IOT BASED FACIAL RECOGNITION HOME SECURITY SYSTEM WITH EMAIL ALERT USING RASPBERRY PI

A Project report submitted in partial fulfilment

of the requirements for the degree of B. Tech in Electrical Engineering

By

1. **JAKIRUDDIN KHAN (11701619011)**
2. **JEET BERA (11701618042)**
3. **MANOBANDRO MRIDHA (11701619016)**

Under the supervision of

Mr. Deepam Gangopadhyay

Assistant Professor

Department of Electrical Engineering

Department of Electrical Engineering

RCC INSTITUTE OF INFORMATION TECHNOLOGY

CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015, WEST BENGAL

Maulana Abul Kalam Azad University of Technology (MAKAUT)

Department of Electrical Engineering

RCC INSTITUTE OF INFORMATION TECHNOLOGY

CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015, WEST BENGAL

CERTIFICATE

To whom it may concern

This is to certify that the project work entitled **IOT BASED FACIAL RECOGNITION HOME SECURITY SYSTEM WITH EMAIL ALERT USING RASPBERRY PI**_is the bona fide work carried out by **Jakiruddin Khan (Roll No. 11701619011)**, a student of B.Tech in the Dept. of Electrical Engineering, RCC Institute of Information Technology (RCCIIT), Canal South Road, Beliaghata, Kolkata-700015, affiliated to Maulana Abul Kalam Azad University of Technology (MAKAUT), West Bengal, India, during the academic year 2021-22, in partial fulfillment of the requirements for the degree of Bachelor of Technology in Electrical Engineering and this project has not submitted previously for the award of any other degree, diploma and fellowship.

Signature of the Guide	**Signature of the HOD, EE**
Name:	**Name:**
Designation	**Designation**

Signature of the External Examiner

Name:

Designation:

Department of Electrical Engineering

RCC INSTITUTE OF INFORMATION TECHNOLOGY

CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015, WEST BENGAL

CERTIFICATE

To whom it may concern

This is to certify that the project work entitle **IOT BASED FACIAL RECOGNITION HOME SECURITY SYSTEM WITH EMAIL ALERT USING RASPBERRY PI** _is the bona fide work carried out by **Jeet Bera (Roll No. 11701618042)** , a student of B.Tech in the Dept. of Electrical Engineering, RCC Institute of Information Technology (RCCIIT), Canal South Road, Beliaghata, Kolkata-700015, affiliated to Maulana Abul Kalam Azad University of Technology (MAKAUT), West Bengal, India, during the academic year 2021-22, in partial fulfillment of the requirements for the degree of Bachelor of Technology in Electrical Engineering and this project has not submitted previously for the award of any other degree, diploma and fellowship.

Signature of the Guide

Name:

Designation

Signature of the HOD, EE

Name:

Designation

Signature of the External Examiner

Name:

Designation:

Department of Electrical Engineering

RCC INSTITUTE OF INFORMATION TECHNOLOGY

CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015, WEST BENGAL

CERTIFICATE

To whom it may concern

This is to certify that the project work **IOT BASED FACIAL RECOGNITION HOME SECURITY SYSTEM WITH EMAIL ALERT USING RASPBERRY PI** _is the bona fide work carried out by **Manobandro Mridha (Roll No. 11701619016)**, a student of B.Tech in the Dept. of Electrical Engineering, RCC Institute of Information Technology (RCCIIT), Canal South Road, Beliaghata, Kolkata-700015, affiliated to Maulana Abul Kalam Azad University of Technology (MAKAUT), West Bengal, India, during the academic year 2021-22, in partial fulfillment of the requirements for the degree of Bachelor of Technology in Electrical Engineering and this project has not submitted previously for the award of any other degree, diploma and fellowship.

Signature of the Guide	**Signature of the HOD, EE**
Name:	**Name:**
Designation	**Designation**

Signature of the External Examiner

Name:

Designation:

Acknowledgements

It is my great fortune that I have got opportunity to carry out this project work under the supervision of **Prof Mr. Deepam Gangopadhyay** in the Department of Electrical Engineering, RCC Institute of Information Technology (RCCIIT), Canal South Road, Beliaghata, Kolkata-700015, affiliated to Maulana Abul Kalam Azad University of Technology (MAKAUT), West Bengal, India. I express my sincere thanks and deepest sense of gratitude to my guide for his constant support, unparalleled guidance and limitless encouragement.

I wish to convey my gratitude to Prof. (Dr.) Shilpi Bhattacharya, HOD, Department of Electrical Engineering, RCCIIT and to the authority of RCCIIT for providing all kinds of infrastructural facility towards the research work.

I would also like to convey my gratitude to all the faculty members and staffs of the Department of Electrical Engineering, RCCIIT for their whole hearted cooperation to make this work turn into reality.

Signature of the Students

Place:

Date:

ABSTRACT

The rapid advancement of Internet of Things (IoT) technology has revolutionized the way we interact with our surroundings. In this project, we propose an IoT-based facial recognition home security system with email alert using Raspberry Pi. The system aims to enhance home security by employing facial recognition techniques to identify individuals and trigger appropriate actions.

The proposed system utilizes a Raspberry Pi as the central control unit, equipped with a camera module for capturing facial images. A trained machine learning model is employed to recognize faces and match them against a pre-defined database of authorized individuals. Upon successful identification, the system grants access to the recognized person, activating or deactivating security measures accordingly.

Additionally, the system incorporates email alert functionality to promptly notify homeowners of any unauthorized access attempts. In case of a potential security breach, an email containing relevant details, such as images of the intruder, date, and time, is sent to the homeowner's specified email address.

The implementation of this system offers several benefits, including real-time monitoring, improved accuracy in identifying authorized individuals, and quick notification of security breaches. It provides homeowners with a robust and reliable home security solution, leveraging the power of IoT and facial recognition technology.

To evaluate the system's performance, we conducted extensive testing, considering various scenarios and evaluating the accuracy of facial recognition, response time, and email alert reliability. The results demonstrate the effectiveness and feasibility of the proposed system in enhancing home security.

Keywords: IoT, facial recognition, home security, Raspberry Pi, email alert.

TABLE OF CONTENTS

Executive Summery

Chapter 1: Introduction	07-09
• Identification of the Problem and related current scenario • How this problem is being solved at present • Where is the flaw of current system? • What works already have been done by others, to address these flaws • What Change our work is expected to bring	
Chapter 2: Theoretical Study	10 - 26
• Embedded System & IoT • WiFi Technology • Raspberry Pi & Camera • Power Supply	
Chapter 3: Technology behind your work	27 - 29
• Technology Concepts • Block diagram of different modules of our work	
Chapter 4: Detail design and description	30 - 45
• Detail design and description of each module • Complete circuit diagram of each Module	
Chapter 5: Software and Coding	46 - 52
• Software and IDE • Flow Chart and Coding	
Chapter 6: Bill of Materials and Cost Estimation	53-54
• Bill of Materials • Comparative Analysis of cost vs. benefit	
Chapter 7: Results and interpretations	55-57
• Results • Interpretation	
Chapter 8: Conclusions and Future Scope	58-59
• Conclusion • Future scope	
Chapter 9: References	60
Appendix A : Appendix B :	

Chapter 1: Introduction

1.1 General Introduction

The rapid growth of Internet of Things (IoT) technology has paved the way for innovative applications in various domains, including home security. This final year B.Tech project focuses on the development of an IoT-based facial recognition home security system with email alert, utilizing the Raspberry Pi platform. By integrating facial recognition algorithms with IoT principles, the system aims to enhance the security of residential premises. The project emphasizes real-time monitoring, accurate identification of authorized individuals, and timely notification of potential security breaches via email alerts. This introduction sets the stage for a comprehensive exploration of the project, highlighting its significance in addressing modern-day home security challenges.

1.2 Identification of the Problem and related current scenario

In the current scenario, traditional home security systems often rely on physical locks, keys, and alarm systems, which can be susceptible to security breaches and false alarms. These systems lack advanced identification techniques, making it difficult to distinguish authorized individuals from intruders. Additionally, homeowners often face delays in being notified of security breaches, hindering their ability to take prompt action.

