones.



TYPES

Optical reflexive

They are based on the oldest technique, that consists of placing your finger on a glass surface or a prism that is illuminated by a led diode. When the ridges of fingerprints touch the surface, the light is absorbed, while between these crests a total reflection occurs. The resulting light and dark areas are recorded in an image sensor.

In practice there are some difficulties with this technique: the images obtained with wet and dry fingers are very different and, in addition, the system is sensitive to dust and dirt on the surface. The unit has a considerable size, impractical and expensive. This system is easy to cheat and if the skin is damaged, the fingerprint is not recognized correctly. The recognition of the fingerprint of older people is also difficult to do because the skin is not elastic enough. In some circumstances this may result in false recognition. If the stored fingerprint was taken with less pressure, false acceptances may occur.

Capacitive

The sensor is a silicon integrated circuit with a surface that is covered by a large number of transducer elements (or pixels), with a typical resolution of 500 dpi. Each element contains two adjacent metal electrodes. The capacity between the electrodes, which forms a feedback path for an inverting amplifier, is reduced when the finger is applied to said surface: it is further reduced when it detects ridges and less reduced when it detects the space between them.

The sensor is susceptible to electrostatic discharge. These sensors only work with normal healthy skins, they are not operative being used on skins with hard areas, calluses or scars. Moisture, grease or dust can also affect its operation.

Mechanical

It consists of tens of thousands of tiny pressure transducers that are mounted on the surface of the sensor. An alternative design uses switches that are closed when pressed by a ridge, but that stay open when they are under a valley. This only provides one bit of information per pixel, instead of working with a gray scale.

Thermal

In this case the heat driven by the finger is detected, which is greater when there is a ridge than when there is a valley. A silicon component has been developed with a matrix of pixels called "Finger Chip", that is, "finger integrated circuit", each of which is covered with a layer of pyroelectric material in which a change in temperature translates into a change in the load distribution of its surface. The image is in the gray scale that has the right quality even with the finger worn, with dirt, with grease or with humidity. The sensor has a robust protective layer and can provide a dynamic output.

Dynamic output

Most of the sensors described have been altered in the past. To avoid this, a new operating mode has been added. Instead of simply placing your finger statically on the sensor, the finger moves slowly along it. The sensor only has a narrow sensitive area, and generates a complete sequence of images, which can be reassembled, by means of a processor, in a complete image. The benefits are improved appreciably and the elimination of any residual grease is guaranteed.

Uses

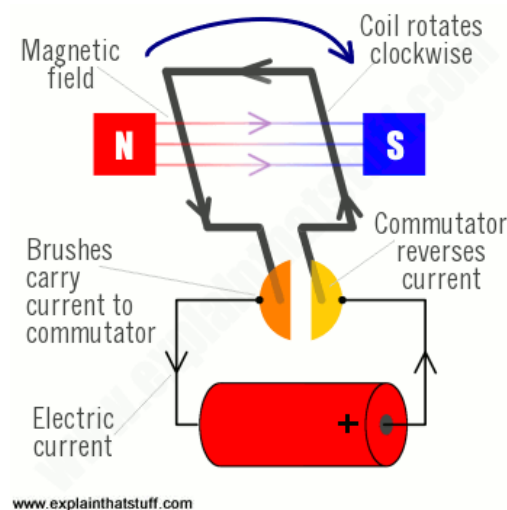
The most important use area is in access control for computers. This is especially important for laptops and PDAs. Thanks to the price falling, more and more devices are equipped with sensors. Other devices with built-in fingerprint sensors include USB hard drives, USB memory modules and card readers. They are also available in mice and keyboards.

Sensors are increasingly used to secure financial transactions and change machines for "online" banking. In the future, the owner's fingerprint will be stored securely on identity cards and credit cards and may also be used for authentication of emails using digital signatures.

Direct physical access to rooms and devices can also be ensured by coupling fingerprint sensors with door opening systems.

4.8 MOTOR:

Motor is a device that produce mechanical output. The working of motor is given as follows: One is to use a kind of electric current that periodically reverses direction, which is known as an **alternating current (AC)**. In the kind of small, battery-powered motors we use around the home, a better solution is to add a component called a **commutator** to the ends of the coil. (Don't worry about the meaningless technical name: this slightly old-fashioned word "commutation" is a bit like the word "commute". It simply means to change back and forth in the same way that commute means to travel back and forth.) In its simplest form, the commutator is a metal ring divided into two separate halves and its job is to reverse the electric current in the coil each time the coil rotates through half a turn. One end of the coil is attached to each half of the commutator. The electric current from the battery connects to the motor's electric **terminals**.



These feed electric power into the commutator through a pair of loose connectors called **brushes**, made either from pieces of graphite (soft carbon similar to pencil "lead") or thin lengths of springy metal, which (as the name suggests) "brush" against the commutator. With the commutator in place, when electricity flows through the circuit, the coil will rotate continually in the same direction.

Components



Rotor

In an electric motor, the moving part is the rotor, which turns the shaft to deliver the mechanical power. The rotor usually has conductors laid into it that carry currents, which interact with the magnetic field of the stator to generate the forces that turn the shaft. Alternatively, some rotors carry permanent magnets, and the stator holds the conductors.

Bearings

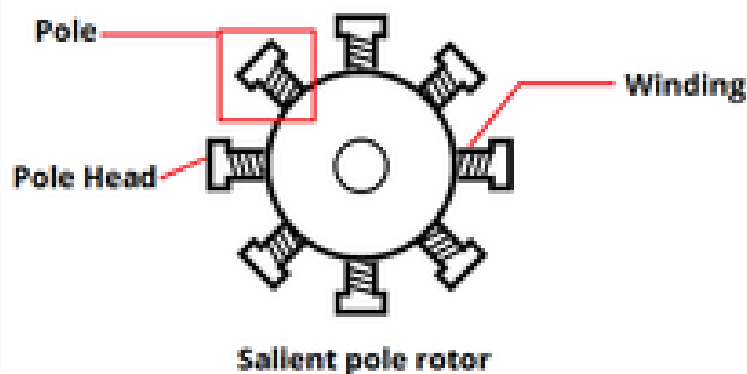
The rotor is supported by bearings, which allow the rotor to turn on its axis. The bearings are in turn supported by the motor housing. The motor shaft extends through the bearings to the outside of the motor, where the load is applied. Because the forces of the load are exerted beyond the outermost bearing, the load.

Stator

The stator is the stationary part of the motor's electromagnetic circuit and usually consists of either windings or permanent magnets. The stator core is made up of many thin metal sheets, called laminations. Laminations are used to reduce energy losses that would result if a solid core were used.

Air gap

The distance between the rotor and stator is called the air gap. The air gap has important effects, and is generally as small as possible, as a large gap has a strong negative effect on performance. It is the main source of the low power factor at which motors operate. The magnetizing current increases with the air gap. For this reason, the air gap should be minimal. Very small gaps may pose mechanical problems in addition to noise and losses.



Salient-pole rotor

Windings

Windings are wires that are laid in coils, usually wrapped around a laminated soft iron magnetic core so as to form magnetic poles when energized with current.

Electric machines come in two basic magnet field pole configurations: salient- and nonsalient-pole configurations. In the salient-pole machine the pole's magnetic field is produced by a winding wound around the pole below the pole face. In the nonsalient-pole, or distributed field, or round-rotor, machine, the winding is distributed in pole face slots.^[60] A shaded-pole motor has a winding around part of the pole that delays the phase of the magnetic field for that pole.

Some motors have conductors that consist of thicker metal, such as bars or sheets of metal, usually copper, alternatively aluminum. These are usually powered by electromagnetic induction.

Commutator



A toy's small DC motor with its commutator

A commutator is a mechanism used to switch the input of most DC machines and certain AC machines. It consists of slip-ring segments insulated from each other and from the shaft. The motor's armature current is supplied through stationary brushes in contact with the revolving commutator, which causes required current reversal, and applies power to the machine in an optimal manner as the rotor rotates from pole to pole.^{[61][62]} In absence of such current reversal, the motor would brake to a stop. In light of improved technologies in the electronic-controller, sensorless-control, induction-motor, and permanent-magnet-motor fields, externally-commutated induction and permanent-magnet motors are displacing electromechanically-commutated motors.