To address these challenges, this final year B.Tech project proposes an IoT-based facial recognition home security system with email alert using Raspberry Pi. By integrating facial recognition technology with IoT principles, the project aims to provide homeowners with a more accurate and efficient method of identifying individuals and promptly alerting them of potential security breaches, improving overall home security.

1.3 How this problem is being solved at present

Currently, advancements in facial recognition technology have led to the development of more sophisticated home security systems. However, many of these systems are expensive, complex to install, and require specialized hardware. In contrast, this final year B.Tech project proposes a cost-effective solution utilizing the Raspberry Pi platform. By leveraging the power of IoT and facial recognition algorithms, the system enables real-time monitoring, accurate identification of authorized individuals, and prompt email notifications in case of security breaches. This project offers an accessible and affordable alternative for homeowners to enhance their home security measures, filling the gap in the current market.

1.4 Where is the flaw of current system?

The current home security systems often lack advanced identification techniques, relying on traditional locks, keys, and alarm systems. This approach can be vulnerable to security breaches, as unauthorized individuals can easily bypass these measures. Moreover, the delay in detecting and notifying homeowners of security breaches poses a significant flaw. The lack of real-time monitoring and accurate identification systems further weakens the overall security. These limitations highlight the need for an improved system that incorporates facial recognition technology, IoT principles, and prompt email notifications to address the flaws of the current system and provide homeowners with a more robust and reliable home security solution.

1.5 What works already have been done by others, to address these flaws

Researchers and developers have made significant progress in addressing the flaws of current home security systems. Several studies and projects have focused on integrating facial recognition technology with IoT to enhance home security. Existing works have explored the use of machine learning algorithms for accurate facial identification and authentication. Additionally, advancements have been made in developing real-time monitoring systems using Raspberry Pi and other IoT devices. Some projects have also implemented email notification systems to alert homeowners of security breaches. These collective efforts provide valuable insights and techniques that can be leveraged to build upon and improve the proposed IoT-based facial recognition home security system with email alert using Raspberry Pi.

1.6 What Change our work is expected to bring

Simplicity: Our system offers a user-friendly interface and easy installation process, making it accessible to homeowners without extensive technical expertise.

Low Cost: By utilizing the affordable Raspberry Pi platform and open-source software, our solution provides a cost-effective alternative compared to expensive commercial systems.

Reliability: The integration of facial recognition technology improves the accuracy of identifying authorized individuals and reduces false alarms, ensuring a more reliable security system.

Smart Work: The incorporation of IoT principles enables real-time monitoring and prompt email alerts, empowering homeowners to take immediate action in case of security breaches.

Overall, our project aims to bring about a transformative change by offering a low-cost, reliable, and smart home security solution that enhances the safety and peace of mind for homeowners.

Chapter 2: Theoretical Study

2.1 Embedded System & IoT

An embedded system is a microprocessor-based computer hardware system whose software is designed to perform a dedicated function, either as an independent system or as part of a larger system. The main part is an integrated circuit designed to calculate for real-time activities. Complexity ranges from a single microcontroller to a suite of processors with connected peripherals and networks; From a user interface to a complex graphical user interface. The complexity of an embedded system varies significantly depending on what it is designed for. Embedded system applications range from digital watches and microwaves to hybrid vehicles and avionics. As much as 98 % of all microprocessors manufactured are used in embedded systems.

Embedded systems are managed by microcontrollers or digital signal processors (DSP), application-specific integrated circuits (ASIC), field-programmable gate arrays (FPGA), GPU technology, and gate arrays. These processing systems are integrated with components dedicated to handling electric and/or mechanical interfacing. Embedded systems programming instructions, referred to as firmware, are stored in read-only memory or flash memory chips, running with limited computer hardware resources. Embedded systems connect with the outside world through peripherals, linking input and output devices.

So, an embedded system is nearly any computing system other than a desktop computer. An embedded system is a dedicated system which performs the desired function upon power up, repeatedly.

Classifications of Embedded systems

1. Small Scale Embedded Systems
2. Medium Scale Embedded Systems
3. Sophisticated Embedded Systems

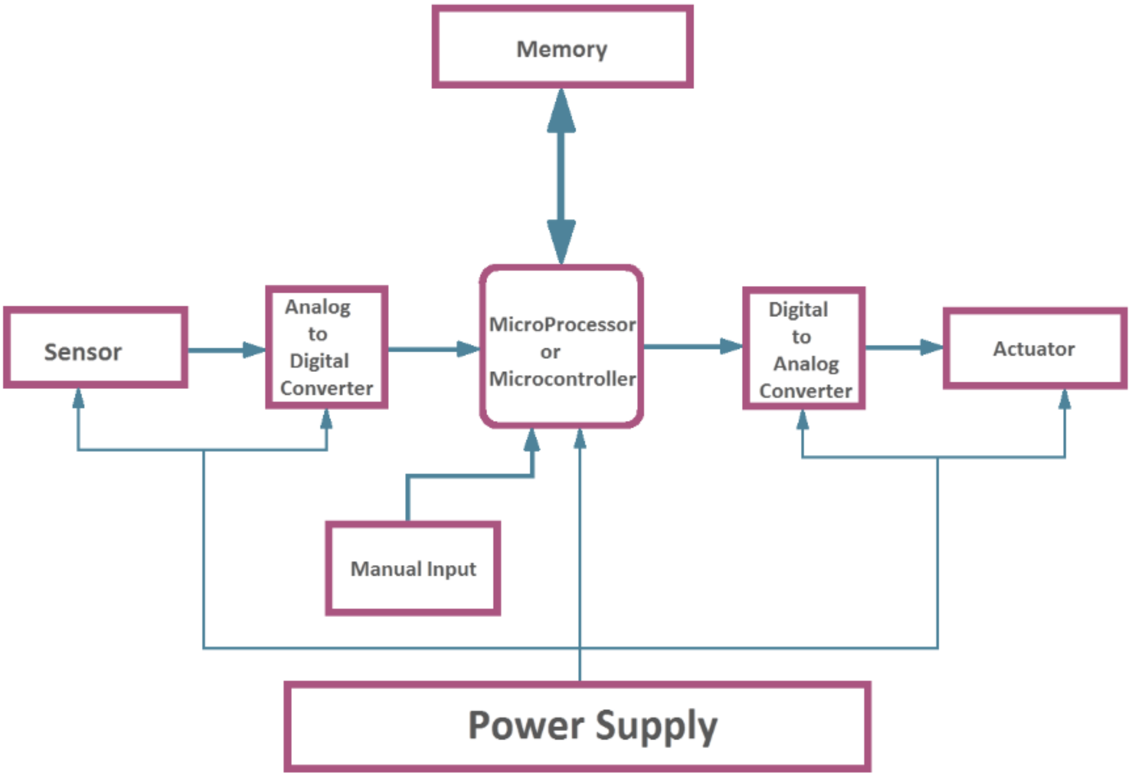
The processing units of the embedded system

1. Processor in an Embedded System A processor is an important unit in the embedded system hardware. A microcontroller is an integrated chip that has the processor, memory and several other hardware units in it; these form the microcomputer part of the embedded system.

2. Commonly used microprocessors, microcontrollers and DSPs in the small-, medium- and large scale embedded systems

3. A recently introduced technology that additionally incorporates the application-specific system processors (ASSPs) in the embedded systems.

4. Multiple processors in a system.


Memory
Sensor
Analog to Digital Converter
MicroProcessor or Microcontroller
Digital to Analog Converter
Actuator
Manual Input
Power Supply

Embedded System Block Diagram

Basic Structure of an Embedded System

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

- Sensor: The sensor measures and converts the physical quantity to an electrical signal, which can then be read by an embedded systems engineer or any electronic instrument. A sensor stores the measured quantity to the memory.
- A-D Converter: An analog-to-digital converter converts the analog signal sent by the sensor into a digital signal.
- Processor & ASICs: Processors assess the data to measure the output and store it to the memory.
- D-A Converter: A digital-to-analog converter changes the digital data fed by the processor to analog data
- Actuator: An actuator compares the output given by the D-A Converter to the actual output stored and stores the approved output.

Microcontroller & Microprocessor in Embedded System

Microprocessor

A microprocessor is a single VLSI chip having a CPU. In addition, it may also have other units such as coaches, floating point processing arithmetic unit, and pipelining units that help in faster processing of instructions.

Microprocessor has only the CPU inside them i.e. only the processing powers such as Intel's Pentium 1,2,3,4, core 2 duo, i3, i5 etc. These microprocessors don't have RAM, ROM, and other peripheral on the chip. A system designer has to add them externally to make them functional.

Microcontroller

A microcontroller is a single-chip VLSI unit (also called microcomputer) which, although having limited computational capabilities, possesses enhanced input/output capability and a number of on-chip functional units like CPU, RAM, ROM, I/O Port, Timer, and Serial COM Port .

Microcontrollers are particularly used in embedded systems for real-time control applications with on-chip program memory and devices.

So in the case of microcontrollers, it has a CPU, in addition with a fixed amount of RAM, ROM and other peripherals all embedded on a single chip. At times it is also termed as a mini computer or a computer on a single chip. Some manufacturers are ATMEL, Microchip, TI, Freescale, Philips, Motorola etc

Embedded Programming Language

An embedded programming language is a programming language that developers use in embedded systems. In general, the languages offer low-level access to the device hardware. Developers use several common programming languages for embedded systems.

Developers use a variety of programming languages in embedded systems. The most used languages include Assembly , C, C++, Python, MicroPython, and Java.

Assembly : Assembly is a low-level programming language that directly communicates with computer hardware. The language is memory efficient and fast. Assembly can be difficult to read and maintain.

C : Developed in the early 1970s, C is a compiled language that serves as a building block of many other languages. C is an efficient and widely used programming language. Industry estimates say 80% of embedded systems use the C programming language.

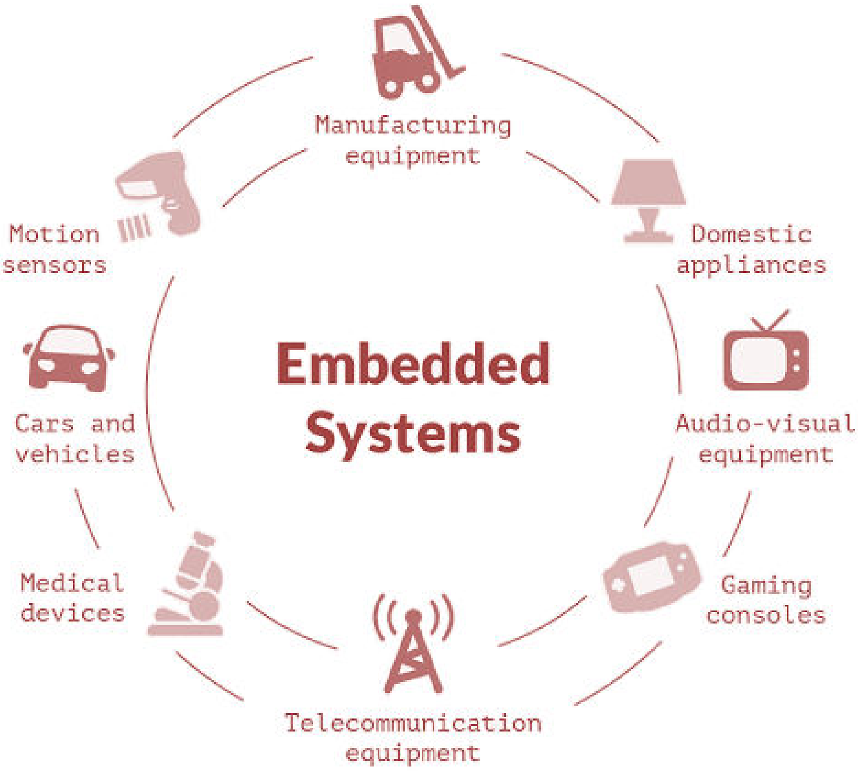
C++ : This compiled language has most or all elements of C, but also other capabilities. The language was designed in part for system programming and embedded systems. C++ can be as efficient as using C, but it has a standards library that can save programmers time in writing code.

Python : Developed in the early 1980's , Python is a popular programming language. It excels in machine learning, artificial intelligence (AI), and data analytics, but you can use it in many other applications. The language is open source, free to use, and easy to learn, read, and write. This language is not deterministic— not for real-time operating systems (RTOS).

MicroPython : This language is a version of Python optimized for microcontrollers. It is also open source, free to use, and easy to learn. The code isn't as fast and may use more memory compared to C or C++

Java : Java is an efficient, general-purpose language used extensively for internet-based applications. In embedded systems, Java is best for those running on the Android OS. Once written in an embedded system, the code is portable to another device and quite reliable.

Other languages for embedded systems : Rust, Ada, Go



IoT (Internet Of Things) :

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

Why is Internet of Things (IoT) so important?

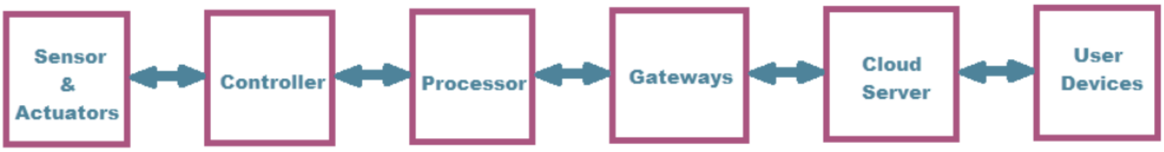
Over the past few years, IoT has become one of the most important technologies of the 21st century. Now that we can connect everyday objects—kitchen appliances, cars, thermostats, baby monitors—to the internet via embedded devices, seamless communication is possible between people, processes, and things.

By means of low-cost computing, the cloud, big data, analytics, and mobile technologies, physical things can share and collect data with minimal human intervention. In this hyper connected world, digital systems can record, monitor, and adjust each interaction between connected things. The physical world meets the digital world—and they cooperate.

What technologies have made IoT possible?

While the idea of IoT has been in existence for a long time, a collection of recent advances in a number of different technologies has made it practical.

- **Access to low-cost, low-power sensor technology.** Affordable and reliable sensors are making IoT technology possible for more manufacturers.
- **Connectivity.** A host of network protocols for the internet has made it easy to connect sensors to the cloud and to other "things" for efficient data transfer.
- **Cloud computing platforms.** The increase in the availability of cloud platforms enables both businesses and consumers to access the infrastructure they need to scale up without actually having to manage it all.
- **Machine learning and analytics.** With advances in machine learning and analytics, along with access to varied and vast amounts of data stored in the cloud, businesses can gather insights faster and more easily. The emergence of these allied technologies continues to push the boundaries of IoT and the data produced by IoT also feeds these technologies.
- **Conversational artificial intelligence (AI).** Advances in neural networks have brought natural-language processing (NLP) to IoT devices (such as digital personal assistants Alexa, Cortana, and Siri) and made them appealing, affordable, and viable for home use.



IOT Basic Block Diagram and Architecture

Sensor and Actuator

Sensor and Actuator work in the opposite principle to each other. A sensor is a device that senses the physical changes in its environment and produces an electrical or electronic signal. An actuator is a device that makes physical changes when it gets electrical or electronic signals.

For example, a temperature sensor senses the temperature across its environment. Similarly, a humidity sensor senses the humidity. A solenoid valve as an Actuator got opens when it gets an electrical power supply and allows the flow of any liquid or gas through it.

Controller

The controller control all those sensors and actuators. Sensors and actuators are analog devices but IoT system works with digital data, so the controller also helps to analog to digital and digital to analog conversion.

Processor

It collects data from all sensors through the controller in the form of digital signals and processes them. The main function of the processor to collect all data, arrange them and process them.

Gateways

Gateways are responsible to transmit data of an IoT system through the internet. They modulate and demodulate the data for transmission. Gateways help to transmit the data from the processor of the IoT system to the server for storing. Gateways use a standard protocol system(ex: MQTT, XMPP) for data transmission. The example of Gateway devices is Modem, GSM system, WiFI, etc.

Cloud Server

Cloud Server is the space for storing data. Cloud server stores all the data of an IoT system and it serves with the users when they requested. This cloud server only helps to control devices over the internet. They serve the data with users located anywhere in the world. Actually, IoT works in almost the same principle as other digital control systems like SCADA, PLC, etc just difference is, the IoT system works through the internet.

User Device

It is the device actually where the data are used to observe the status, analytics, and control the devices of the IoT system.