Motor supply and control

Motor supply

A DC motor is usually supplied through slip ring commutator as described above. AC motors' commutation can be either slip ring commutator or externally commutated type, can be fixed-speed or variable-speed control type, and can be synchronous or asynchronous type. Universal motors can run on either AC or DC.

Motor control

Fixed-speed controlled AC motors are provided with direct-on-line or soft-start starters.

Variable-speed controlled AC motors are provided with a range of different power inverter, variable-frequency drive or electronic commutator technologies.

The term electronic commutator is usually associated with self-commutated brushless DC motor and switched reluctance motor applications.

4.9 GSM

GSM is a mobile communication modem; it stands for global system for mobile communication (GSM). The idea of GSM was developed at Bell Laboratories in 1970. It is widely used mobile communication system in the world. GSM is an open and digital cellular technology used for transmitting mobile voice and data services operates at the 850MHz, 900MHz, 1800MHz and 1900MHz frequency bands.

GSM system was developed as a digital system using time division multiple access (TDMA) technique for communication purpose. A GSM digitizes and reduces the data, then sends it down through a channel with two different streams of client data, each in its own particular time slot. The digital system has an ability to carry 64 kbps to 120 Mbps of data rates.



GSM Modem

There are various cell sizes in a GSM system such as macro, micro, pico and umbrella cells. Each cell varies as per the implementation domain. There are five different cell sizes in a GSM network macro, micro, pico and umbrella cells. The coverage area of each cell varies according to the implementation environment.

Time Division Multiple Access

TDMA technique relies on assigning different time slots to each user on the same frequency. It can easily adapt to data transmission and voice communication and can carry 64kbps to 120Mbps of data rate.

GSM Architecture

A GSM network consists of the following components:

- **A Mobile Station:** It is the mobile phone which consists of the transceiver, the display and the processor and is controlled by a SIM card operating over the network.
- **Base Station Subsystem:** It acts as an interface between the mobile station and the network subsystem. It consists of the Base Transceiver Station which contains the radio transceivers and handles the protocols for communication with mobiles. It also consists of the Base Station Controller which controls the Base Transceiver station and acts as a interface between the mobile station and mobile switching centre.
- **Network Subsystem:** It provides the basic network connection to the mobile stations. The basic part of the Network Subsystem is the Mobile Service Switching Centre which provides access to different networks like ISDN, PSTN etc. It also consists of the Home Location Register and the Visitor Location Register which provides the call routing and roaming capabilities of GSM. It also contains the Equipment Identity Register which maintains an account of all the mobile equipments wherein each mobile is identified by its own IMEI number. IMEI stands for International Mobile Equipment Identity.

Features of GSM Module:

- Improved spectrum efficiency
- International roaming
- Compatibility with integrated services digital network (ISDN)
- Support for new services.
- SIM phonebook management
- Fixed dialing number (FDN)
- Real time clock with alarm management
- High-quality speech
- Uses encryption to make phone calls more secure
- Short message service (SMS)

The security strategies standardized for the GSM system make it the most secure telecommunications standard currently accessible. Although the confidentiality of a call and secrecy of the GSM subscriber is just ensured on the radio channel, this is a major step in achieving end-to-end security.

GSM Modem

A GSM modem is a device which can be either a mobile phone or a modem device which can be used to make a computer or any other processor communicate over a network. A GSM modem requires a SIM card to be operated and operates over a network range subscribed by the network operator. It can be connected to a computer through serial, USB or Bluetooth connection.

A GSM modem can also be a standard GSM mobile phone with the appropriate cable and software driver to connect to a serial port or USB port on your computer. GSM modem is usually preferable to a GSM mobile phone. The GSM modem has wide range of applications in transaction terminals, supply chain management, security applications, weather stations and GPRS mode remote data logging.

Working of GSM Module:

From the below circuit, a GSM modem duly interfaced to the MC through the level shifter IC Max232. The SIM card mounted GSM modem upon receiving digit command by SMS from any cell phone send that data to the MC through serial communication. While the program is executed, the GSM modem receives command 'STOP' to develop an output at the MC, the contact point of which are used to disable the ignition switch. The command so sent by the user is based on an intimation received by him through the GSM modem 'ALERT' a programmed message only if the input is driven low. The complete operation is displayed over 16×2 LCD display.

4.10 POWER SUPPLY UNIT:

A power supply (sometimes known as a power supply unit or PSU) is a device or system that supplies electrical or other types of energy to an output load or group of loads. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

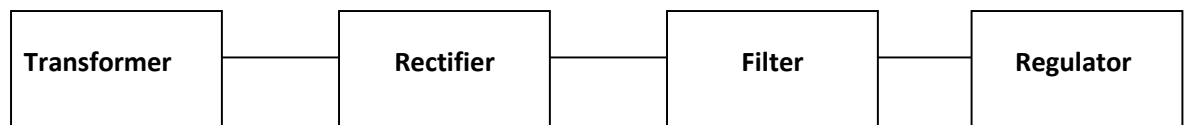


FIG4.1 BLOCK DIAGRAM OF POWER SUPPLY

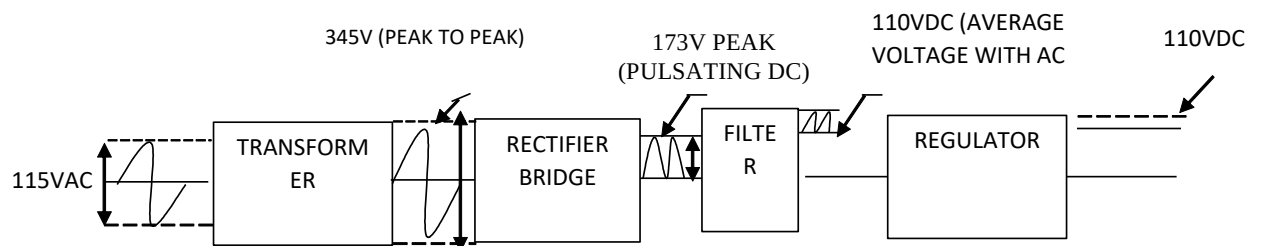


FIG 4.2 PROCESSING OF POWER SUPPLY

The transformer steps up or steps down the input line voltage and isolates the power supply from the power line. The RECTIFIER section converts the alternating current input signal to a pulsating direct current. However, as you proceed in this chapter you will learn that pulsating dc is not desirable. For this reason, a FILTER section is used to convert pulsating dc to a purer, more desirable form of dc voltage.

The final section, the REGULATOR, does just what the name implies. It maintains the output of the power supply at a constant level in spite of large changes in load current or input line voltages. Now that you know what each section does, let's trace an ac signal through the power supply. At this point you need to see how this signal is altered within each section of the power supply. Later on in the chapter you will see

how these changes take place. An input signal of 115 volts ac is applied to the primary of the transformer.

The transformer is a step-up transformer with a turn's ratio of 1:3. You can calculate the output for this transformer by multiplying the input voltage by the ratio of turns in the primary to the ratio of turns in the secondary; therefore, $115 \text{ volts ac} \times 3 = 345 \text{ volts ac}$ (peak-to-peak) at the output. Because each diode in the rectifier section conducts for 180 degrees of the 360-degree input, the output of the rectifier will be one-half, or approximately 173 volts of pulsating dc. The filter section, a network of resistors, capacitors, or inductors, controls the rise and fall time of the varying signal; consequently, the signal remains at a more constant dc level. You will see the filter process more clearly in the discussion of the actual filter circuits. The output of the filter is a signal of 110 volts dc, with ac ripple riding on the dc. The reason for the lower voltage (average voltage) will be explained later in this chapter. The regulator maintains its output at a constant 110-volt dc level, which is used by the electronic equipment (more commonly called the load).