For example, a production plant implemented with IoT. So the manager of the plant can observe production status from anywhere in the world on his smartphone or computer through the Analytics application.

2.2 WiFi Technology

Wi-Fi is a wireless networking technology that allows devices such as computers (laptops and desktops), mobile devices (smart phones and wearables), and other equipment (printers and video cameras) to interface with the Internet. It allows these devices--and many more--to exchange information with one another, creating a network. Internet connectivity occurs through a wireless router. When you access Wi-Fi, you are connecting to a wireless router that allows your Wi-Fi-compatible devices to interface with the Internet.

On the technical side, the IEEE 802.11 standard defines the protocols that enable communications with current Wi-Fi-enabled wireless devices, including wireless routers and wireless access points. Wireless access points support different IEEE standards. Each standard is an amendment that was ratified over time. The standards operate on varying frequencies, deliver different bandwidth, and support different numbers of channels.

ESP8266 WiFi Module

ESP8266EX delivers highly integrated Wi-Fi SoC solution to meet users' continuous demands for efficient power usage, compact design and reliable performance in the Internet of Things industry.

With the complete and self-contained Wi-Fi networking capabilities, ESP8266EX can perform either as a standalone application or as the slave to a host MCU. When ESP8266EX hosts the application, it promptly boots up from the flash. The integrated highspeed cache helps to increase the system performance and optimize the system memory. Also, ESP8266EX can be applied to any microcontroller design as a Wi-Fi adaptor through SPI/SDIO or UART interfaces.

ESP8266EX integrates antenna switches, RF balun, power amplifier, low noise receive amplifier, filters and power management modules. The compact design minimizes the PCB size and requires minimal external circuitries.

Besides the Wi-Fi functionalities, ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor and on-chip SRAM. It can be interfaced with external sensors and other devices through the GPIOs. Software Development Kit (SDK) provides sample codes for various applications.

Smart Connectivity Platform (ESCP) enables sophisticated features including:

1. Fast switch between sleep and wakeup mode for energy-efficient purpose;
2. Adaptive radio biasing for low-power operation
3. Advance signal processing
4. Spur cancellation and RF co-existence mechanisms for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation

Wi-Fi Key Features

1. 802.11 b/g/n support
2. 802.11 n support (2.4 GHz), up to 72.2 Mbps
3. Defragmentation
4. x virtual Wi-Fi interface
5. Automatic beacon monitoring (hardware TSF)
6. Support Infrastructure BSS Station mode/SoftAP mode/Promiscuous mode

Functional Block Diagram

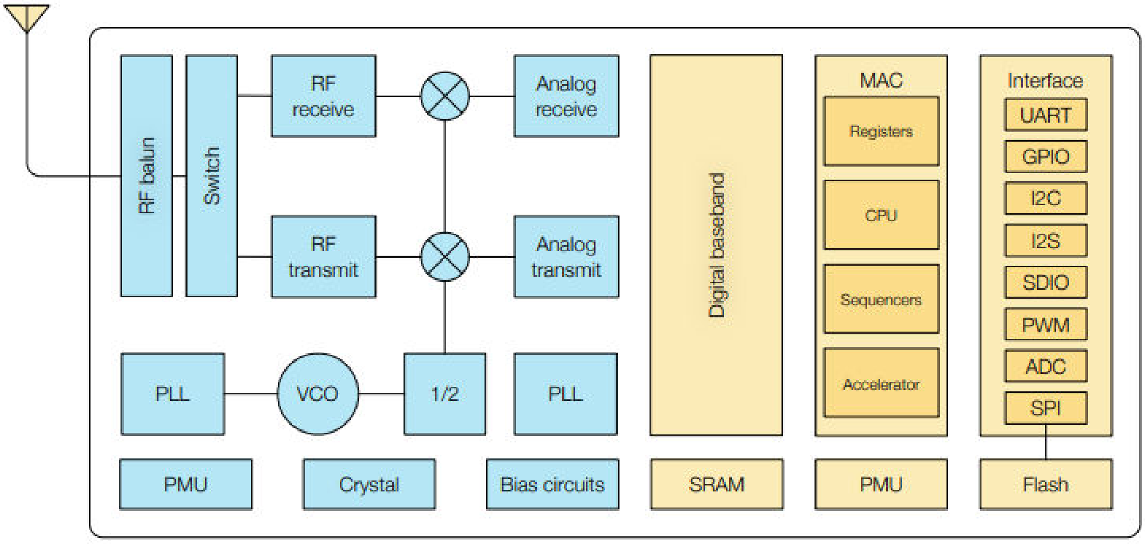


Figure 3-1. Functional Block Diagram

Pin Diagram

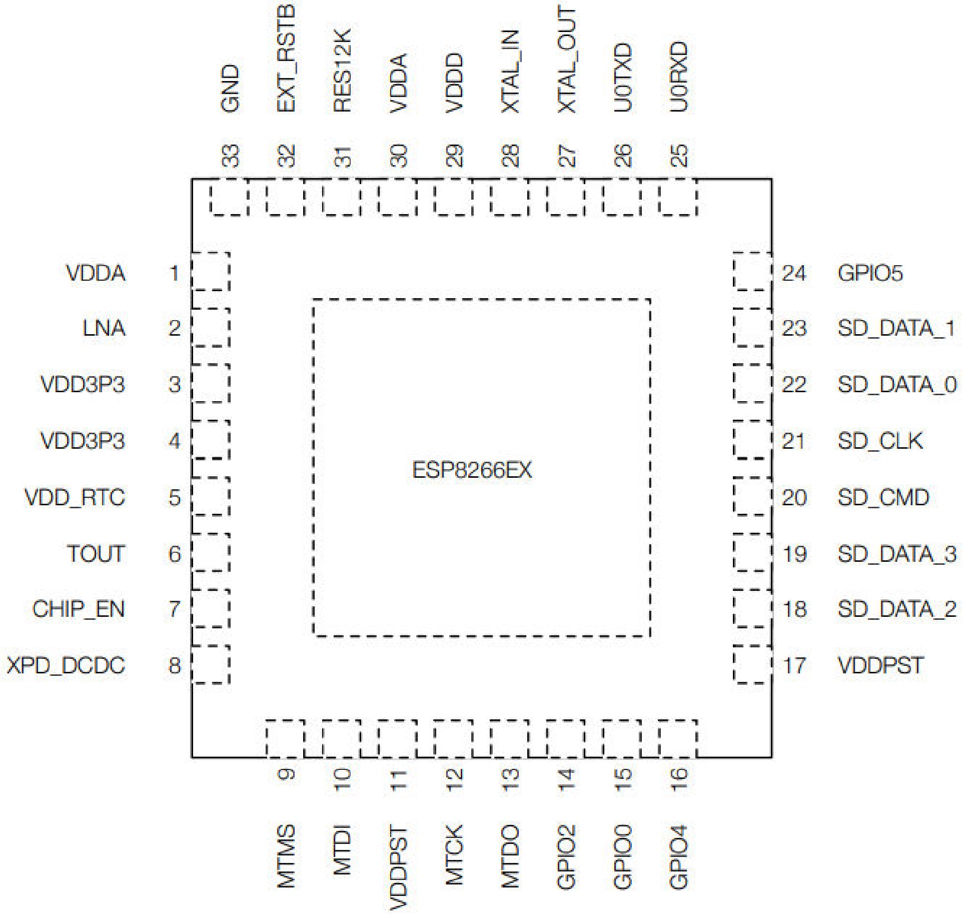


Figure 2-1. Pin Layout (Top View)