Simple 5V power supply for digital circuits

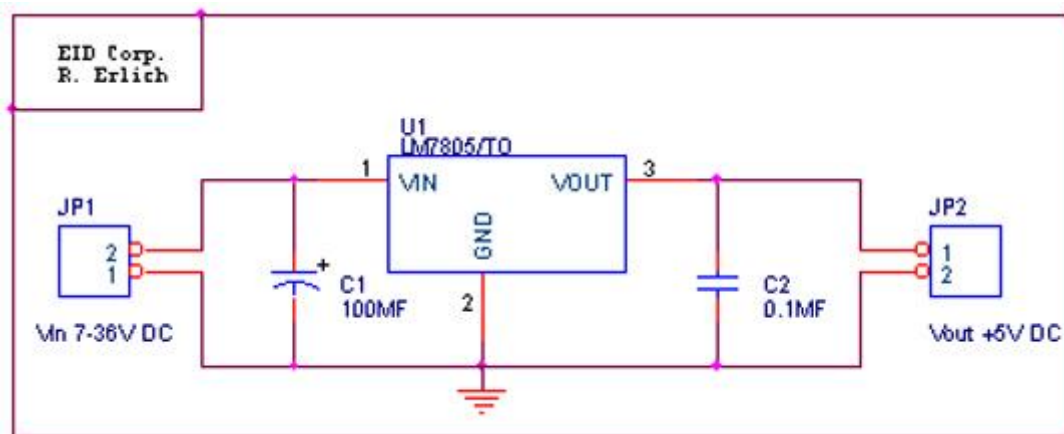
- Brief description of operation: Gives out well-regulated +5V output, output current capability of 100 mA
- Circuit protection: Built-in overheating protection shuts down output when regulator IC gets too hot
- Circuit complexity: Very simple and easy to build
- Circuit performance: Very stable +5V output voltage, reliable operation
- Availability of components: Easy to get, uses only very common basic components
- Design testing: Based on datasheet example circuit, I have used this circuit successfully as part of many electronics projects
- Applications: Part of electronics devices, small laboratory power supply
- Power supply voltage: Unregulated DC 8-18V power supply
- Power supply current: Needed output current + 5 mA

- Component costs: Few dollars for the electronics components + the input transformer cost

Circuit description

This circuit is a small +5V power supply, which is useful when experimenting with digital electronics. Those transformers are easily available, but usually their voltage regulation is very poor, which makes them not very usable for digital circuit experimenter unless a better regulation can be achieved in some way.

This circuit can give +5V output at about 150 mA current, but it can be increased to 1 A when good cooling is added to 7805 regulator chip. The circuit has over overload and thermal protection.



Circuit diagram

Circuit diagram of the power supply:

The capacitors must have enough high voltage rating to safely handle the input voltage feed to circuit. The circuit is very easy to build into a piece of Vero board.

Pinout of the 7805 regulator IC.

1. Unregulated voltage in
2. Ground
3. Regulated voltage out

COMPONENT LIST

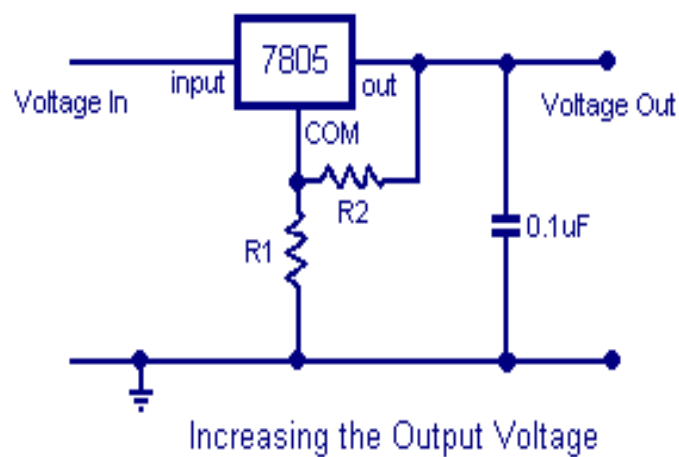
- 7805 regulator IC.
- 100 uF electrolytic capacitor, at least 25V voltage rating.

- 10 μF electrolytic capacitor, at least 6V voltage rating.
- 100 nF ceramic or polyester capacitor.

MORE OUTPUT CURRENT

If we need more than 150 mA of output current, we can update the output current up to 1A doing the following modifications:

- Change the transformer from where we take the power to the circuit to a model which can give as much current as we need from output
- Put a heat sink to the 7805 regulators (so big that it does not overheat because of the extra losses in the regulator)



MORE OUTPUT CURRENT

OTHER OUTPUT VOLTAGES

If we need other voltages than +5V, we can modify the circuit by replacing the 7805 chips with another regulator with different output voltage from regulator 78xx chip family. The last numbers in the chip code tells the output voltage. The input voltage must be at least 3V greater than regulator output voltage to otherwise the regulator does not work well.

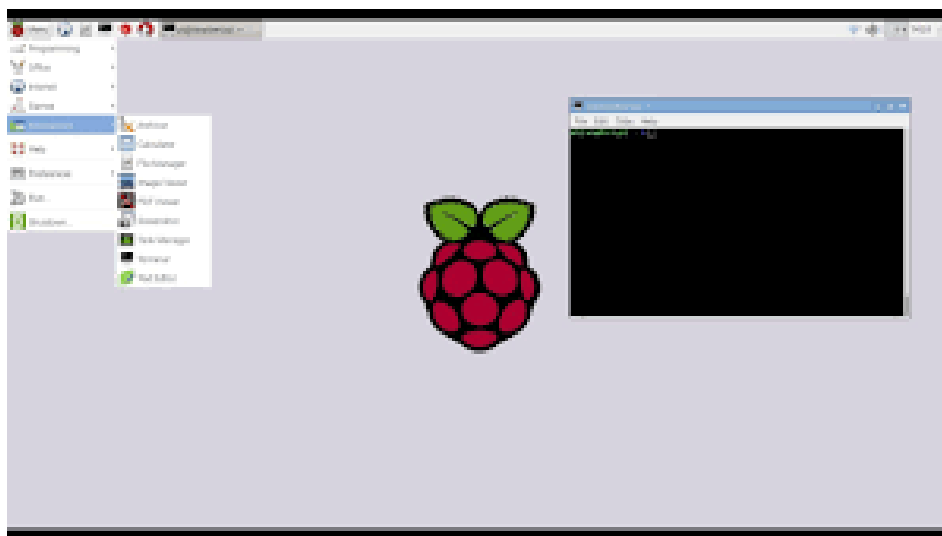
CHAPTER-5

SOFTWARE DESCRIPTIONS

5.1 RASPBIAN OS

Raspberry Pi OS (formerly **Raspbian**) is a Debian-based operating system for Raspberry Pi. Since 2015, it has been officially provided by the Raspberry Pi Foundation as the primary operating system for the Raspberry Pi family of compact single-board computers. The first version of Raspbian was created by Mike Thompson and Peter Green as an independent project.^[5] The initial build was completed in June 2012.

Previous, Raspberry Pi OS versions were 32-bit only and based on Debian, taking the name Raspbian. Since the more recent 64-bit versions no longer use Debian, the name was changed to Raspberry Pi OS for both the 64-bit and 32-bit versions. As of 1 February 2021, the 64-bit version is in beta and is not suitable for general use.



Raspberry Pi OS is highly optimized for the Raspberry Pi line of compact single-board computers with ARM CPUs. It runs on every Raspberry Pi except the Pico microcontroller. Raspberry Pi OS uses a modified LXDE as its desktop environment with the OpenBox stacking window manager, along with a unique theme. The distribution is shipped with a copy of the algebra program Wolfram Mathematica^[4] and a version of Minecraft called Minecraft: Pi Edition, as well as a lightweight version of the Chromium web browser.

5.1.1 FEATURES:

User Interface

Raspberry Pi OS looks similar to many common desktops, such as macOS and Microsoft Windows. The menu bar is positioned at the top and contains an application menu and shortcuts to Terminal, Chromium, and File Manager. On the right is a Bluetooth menu, a Wi-Fi menu, volume control, and a digital clock.

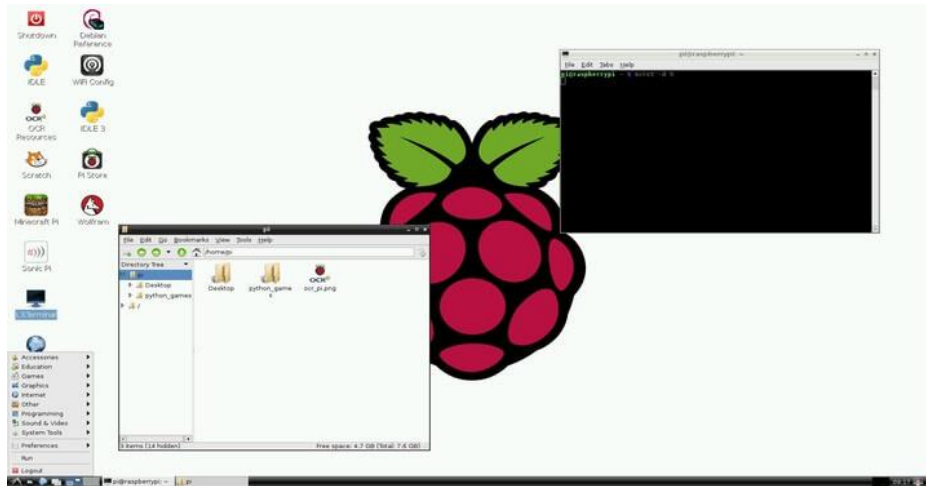
Package Management

Packages can be installed via APT, the Recommended Software app, and by using the Add/Remove Software tool, a GUI wrapper for APT.

Microsoft Repository Controversy

In late January 2021, Raspberry Pi OS' raspberrypi-sys-mods package added a trusted GPG key and sources.list.d entry to APT without user consent. This addition granted Microsoft the ability to install and run any software during the daily critical update process on all Pi that had done a manual apt upgrade to receive the change. The change was not pushed as a critical update and, as of yet, the excessive permission has not been abused by Microsoft and would seem unlikely to ever be abused. The author of the change acknowledged on GitHub that too many rights were granted to Microsoft and also acknowledged delaying the public release of the source code for the change.

In addition to the permissions, the change also causes Pi running an updated Raspberry Pi OS to contact packages.microsoft.com daily and thereby reveal their IP address as a Raspberry Pi OS user for potential use in tracking or marketing efforts. On 8 February 2021, the original author made another change that restricted Microsoft's ability to install software to packages beginning with the string "code" but Microsoft can still run code as root so this restriction is trivial to bypass. As of 8 February 2021, the issue is not resolved and the Raspberry Pi Foundation has locked or deleted many of the related threads on their public forum and their GitHub pages but has acknowledged there is a problem to be resolved and that they are working on it.



Components

PCManFM is a file browser allowing quick access to all areas of the computer, and was redesigned in the first Raspberry Pi OS Buster release (2019-06-20).

Raspberry Pi OS originally used Epiphany as the web browser, but switched to Chromium with the launch of its redesigned desktop.^[13]

Raspberry Pi OS comes with many beginner IDEs, such as Thonny Python IDE, Mu Editor, and Greenfoot. It also ships with educational software like Scratch and Bookshelf.

5.2 PYTHON

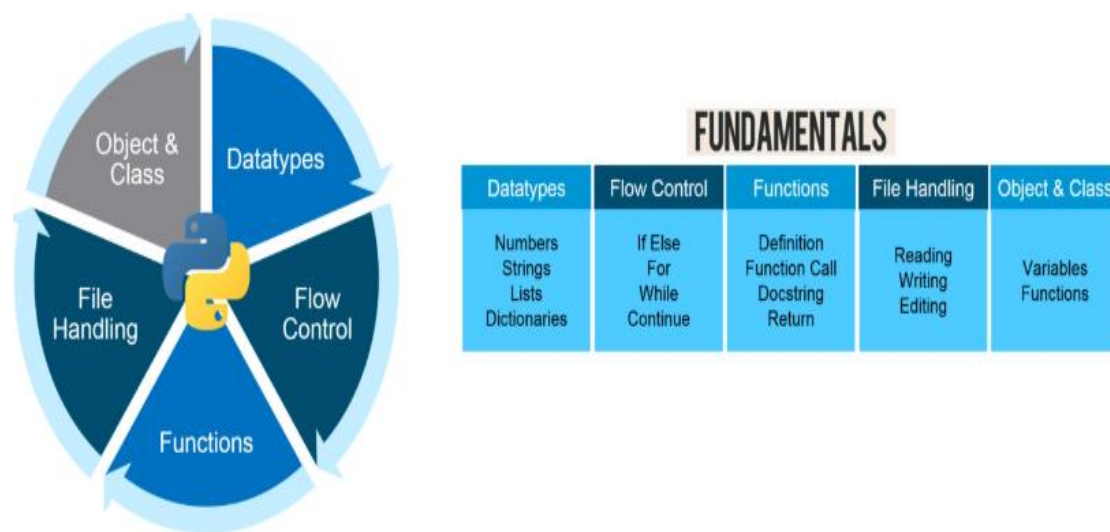
Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. Van Rossum led the language community until stepping down as leader in July 2018.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library

Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by metaprogramming and metaobjects (magic methods)). Many other paradigms are supported via extensions, including design by contract and logic programming.

Python uses dynamic typing, and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

Python's design offers some support for functional programming in the Lisp tradition. It has filter (), map (), and reduce () functions; list comprehensions, dictionaries, and sets; and generator expressions. The standard library has two modules (itertools and functools) that implement functional tools borrowed from Haskell and Standard M.



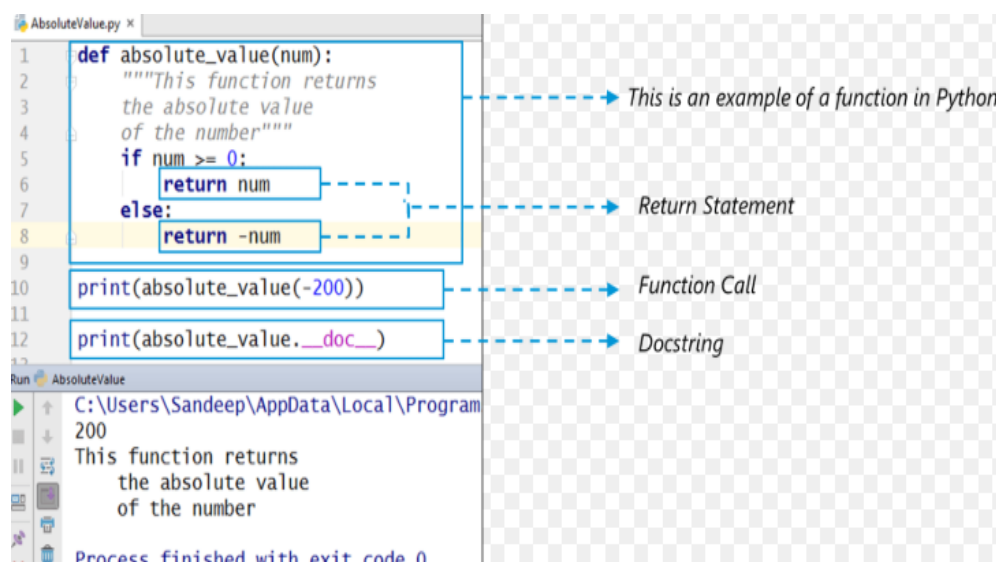
The language's core philosophy is summarized in the document *The Zen of Python* (PEP 20), which includes aphorisms such as:

- Beautiful is better than ugly
- Explicit is better than implicit
- Simple is better than complex
- Complex is better than complicated
- Readability counts

Rather than having all of its functionality built into its core, Python was designed to be highly extensible. This compact modularity has made it

particularly popular as a means of adding programmable interfaces to existing applications. Van Rossum's vision of a small core language with a large standard library and easily extensible interpreter stemmed from his frustrations with ABC, which espoused the opposite approach.