ESP8266 -01 Module – Ai-Thinker

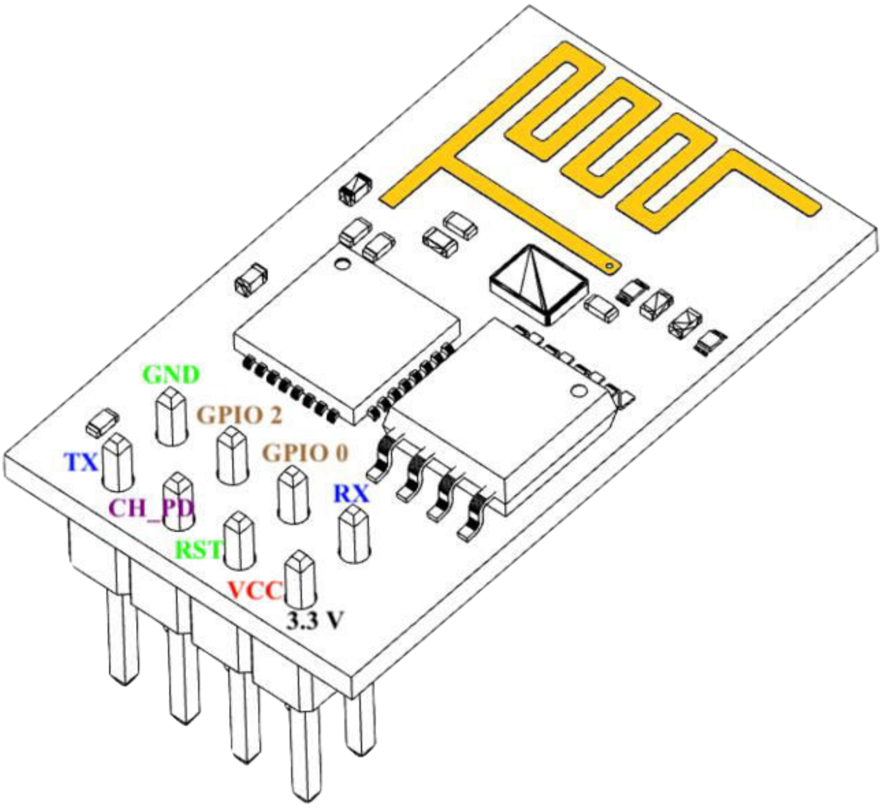
ESP-01 WiFi module is developed by Ai-thinker Team. Core processor ESP8266 in smaller sizes of the module encapsulates Tensilica L106 integrates ultra low power 32-bit bit MCU micro, with the 16 16-bit short mode. Clock speed support 80MHz, 160MHz, supports the RTOS, integrated Wi-Fi MAC/BB/ RF/PA/LNA, /BB/RF/PA/LNA, on on-board antenna.

The module supports standard IEEE802.11 b/g/n agreement, complete TCP/IP protocol stack. Users can use the add modules to an existing device networking, or building a separate network controller. When ESP8266EX hosts the application, it boots up directly from an external flash. In has integrated cache to improve the performance of the system in such applications

This can be easily developed for secondary development, access to cloud services, and realize mobile phones 3/4G global control anytime, anywhere, accelerate product prototype design.

Pin Description:

- VCC +3.3V power supply
- GND Ground (0V)
- GPIO0 General Purpose Input/Output pin 0
- GPIO2 General Purpose Input/Output pin 2
- CH_PD Chip Enable
- RST Reset
- RX Receive line of Serial Interface
- TX Transmit line of Serial Interface



2.3 Raspberry Pi & Camera

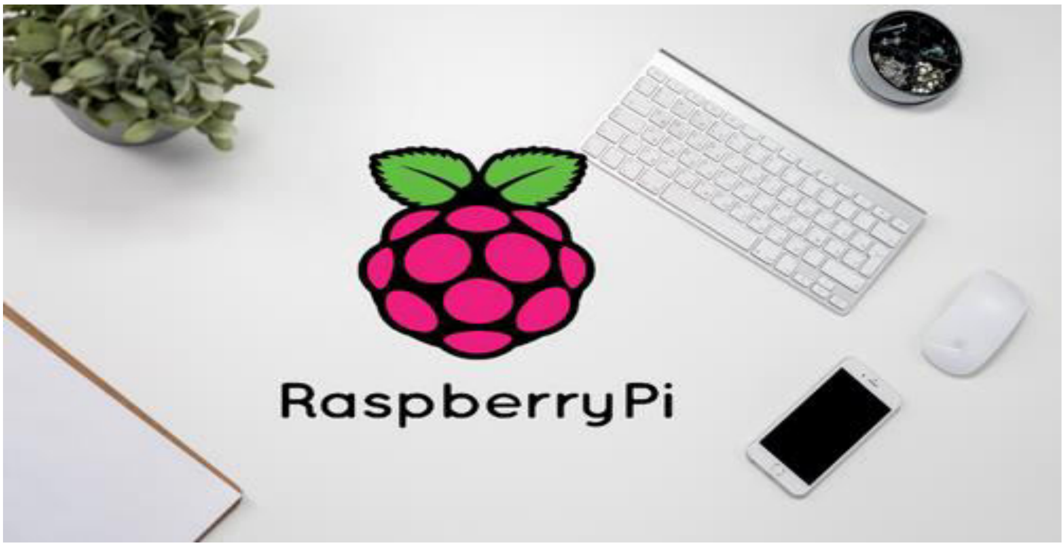
The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spread sheets, word-processing, and playing games.

What's more, the Raspberry Pi has the ability to interact with the outside world, and has been used in a wide array of digital maker projects, from music machines and parent detectors to weather stations and tweeting birdhouses with infra-red cameras. We want to see the Raspberry Pi being used by kids all over the world to learn to program and understand how computers work.

The Raspberry Pi 3 Model A+ is the latest product in the Raspberry Pi 3 range. Like the Raspberry Pi 3 Model B+, it boasts a 64-bit quad core processor running at 1.4 GHz, dual-band 2.4 GHz and 5 GHz wireless LAN, and Bluetooth 4.2/BLE.

The dual-band wireless LAN comes with modular compliance certification, allowing the board to be designed into end products with significantly reduced wireless LAN compliance testing, improving both cost and time to market.

The Raspberry Pi 3 Model A+ has the same mechanical footprint as the Raspberry Pi 1 Model A+.



Specifications:

A. Processor: Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4 GHz

B. Memory: 512MB LPDDR2 SDRAM

C. Connectivity: 2.4 GHz and 5 GHz IEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2/BLE 1 × USB 2.0 port

D. Access: Extended 40-pin GPIO header

E. Video & sound: 1 × full size HDMI MIPI DSI display port MIPI CSI camera port 4 pole stereo output and composite video port

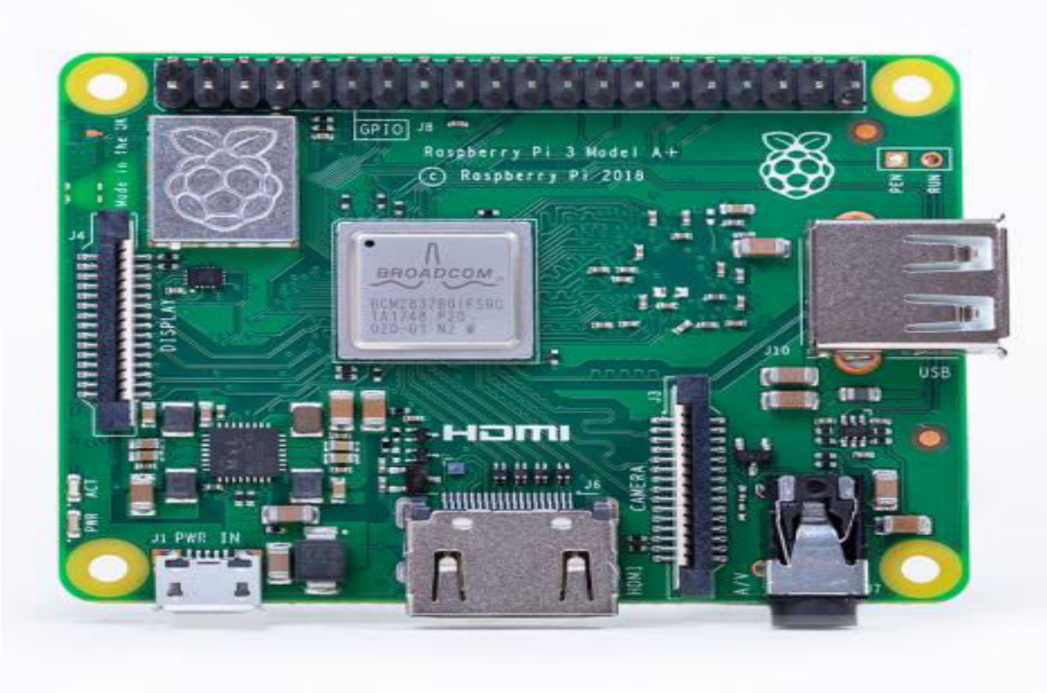
F. Multimedia: H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics

G. SD card support: Micro SD format for loading operating system and data storage

H. Input power: 5 V/2.5 A DC via micro USB connector 5 V DC via GPIO header

I. Environment: Operating temperature, 0–50°C

J. Production lifetime: The Raspberry Pi 3 Model A+ will remain in production until at least January 2023



Software & OS / Software installation

Raspberry Pi OS

The Raspberry Pi needs an operating system to work. This is it. Raspberry Pi OS (previously called Raspbian) is the official supported operating system.

Raspberry Pi OS is a free operating system based on Debian, optimised for the Raspberry Pi hardware, and is the recommended operating system for normal use on a Raspberry Pi. The OS comes with over 35,000 packages: precompiled software bundled in a nice format for easy installation on your Raspberry Pi.

Raspberry Pi OS is under active development, with an emphasis on improving the stability and performance of as many Debian packages as possible on Raspberry Pi.

Updating and Upgrading Raspberry Pi OS

It's important to keep your Raspberry Pi up to date. The first and probably the most important reason is security. A device running Raspberry Pi OS contains millions of lines of code that you rely on. Over time, these millions of lines of code will expose well-known vulnerabilities, which are documented in publicly available databases meaning that they are easy to exploit. The only way to mitigate these exploits as a user of Raspberry Pi OS is to keep your software up to date, as the upstream repositories track CVEs closely and try to mitigate them quickly.

The second reason, related to the first, is that the software you are running on your device most certainly contains bugs. Some bugs are CVEs, but bugs could also be affecting the desired functionality without being related to security. By keeping your software up to date, you are lowering the chances of hitting these bugs.

Setting up your Raspberry Pi

To get started with your Raspberry Pi computer you'll need the following accessories:

A computer monitor or television. Most should work as a display for the Raspberry Pi, but for best results, you should use a display with HDMI input. You'll also need an appropriate display cable, to connect your monitor to your Raspberry Pi.

A computer keyboard and mouse, any standard USB keyboard and mouse will work with your Raspberry Pi.

Raspberry Pi computers use a micro SD card

A network (Ethernet) cable to connect your Raspberry Pi to your local network and the Internet.

A good quality power supply. Recommend the official Raspberry Pi Power Supply which has been specifically designed to consistently provide +5.1V despite rapid fluctuations in current draw.

Introducing the Raspberry Pi Cameras

Raspberry Pi currently sells two types of camera board: an 8MP device and a 12MP High Quality (HQ) camera. The 8MP device is also available in NoIR form without an IR filter. The original 5MP device is no longer available from Raspberry Pi.

All Raspberry Pi cameras are capable of taking high-resolution photographs, along with full HD 1080p video, and can be fully controlled programmatically. Once installed, there are various ways the cameras can be used. The simplest option is to use one of the provided camera applications, such as libcamera-still or raspistill.

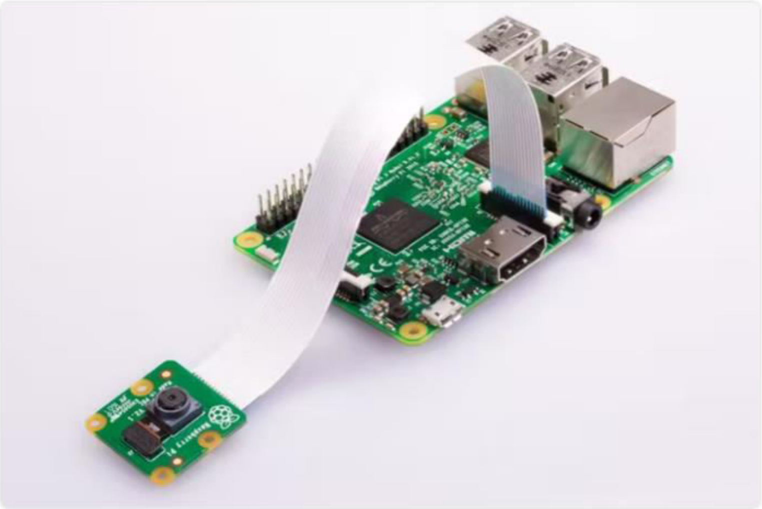
Camera Modules: There are several official Raspberry Pi camera modules. The original 5-megapixel model was released in 2013, and an 8-megapixel Camera Module v2 was released in 2016. For both iterations, there are visible light and infrared versions. A 12-megapixel High Quality Camera was released in 2020. There is no infrared version of the HQ Camera, however the IR Filter can be removed if required.

Installing a Raspberry Pi camera:

Cameras are sensitive to static. Earth yourself prior to handling the PCB. A sink tap or similar should suffice if you don’t have an earthing strap.

Connecting the Camera : The flex cable inserts into the connector labelled CAMERA on the Raspberry Pi, which is located between the Ethernet and HDMI ports. The cable must be inserted with the silver contacts facing the HDMI port. To open the connector, pull the tabs on the top of the connector upwards, then towards the Ethernet port. The flex cable should be inserted firmly into the connector, with care taken not to bend the flex at too acute an angle. To close the connector, push the top part of the connector towards the HDMI port and down, while holding the flex cable in place.

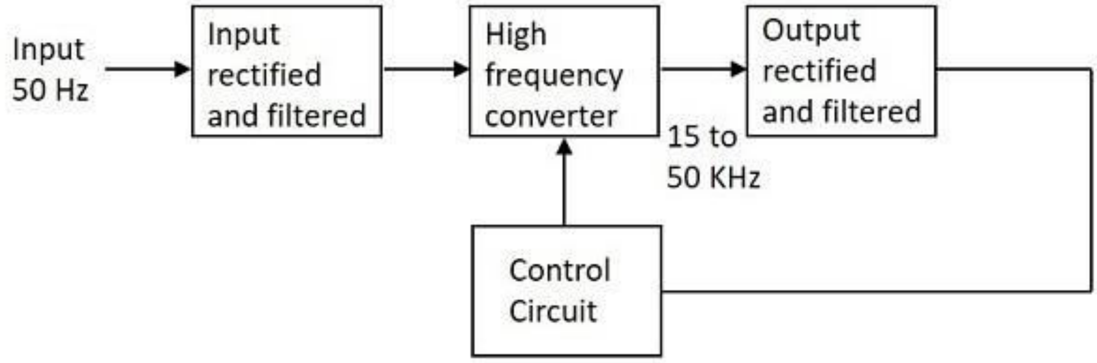
Depending on the model, the camera may come with a small piece of translucent blue plastic film covering the lens. This is only present to protect the lens while it is being mailed, and needs to be removed by gently peeling it off.



2.5 Power Supply

The Linear Power Supply LPSLPS is the regulated power supply which dissipates much heat in the series resistor to regulate the output voltage which has low ripple and low noise. The disadvantages of LPS such as lower efficiency, the need for large value of capacitors to reduce ripples and heavy and costly transformers etc. are overcome by the implementation of Switched Mode Power Supplies.

The working of SMPS is simply understood by knowing that the transistor used in LPS is used to control the voltage drop while the transistor in SMPS is used as a controlled switch.



Input Stage

The AC input supply signal 50 Hz is given directly to the rectifier and filter circuit combination without using any transformer. This output will have many variations and the capacitance value of the capacitor should be higher to handle the input fluctuations. This unregulated dc is given to the central switching section of SMPS.

Switching Section

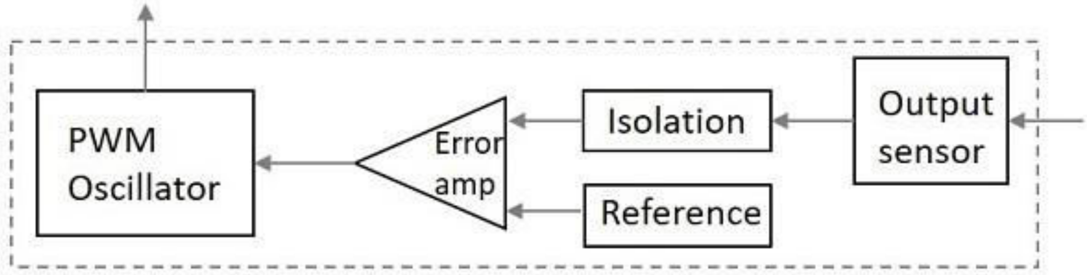
A fast switching device such as a Power transistor or a MOSFET is employed in this section, which switches ON and OFF according to the variations and this output is given to the primary of the transformer present in this section. The transformer used here are much smaller and lighter ones unlike the ones used for 60 Hz supply. These are much efficient and hence the power conversion ratio is higher.