While offering choice in coding methodology, the Python philosophy rejects exuberant syntax (such as that of Perl) in favor of a simpler, less-cluttered grammar. As Alex Martelli put it: "To describe something as 'clever' is *not* considered a compliment in the Python culture." Python's philosophy rejects the Perl "there is more than one way to do it" approach to language design in favor of "there should be one—and preferably only one—obvious way to do it".



Python's developers strive to avoid premature optimization, and reject patches to non-critical parts of the CPython reference implementation that would offer marginal increases in speed at the cost of clarity.^[51] When speed is important, a Python programmer can move time-critical functions to extension modules written in languages such as C, or use PyPy, a just-in-time compiler. Cython is also available, which translates a Python script into C and makes direct C-level API calls into the Python interpreter.

An important goal of Python's developers is keeping it fun to use. This is reflected in the language's name—a tribute to the British comedy group Monty Python—and in occasionally playful approaches to tutorials and reference materials, such as examples that refer to spam and eggs

(from a famous Monty Python sketch) instead of the standard foo and bar.

A common neologism in the Python community is *pythonic*, which can have a wide range of meanings related to program style. To say that code is pythonic is to say that it uses Python idioms well, that it is natural or shows fluency in the language, that it conforms with Python's minimalist philosophy and emphasis on readability. In contrast, code that is difficult to understand or reads like a rough transcription from another programming language is called *unpythonic*.

Users and admirers of Python, especially those considered knowledgeable or experienced, are often referred to as *Pythonists*, *Pythonistas*, and *Pythoneers*

Syntax and semantics

Python is meant to be an easily readable language. Its formatting is visually uncluttered, and it often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use curly brackets to delimit blocks, and semicolons after statements are optional. It has fewer syntactic exceptions and special cases than C or Pascal.

Indentation

Main article: Python syntax and semantics § Indentation

Python uses whitespace indentation, rather than curly brackets or keywords, to delimit blocks. An increase in indentation comes after certain statements; a decrease in indentation signifies the end of the current block. Thus, the program's visual structure accurately represents the program's semantic structure.^[1] This feature is also sometimes termed the off-side rule.

Statements and control flow

Python's statements include (among others):

- The assignment statement (token '=', the equals sign). This operates differently than in traditional imperative programming languages, and

this fundamental mechanism (including the nature of Python's version of *variables*) illuminates many other features of the language. Assignment in C, e.g., `x = 2`, translates to "typed variable name `x` receives a copy of numeric value 2". The (right-hand) value is copied into an allocated storage location for which the (left-hand) variable name is the symbolic address. The memory allocated to the variable is large enough (potentially quite large) for the declared type. In the simplest case of Python assignment, using the same example, `x = 2`, translates to "(generic) name `x` receives a reference to a separate, dynamically allocated object of numeric (int) type of value 2." This is termed *binding* the name to the object. Since the name's storage location doesn't *contain* the indicated value, it is improper to call it a *variable*. Names may be subsequently rebound at any time to objects of greatly varying types, including strings, procedures, complex objects with data and methods, etc. Successive assignments of a common value to multiple names, e.g., `x = 2; y = 2; z = 2` result in allocating storage to (at most) three names and one numeric object, to which all three names are bound. Since a name is a generic reference holder it is unreasonable to associate a fixed data type with it. However, at a given time a name will be bound to *some* object, which **will** have a type; thus, there is dynamic typing.

- The `if` statement, which conditionally executes a block of code, along with `else` and `elif` (a contraction of else-if).
- The `for` statement, which iterates over an iterable object, capturing each element to a local variable for use by the attached block.
- The `while` statement, which executes a block of code as long as its condition is true.
- The `try` statement, which allows exceptions raised in its attached code block to be caught and handled by `except` clauses; it also ensures that clean-up code in a `finally` block will always be run regardless of how the block exits.
- The `raise` statement, used to raise a specified exception or re-raise a caught exception.
- The `class` statement, which executes a block of code and attaches its local namespace to a class, for use in object-oriented programming.
- The `def` statement, which defines a function or method.

- The `with` statement, from Python 2.5 released on September 2006, which encloses a code block within a context manager (for example, acquiring a lock before the block of code is run and releasing the lock afterwards, or opening a file and then closing it), allowing Resource Acquisition Is Initialization (RAII)-like behavior and replaces a common try/finally idiom.
- The `pass` statement, which serves as a NOP. It is syntactically needed to create an empty code block.
- The `assert` statement, used during debugging to check for conditions that ought to apply.
- The `yield` statement, which returns a value from a generator function. From Python 2.5, `yield` is also an operator. This form is used to implement coroutines.
- The `import` statement, which is used to import modules whose functions or variables can be used in the current program. There are three ways of using import: `import <module name> [as <alias>]` or `from <module name> import *` or `from <module name> import <definition 1> [as <alias 1>], <definition 2> [as <alias 2>], ...`
- The `print` statement was changed to the `print()` function in Python 3.

Python does not support tail call optimization or first-class continuations, and, according to Guido van Rossum, it never will. However, better support for coroutine-like functionality is provided in 2.5, by extending Python's generators. Before 2.5, generators were lazy iterators; information was passed unidirectionally out of the generator. From Python 2.5, it is possible to pass information back into a generator function, and from Python 3.3, the information can be passed through multiple stack levels.

Expressions

Some Python expressions are similar to languages such as C and Java, while some are not:

- Addition, subtraction, and multiplication are the same, but the behavior of division differs. There are two types of divisions in Python. They are floor division and integer division. Python also added the `**` operator for exponentiation.

- From Python 3.5, the new `@` infix operator was introduced. It is intended to be used by libraries such as NumPy for matrix multiplication
- In Python, `==` compares by value, versus Java, which compares numeric by value^[69] and objects by reference.^[70] (Value comparisons in Java on objects can be performed with the `equals()` method.) Python's `is` operator may be used to compare object identities (comparison by reference). In Python, comparisons may be chained, for example `a <= b <= c`.
- Python uses the words `and`, `or`, `not` for its boolean operators rather than the symbolic `&&`, `||`, `!` used in Java and C.
- Python has a type of expression termed a *list comprehension*. Python 2.4 extended list comprehensions into a more general expression termed a *generator expression*.
- Anonymous functions are implemented using lambda expressions; however, these are limited in that the body can only be one expression.
- Conditional expressions in Python are written as `x if c else y`^[71] (different in order of operands from the `c ? x : y` operator common to many other languages).
- Python makes a distinction between lists and tuples. Lists are written as `[1, 2, 3]`, are mutable, and cannot be used as the keys of dictionaries (dictionary keys must be immutable in Python). Tuples are written as `(1, 2, 3)`, are immutable and thus can be used as the keys of dictionaries, provided all elements of the tuple are immutable. The `+` operator can be used to concatenate two tuples, which does not directly modify their contents, but rather produces a new tuple containing the elements of both provided tuples. Thus, given the variable `t` initially equal to `(1, 2, 3)`, executing `t = t + (4, 5)` first evaluates `t + (4, 5)`, which yields `(1, 2, 3, 4, 5)`, which is then assigned back to `t`, thereby effectively "modifying the contents" of `t`, while conforming to the immutable nature of tuple objects. Parentheses are optional for tuples in unambiguous contexts.
- Python features *sequence unpacking* where multiple expressions, each evaluating to anything that can be assigned to (a variable, a writable property, etc.), are associated in the identical manner to that forming

tuple literals and, as a whole, are put on the left-hand side of the equal sign in an assignment statement. The statement expects an *iterable* object on the right-hand side of the equal sign that produces the same number of values as the provided writable expressions when iterated through, and will iterate through it, assigning each of the produced values to the corresponding expression on the left.