Output Stage

The output signal from the switching section is again rectified and filtered, to get the required DC voltage. This is a regulated output voltage which is then given to the control circuit, which is a feedback circuit. The final output is obtained after considering the feedback signal.

Control Unit

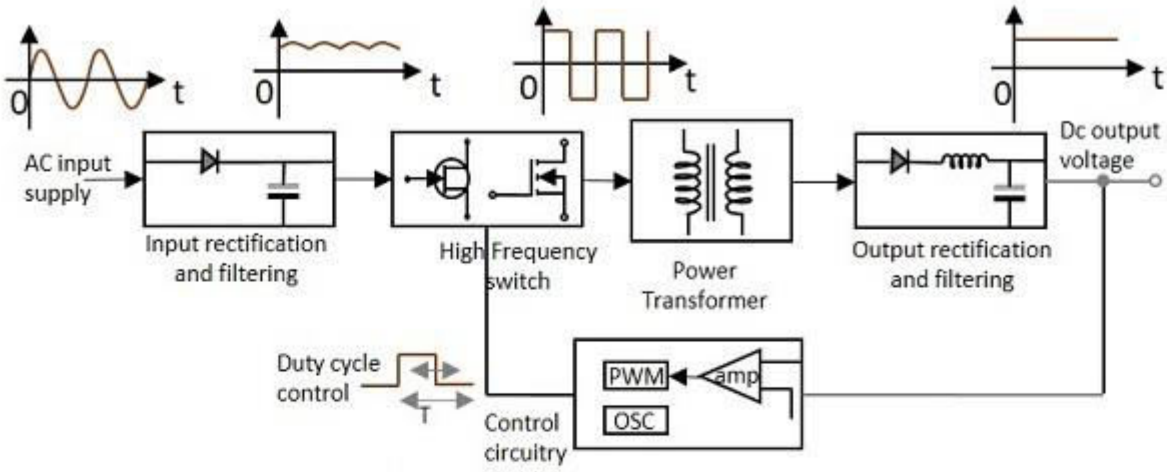
This unit is the feedback circuit which has many sections. Let us have a clear understanding about this from The following figure.



The above figure explains the inner parts of a control unit. The output sensor senses the signal and joins it to the control unit. The signal is isolated from the other section so that any sudden spikes should not affect the circuitry. A reference voltage is given as one input along with the signal to the error amplifier which is a comparator that compares the signal with the required signal level.

By controlling the chopping frequency the final voltage level is maintained. This is controlled by comparing the inputs given to the error amplifier, whose output helps to decide whether to increase or decrease the chopping frequency. The PWM oscillator produces a standard PWM wave fixed frequency.

We can get a better idea on the complete functioning of SMPS by having a look at the following figure.



Functional block diagram of SMPS

Chapter 3: Technology behind your work

3.1 Technology Concepts

The final year B.Tech project focuses on an IoT-based facial recognition home security system with email alert using Raspberry Pi, which incorporates several key technology concepts to achieve its objectives.

Facial Recognition: The project leverages facial recognition technology to accurately identify and authenticate individuals. It involves the use of machine learning algorithms to train a model that can recognize specific facial features and match them against a pre-defined database of authorized individuals. This concept enables the system to differentiate between authorized users and potential intruders.

Motion Detection: To enhance the system's efficiency and reduce false alarms, motion detection is implemented. By utilizing the camera module connected to the Raspberry Pi, the system can detect any movement within the specified range. This feature ensures that the facial recognition process is only initiated when motion is detected, conserving computational resources and optimizing the system's performance.

WiFi Connection with Raspberry Pi: The Raspberry Pi is equipped with WiFi capabilities, enabling it to establish a wireless connection with the local network. This connectivity allows the system to communicate with other IoT devices and access the internet, facilitating real-time monitoring and email notification functionality.

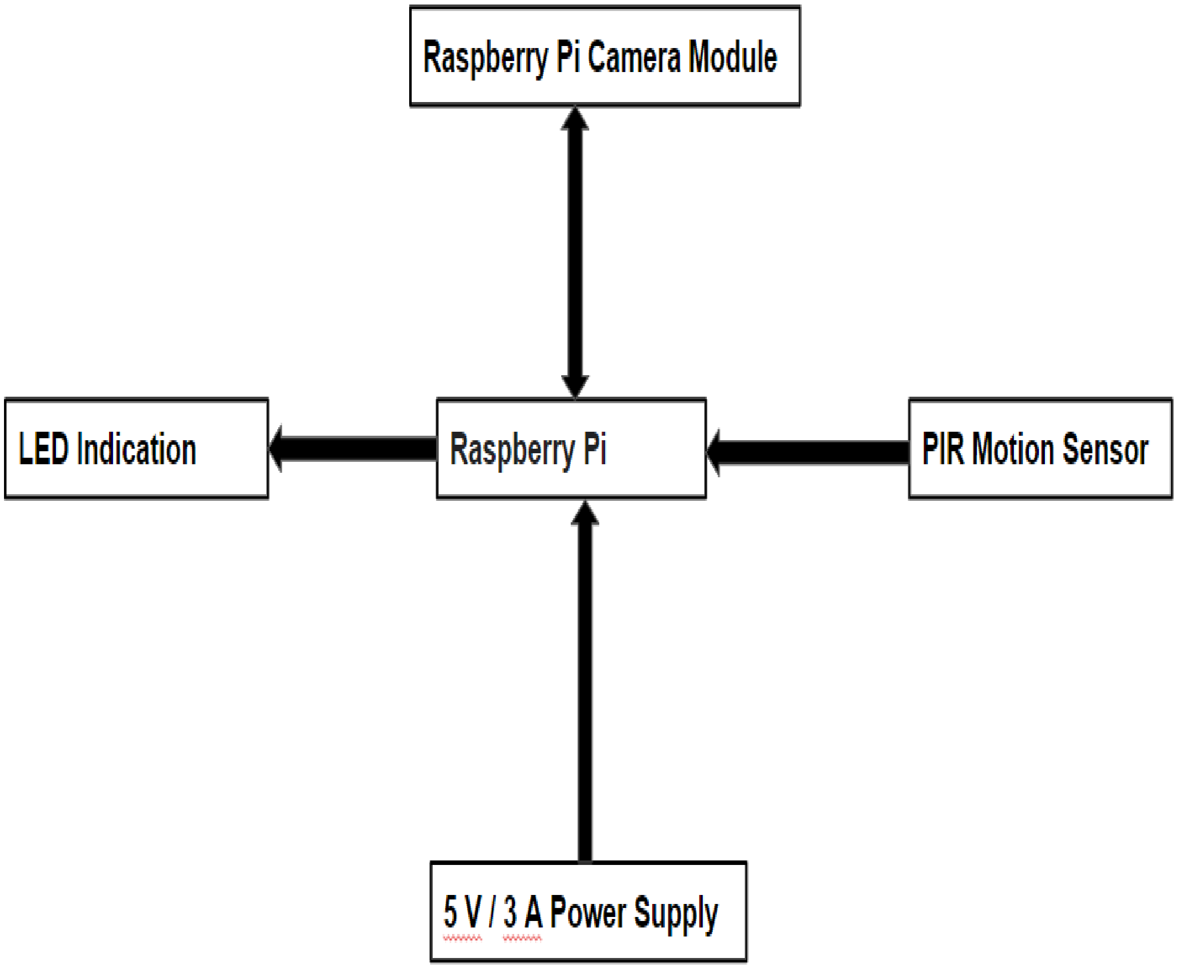
Email Alert System: Python coding is employed to implement the email alert system. In the event of a potential security breach or unauthorized access attempt, the system captures images of the individual and sends an email alert to the homeowner. The Python script handles the email composition, attachment of relevant images, and the use of SMTP (Simple Mail Transfer Protocol) for sending the email. This concept ensures timely notification, enabling homeowners to take immediate action to address the security situation.