- Python has a "string format" operator `%`. This functions analogous to `printf` format strings in C, e.g. `"spam=%s eggs=%d" % ("blah", 2)` evaluates to `"spam=blah eggs=2"`. In Python 3 and 2.6+, this was supplemented by the `format()` method of the `str` class, e.g. `"spam={0} eggs={1}".format("blah", 2)`. Python 3.6 added "f-strings": `blah = "blah"; eggs = 2; f'spam={blah} eggs={eggs}'`.
- Python has various kinds of string literals:
 - Strings delimited by single or double quote marks. Unlike in Unix shells, Perl and Perl-influenced languages, single quote marks and double quote marks function identically. Both kinds of string use the backslash (`\`) as an escape character. String interpolation became available in Python 3.6 as "formatted string literals".
 - Triple-quoted strings, which begin and end with a series of three single or double quote marks. They may span multiple lines and function like here documents in shells, Perl and Ruby.
 - Raw string varieties, denoted by prefixing the string literal with an `r`. Escape sequences are not interpreted; hence raw strings are useful where literal backslashes are common, such as regular expressions and Windows-style paths. Compare "`@-quoting`" in C#.
- Python has array index and array slicing expressions on lists, denoted as `a[key]`, `a[start:stop]` or `a[start:stop:step]`. Indexes are zero-based, and negative indexes are relative to the end. Slices take elements from the *start* index up to, but not including, the *stop* index. The third slice parameter, called *step* or *stride*, allows elements to be skipped and reversed. Slice indexes may be omitted, for example `a[:]` returns a copy of the entire list. Each element of a slice is a shallow copy.

In Python, a distinction between expressions and statements is rigidly enforced, in contrast to languages such as Common Lisp, Scheme, or Ruby. This leads to duplicating some functionality. For example:

- List comprehensions vs. `for`-loops
- Conditional expressions vs. `if` blocks
- The `eval()` vs. `exec()` built-in functions (in Python 2, `exec` is a statement); the former is for expressions, the latter is for statements.

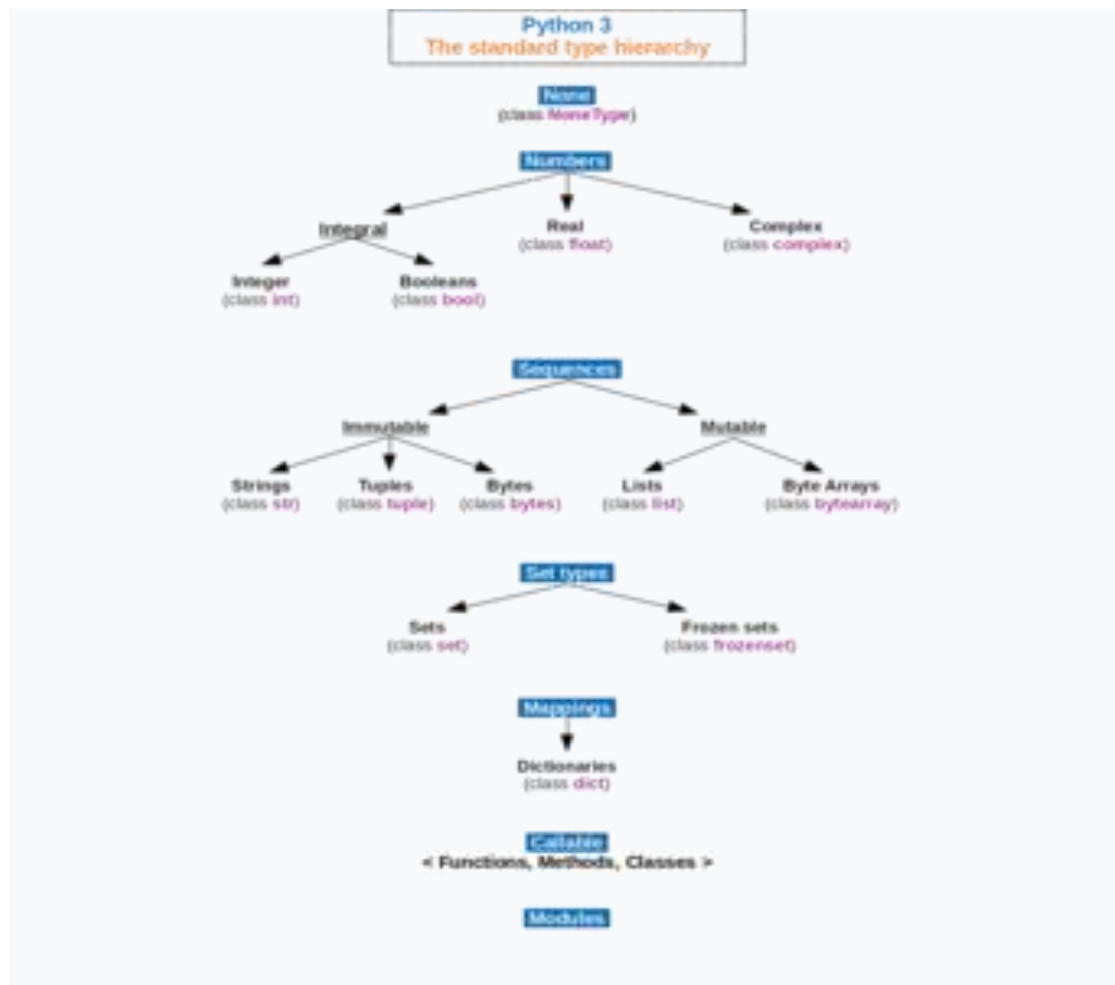
Statements cannot be a part of an expression, so list and other comprehensions or lambda expressions, all being expressions, cannot contain statements. A particular case of this is that an assignment statement such as `a = 1` cannot form part of the conditional expression of a conditional statement. This has the advantage of avoiding a classic C error of mistaking an assignment operator `=` for an equality operator `==` in conditions: `if (c = 1) { ... }` is syntactically valid (but probably unintended) C code but `if c = 1: ...` causes a syntax error in Python.

Methods

Methods on objects are functions attached to the object's class; the syntax `instance.Method(argument)` is, for normal methods and functions, syntactic sugar for `Class.Method(instance, argument)`. Python methods have an explicit `self`-parameter to access instance data, in contrast to the implicit `self` (or `this`) in some other object-oriented programming languages (e.g., C++, Java, Objective-C, or Ruby).

Typing

Python uses duck typing and has typed objects but untyped variable names. Type constraints are not checked at compile time; rather, operations on an object may fail, signifying that the given object is not of a suitable type. Despite being dynamically typed, Python is strongly typed, forbidding operations that are not well-defined (for example, adding a number to a string) rather than silently attempting to make sense of them.



5.3 EXAMPLE OF PYTHON PROGRAMMING

Python allows programmers to define their own types using classes, which are most often used for object-oriented programming. New instances of classes are constructed by calling the class (for example, `SpamClass()` or `EggsClass()`), and the classes are instances of the metaclass type (itself an instance of itself), allowing metaprogramming and reflection.

```

menu=["Bread","Doughnut","Muffine","COOKIE"]
prices=[5,2,3,1]

def setup():
    size(500,500)

def draw():
    line(250,0,250,500)

    text("Menu",100,50)

    text("Bread: $5", 30, 100)
    text("Doughnut: $2", 30, 120)
    text("Muffin: $2", 30, 140)
    text("COOKIE: $2", 30, 160)

    text("Prices",350,50)

```

Before version 3.0, Python had two kinds of classes: *old-style* and *new-style*. The syntax of both styles is the same, the difference being whether the class object is inherited from, directly or indirectly (all new-style classes inherit from object and are instances of type). In versions of Python 2 from Python 2.2 onwards, both kinds of classes can be used. Old-style classes were eliminated in Python 3.0.

The long-term plan is to support gradual typing and from Python 3.5, the syntax of the language allows specifying static types but they are not checked in the default implementation, CPython. An experimental optional static type checker named *mypy* supports compile-time type checking.

5.4PYTHON IS USED BECAUSE:

1)Readable and Maintainable Code

While writing a software application, you must focus on the quality of its source code to simplify maintenance and updates. The syntax rules of Python allow you to express concepts without writing additional code. At the same time, Python, unlike other programming languages, emphasizes

on code readability, and allows you to use English keywords instead of punctuations. Hence, you can use Python to build custom applications without writing additional code. The readable and clean code base will help you to maintain and update the software without putting extra time and effort.

2) Multiple Programming Paradigms

Like other modern programming languages, Python also supports several programming paradigm. It supports object oriented and structured programming fully. Also, its language features support various concepts in functional and aspect-oriented programming. At the same time, Python also features a dynamic type system and automatic memory management. The programming paradigms and language features help you to use Python for developing large and complex software applications.

3) Compatible with Major Platforms and Systems

At present, Python is supporting many operating systems. You can even use Python interpreters to run the code on specific platforms and tools. Also, Python is an interpreted programming language. It allows you to you to run the same code on multiple platforms without recompilation. Hence, you are not required to recompile the code after making any alteration. You can run the modified application code without recompiling and check the impact of changes made to the code immediately. The feature makes it easier for you to make changes to the code without increasing development time.

4) Robust Standard Library

Its large and robust standard library makes Python score over other programming languages. The standard library allows you to choose from a wide range of modules according to your precise needs. Each module further enables you to add functionality to the Python application without writing additional code. For instance, while writing a web application in Python, you can use specific modules to implement web services, perform string operations, manage operating system interface or work with internet protocols. You can even gather information about various modules by browsing through the Python Standard Library documentation.

5) Many Open-Source Frameworks and Tools

As an open-source programming language, Python helps you to curtail software development cost significantly. You can even use several open-source Python frameworks, libraries and development tools to curtail

development time without increasing development cost. You even have option to choose from a wide range of open-source Python frameworks and development tools according to your precise needs. For instance, you can simplify and speedup web application development by using robust Python web frameworks like Django, Flask, Pyramid, Bottle and CherryPy. Likewise, you can accelerate desktop GUI application development using **Python GUI frameworks** and toolkits like PyQt, PyJs, PyGUI, Kivy, PyGTK and WxPython.

6) Simplify Complex Software Development

Python is a general-purpose programming language. Hence, you can use the programming language for developing both desktop and web applications. Also, you can use Python for developing complex scientific and numeric applications. Python is designed with features to facilitate data analysis and visualization. You can take advantage of the data analysis features of Python to create custom big data solutions without putting extra time and effort. At the same time, the data visualization libraries and APIs provided by Python help you to visualize and present data in a more appealing and effective way. Many **Python developers** even use Python to accomplish artificial intelligence (AI) and natural language processing tasks.

7) Adopt Test Driven Development

You can use Python to create prototype of the software application rapidly. Also, you can build the software application directly from the prototype simply by refactoring the Python code. Python even makes it easier for you to perform coding and testing simultaneously by adopting test driven development (TDD) approach. You can easily write the required tests before writing code and use the tests to assess the application code continuously. The tests can also be used for checking if the application meets predefined requirements based on its source code.

However, Python, like other programming languages, has its own shortcomings. It lacks some of the built-in features provided by other modern programming language. Hence, you have to use Python libraries, modules, and frameworks to accelerate custom software development. Also, several studies have shown that Python is slower than several widely used programming languages including Java and C++. You have to speed up the Python application by making changes to the application

code or using custom runtime. But you can always use Python to speed up software development and simplify software maintenance.

5.4.1 Advantages or Benefits of Python

The Python language has diversified application in the software development companies such as in gaming, web frameworks and applications, language development, prototyping, graphic design applications, etc. This provides the language a higher plethora over other programming languages used in the industry. Some of its advantages are-

- **Extensive Support Libraries**

It provides large standard libraries that include the areas like string operations, Internet, web service tools, operating system interfaces and protocols. Most of the highly used programming tasks are already scripted into it that limits the length of the codes to be written in Python.

- **Integration Feature**

Python integrates the Enterprise Application Integration that makes it easy to develop Web services by invoking COM or COBRA components. It has powerful control capabilities as it calls directly through C, C++ or Java via Jython. Python also processes XML and other markup languages as it can run on all modern operating systems through same byte code.

- **Improved Programmer's Productivity**

The language has extensive support libraries and clean object-oriented designs that increase two to tenfold of programmer's productivity while using the languages like Java, VB, Perl, C, C++ and C#.

- **Productivity**

With its strong process integration features, unit testing framework and enhanced control capabilities contribute towards the increased speed for most applications and productivity of applications. It is a great option for building scalable multi-protocol network applications.

Limitations or Disadvantages of Python

Python has varied advantageous features, and programmers prefer this language to other programming languages because it is easy to learn and code too. However, this language has still not made its place in some computing arenas that includes Enterprise Development Shops. Therefore, this language may not solve some of the enterprise solutions, and limitations include-

- **Difficulty in Using Other Languages**

The Python lovers become so accustomed to its features and its extensive libraries, so they face problem in learning or working on other programming languages. Python experts may see the declaring of cast “values” or variable “types”, syntactic requirements of adding curly braces or semi colons as an onerous task.

- **Weak in Mobile Computing**

Python has made its presence on many desktop and server platforms, but it is seen as a weak language for mobile computing. This is the reason very few mobile applications are built in it like Carbonnelle.

- **Gets Slow in Speed**

Python executes with the help of an interpreter instead of the compiler, which causes it to slow down because compilation and execution help it to work normally. On the other hand, it can be seen that it is fast for many web applications too.

- **Run-time Errors**

The Python language is dynamically typed so it has many design restrictions that are reported by some Python developers. It is even seen that it requires more testing time, and the errors show up when the applications are finally run.

- **Underdeveloped Database Access Layers**

As compared to the popular technologies like JDBC and ODBC, the Python’s database access layer is found to be bit underdeveloped and primitive. However, it cannot be applied in the enterprises that need smooth interaction of complex legacy data.

5.4.2 Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows a student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintaining.

- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features. A, few are listed below –

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

5.4.3 Languages Influenced by Python

Python's design and philosophy have influenced many other programming languages:

- Boo uses indentation, a similar syntax, and a similar object model.^[153]
- Cobra uses indentation and a similar syntax, and its "Acknowledgements" document lists Python first among languages that influenced it. However, Cobra directly supports design-by-contract, unit tests, and optional static typing.
- Coffee Script, a programming language that cross-compiles to JavaScript, has Python-inspired syntax.
- ECMAScript borrowed iterators and generators from Python.^[156]
- Go is designed for the "speed of working in a dynamic language like Python" and shares the same syntax for slicing arrays.
- Groovy was motivated by the desire to bring the Python design philosophy to Java.
- Julia was designed "with true macros [. and to be] as usable for general programming as Python [and] should be as fast as C". Calling to or from Julia is possible; to with PyCall.jl and a Python package pyjulia allows calling, in the other direction, from Python.
- Kotlin is a functional programming language with an interactive shell similar to Python. However, Kotlin is strongly typed with access to standard Java libraries.
- Ruby's creator, Yukihiro Matsumoto, has said: "I wanted a scripting language that was more powerful than Perl, and more object-oriented than Python. That's why I decided to design my own language".
- Swift, a programming language developed by Apple, has some Python-inspired syntax.
- GDScript, dynamically typed programming language used to create video-games. It is extremely similar to Python with a few minor differences.

Python's development practices have also been emulated by other languages. For example, the practice of requiring a document describing the rationale for, and issues surrounding, a change to the language (in Python, a PEP) is also used in Tcl and Erlang.

Python received TIOBE's Programming Language of the Year awards in 2007, 2010 and 2018. The award is given to the language with the

greatest growth in popularity over the year, as measured by the TIOBE index

There are two attributes that make development time in Python faster than in other programming languages:

1. Python is an interpreted language, which precludes the need to compile code before executing a program because Python does the compilation in the background. Because Python is a high-level programming language, it abstracts many sophisticated details from the programming code. Python focuses so much on this abstraction that its code can be understood by most novice programmers.
2. Python code tends to be shorter than comparable codes. Although Python offers fast development times, it lags slightly in terms of execution time. Compared to fully compiling languages like C and C++, Python programs execute slower. Of course, with the processing speeds of computers these days, the speed differences are usually only observed in benchmarking tests, not in real-world operations. In most cases, Python is already included in Linux distributions and Mac OS X machines.

5.5 APPLICATIONS:

Web Applications

You can create scalable Web Apps using frameworks and CMS (Content Management System) that are built on Python. Some of the popular platforms for creating Web Apps are: Django, Flask, Pyramid, Plone, Django CMS.

Sites like Mozilla, Reddit, Instagram and PBS are written in Python.

Scientific and Numeric Computing

There are numerous libraries available in Python for scientific and numeric computing. There are libraries like: SciPy and NumPy that are used in general purpose computing. And, there are specific libraries like: EarthPy for earth science, AstroPy for Astronomy and so on.

Also, the language is heavily used in machine learning, data mining and deep learning.

Creating software Prototypes

Python is slow compared to compiled languages like C++ and Java. It might not be a good choice if resources are limited and efficiency is a must.

However, Python is a great language for creating prototypes. For example: You can use Pygame (library for creating games) to create your game's prototype first. If you like the prototype, you can use language like C++ to create the actual game.

Good Language to Teach Programming

Python is used by many companies to teach programming to kids and newbies.

It is a good language with a lot of features and capabilities. Yet, it's one of the easiest languages to learn because of its simple easy-to-use syntax.

5.6 FEATURES OF PYTHON:

1. A simple language which is easier to learn

Python has a very simple and elegant syntax. It's much easier to read and write Python programs compared to other languages like: C++, Java, C#. Python makes programming fun and allows you to focus on the solution rather than syntax.

If you are a newbie, it's a great choice to start your journey with Python.

2. Free and open-source

You can freely use and distribute Python, even for commercial use. Not only can you use and distribute software's written in it, you can even make changes to the Python's source code.

Python has a large community constantly improving it in each iteration.

3. **Portability**

You can move Python programs from one platform to another, and run it without any changes.

It runs seamlessly on almost all platforms including Windows, Mac OS X and Linux.

4. **Extensible and Embeddable**

Suppose an application requires high performance. You can easily combine pieces of C/C++ or other languages with Python code.

This will give your application high performance as well as scripting capabilities which other languages may not provide out of the box.

5. **A high-level, interpreted language**

Unlike C/C++, you don't have to worry about daunting tasks like memory management, garbage collection and so on.

Likewise, when you run Python code, it automatically converts your code to the language your computer understands. You don't need to worry about any lower-level operations.

6. **Large standard libraries to solve common tasks**

Python has a number of standard libraries which makes life of a programmer much easier since you don't have to write all the code yourself. For example: Need to connect MySQL database on a Web server? You can use MySQL dB library using `import MySQL dB`.

Standard libraries in Python are well tested and used by hundreds of people. So, you can be sure that it won't break your application.

7. **Object-oriented**

Everything in Python is an object. Object oriented programming (OOP) helps you solve a complex problem intuitively.

With OOP, you are able to divide these complex problems into smaller sets by creating objects.

CHAPTER-6

CONCLUSION

In our country, many people have lost their life by accidents, because of causalities or improper communication. So, an automatic vehicle accident and theft detection system are implemented. By, using this system we can reduce death due to accidents and we can take immediate action after the accident. Also, it is used in the theft detection. A fingerprint recognition system, is used in this project. GSM module is used for conveying information about the vehicle theft to the owner. Thus, techniques presented in this project provide high security and dependability to the vehicle. The scope of the proposed methodology lies in achieving a faster and efficient vehicle security system. The demand for auto-guard systems for protecting the vehicle from theft and loss is increasing day by day. The proposed system will be an intellectual system to meet this demand.

REFERENCE

- [1] H. S. Kim, Y. Hwang, D. Yoon, W. Choi, and C. H. Park, "Driver workload characteristics analysis using EEG data from an urban road," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1844–1849, 2014.
- [2] P. Zarjam, J. Epps, and N. H. Lovell, "Beyond Subjective Self-Rating: EEG Signal Classification of Cognitive Workload," *IEEE Trans. Auton. Ment. Dev.*, vol. 7, no. 4, pp. 301–310, 2015.
- [3] NASA TLX: Task Load Index, "NASA TLX." [Online]. Available: <https://humansystems.arc.nasa.gov/groups/TLX/tlxapp.php>. [Accessed: 03-Jul-2019].
- [4] N. A. T. S. Air Traffic Management Development Centre, "Instantaneous Self-Assessment of workload (ISA)." [Online]. Available: <https://ext.eurocontrol.int/ehp/?q=node/1585>. [Accessed: 03Jul-2019].
- [5] S. Wang, J. Gwizdka, and W. A. Chaovalitwongse, "Using wireless EEG signals to assess memory workload in the \$ n \$-back task," *IEEE Trans. Human-Machine Syst.*, vol. 46, no. 3, pp. 424–435, 2015.
- [6] C. J. Behr, A. Kumar, and G. P. Hancke, "A smart helmet for air quality and hazardous event detection for the mining industry," *Proc. IEEE Int. Conf. Ind. Technol.*, vol. 2016-May, pp. 2026-2031, 2016.
- [7] S. J. Jung, H. S. Shin, and W. Y. Chung, "Highly sensitive driver health condition monitoring system using nonintrusive active electrodes," *Sensors Actuators, B Chem.*, vol. 171-172, pp. 691-698, 2012.
- [8] W. Von Rosenberg, T. Chanwimalueang, V. Goverdovsky, D. Looney, D. Sharp, and D. P. Mandic, "Smart Helmet:Wearable

Multichannel ECG and Eeg,” IEEE J. Transl. Eng. Heal. Med., vol. 4, no. August, pp. 1-11, 2016.

[9] M. K. A. Mohd Rasli, N. K. Madzhi, and J. Johari, “Smart helmet with sensors for accident prevention,” 2013 Int. Conf. Electr. Electron. Syst. Eng. ICEESE 2013, vol. 1, pp. 21-26, 2013.

[10] A. R. Budiman, D. W. Sudiharto, and T. Brotoharsono, “The Prototype of Smart Helmet with Safety Riding Notification for Motorcycle Rider,” in 2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE), 2019, pp. 362-367.

[11] M. Marimuthu, S. Nivetha, and K. Sirushti, “Accident Detection and Reporting System using Internet of Things,” Res. J. Sci. Eng. Syst., vol. 3, no. 2, pp. 121-129, 2018.

[12] D. Selvathi, P. Pavithra, and T. Preethi, “Intelligent transportation system for accident prevention and detection,” Proc. 2017 Int. Conf. Intell. Comput. Control Syst. ICICCS 2017, vol. 2018-Janua, pp. 442-446, 2018.

[13] A. Ajay, G. Vishnu, V. Kishoreswaminathan, V. Vishwanth, K. Srinivasan, and S. Jeevanantham, “Accidental identification and navigation system in helmet,” in 2017 International Conference on Nextgen Electronic Technologies: Silicon to Software, ICNETS2 2017, 2017, pp. 202-204.

[14] F. M. Abdul Jalil, “A framework for vehicular accident management using wireless networks,” Proc. 2012 IEEE 13th Int. Conf. Inf. Reuse Integr. IRI 2012, pp. 727-729, 2012.

[15] S. R. Rupanagudi et al., “A novel video processing based smart helmet for rear vehicle intimation & collision avoidance,” 2015 Int. Conf. Comput. Newt. Commun. CoCoNet 2015, pp. 799-805, 2016.

[16] B. Shabrin, M. Poojary, T. Pooja, and B. Sadhana, "Smart Helmet Intelligent Safety for Motorcyclist Using Raspberry Pi and Open Cv," *Int. Res. J. Eng. Technol.*, vol. 3, no. 3, pp. 589-593, 2016.

[17] M. Mehrubeoglu, L. M. Pham, H. T. Le, R. Muddu, and D. Ryu, "Realtime eye tracking using a smart camera," *Proc. - Appl. Image. Pattern Recognin. Work.*, pp. 1-7, 2011.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "D l-[^] f t[^] f f i0 5 - 2_2015_Yolo (Cvpr)," *Proc. 2016 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR 2016)*, pp. 779-788, 2015.

[19] J. D. Dignam, P. L. Martin, B. S. Shastry, and R. G. Roeder, "Eukaryotic gene transcription with purified components," *Methods Enzymol.*, vol. 101, no. C, pp. 582-598, 1983.

[20] S.-W. Ahn, "Design and Implementation of Smart LED Bicycle Helmet using Arduino," *J. Korea Inst. Inf. Commun. Eng.*, vol. 20, no. 6, pp. 1148 1153, 2016.