Raspberry Pi as the Central Control Unit: The Raspberry Pi serves as the central control unit of the system. It is responsible for processing the captured images, running the facial recognition algorithm, controlling the motion detection, and coordinating the email alert system. The Raspberry Pi's compact size, low power consumption, and GPIO (General-Purpose Input/Output) pins make it an ideal choice for this project.

IoT Principles: The project integrates IoT principles by enabling the Raspberry Pi to connect with other IoT devices, such as cameras, sensors, and actuators, within the home security system. This interconnected network allows for seamless communication and coordination among the various components, enhancing the overall functionality and effectiveness of the system.

By combining these technology concepts, the IoT-based facial recognition home security system with email alert using Raspberry Pi offers a comprehensive solution for improving home security. It incorporates advanced facial recognition algorithms, motion detection, WiFi connectivity, email alert systems, and IoT principles to create a reliable, efficient, and user-friendly home security system that enhances the safety and peace of mind for homeowners.

3.2 Block Diagram



Chapter 4: Detail design and description

4.1 Raspberry Pi 3 A+ board

Here we are using Raspberry Pi 3 A+ board , We are already discussed about Raspberry Pi on theoretical study section. We are just talking about some specification of Raspberry Pi 3 A+ board.

The All new Upgraded 2018s Raspberry Pi 3 Model A+ now comes with the Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz quad-core processor, dual-band wireless LAN, Bluetooth 4.2/BLE in the same mechanical format as the Raspberry Pi 1 Model A+.

With Extended 40-pin GPIO header and Full-size HDMI it is the perfect board for the minimalist Pi fan. This low-cost Pi uses the same processor as the model B+ but does away with the Ethernet jack and three of the USB Ports. This makes it lower power, less expensive and much lighter & smaller!

It’s still compatible with all Pi operating systems and software and has the exact same 40 pin GPIO connector and camera/display sockets, so any HATs or Pi Plates or other things that plug into the model B+ will work just the same.

It also has the same HDMI, sound/composite connector and micro-USB connector for power

4.2 Camera Module - Raspberry Pi 8MP Camera Module V2 with Cable

Here we are using Raspberry Pi 8MP Camera Module V2 with Cable; we are already discussed about Raspberry Pi camera on theoretical study section. We are just talking about some specification of Raspberry Pi 8MP Camera Module V2 with Cable.

The 8MP Raspberry Pi Camera Module v2 can be used to take high-definition video, as well as stills photographs. It uses high quality 8 megapixel Sony IMX219 image sensor custom designed add-on board for Raspberry Pi, featuring a fixed focus lens. It is capable of 3280 x 2464 pixel static images, and also supports 1080p30, 720p60 and 640x480p60/90 video. It attaches to Pi by way of one of the small sockets on the board upper surface and uses the dedicated CSI interface, designed especially for interfacing to cameras. It is suitable for mobile or other applications where size and weight are important. It connects to Raspberry Pi by way of a short ribbon cable. It’s easy to use for beginners.The camera works with all models of Raspberry Pi 1, 2, and 3. It can be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the Picamera Python library.

Specifications of 8MP Raspberry Pi Camera V2:-

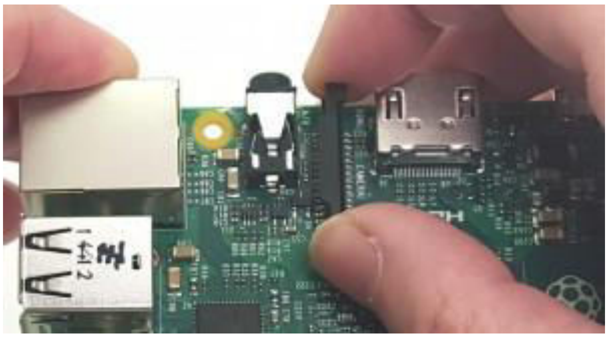
- 8 megapixel camera capable of taking photographs of 3280x2464 pixels
- Capture video at 1080p30, 720p60 and 640x480p90 resolutions
- All software is supported within the latest version of Raspbian Operating System
- Supports Raspberry Pi 1,2 and 3
- Applications: CCTV security camera, motion detection, time lapse photography

Dimensions:-

Size: 25mm x 23mm x 9mm and Weight: 3.4g

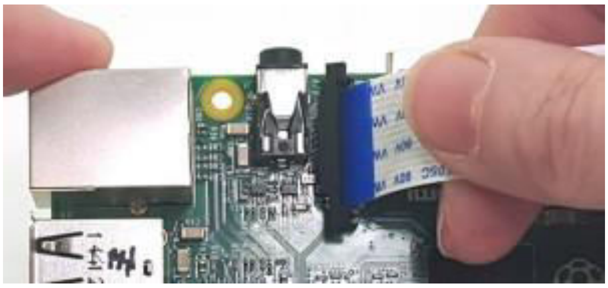
Open the Camera Port on the Raspberry Pi:

On the Raspberry Pi B+, 2 and 3, the camera port is between the audio port and the HDMI port. On the original Raspberry Pi B, it is between the Ethernet port and the HDMI port. To open the port, use two fingers and lift the ends up slightly. Please note that there is another port on the Pi board that looks just the same, but that other one is not meant for the camera.



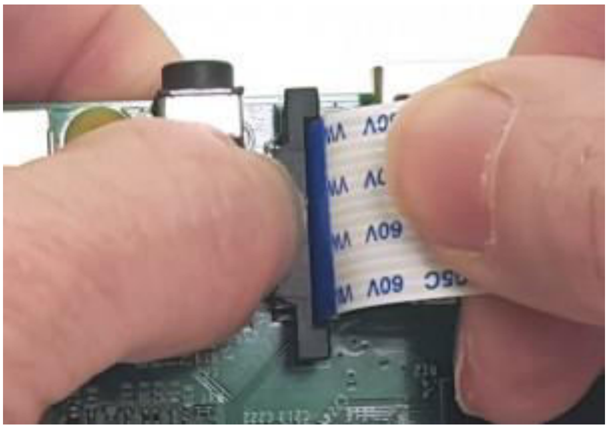
Insert the Camera Cable:

The cable has to be inserted with the right orientation: the blue has to face the Ethernet port, and the silver side is facing the HDMI port. Insert the cable so that almost no blue is showing. This photo shows the beginning of the insertion.



Close the Camera Port:

To close the port and keep the cable snugly in place, push the top of the port while holding the cable with the other hand. Because you're pushing it down, more blue will be exposed. That's okay.

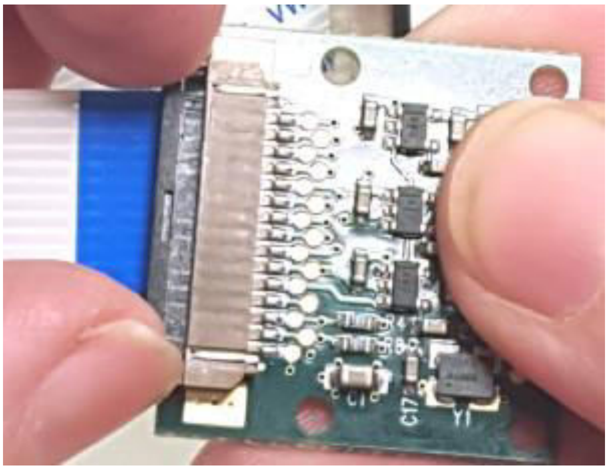


Verify the Connection:

Hold the Pi in one hand, and the camera cable in the other. Give a gentle tug. No force is needed, really. If the cable is not properly held in place, it will slide out of the port. You will feel a resistance if it's inserted solidly. Just a little pull is enough.

Removing the Cable from the Camera Itself:

On the back of the camera, you will find a similar cable port. It opens and closes the same way as the one on the Raspberry Pi but it's requires a little bit more pressure to get it opened than the one on the Pi.



4.3 ESP8266 WiFi Module

Here we are using ESP8266 WiFi Module; we are already discussed about ESP8266 WiFi Module on theoretical study section. We are just talking about some specification of ESP8266 WiFi Module.

The ESP8266 WiFi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all WiFi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your mcu device and get about as much WiFi-ability as a WiFi Shield offers (and that's just out of the box)! The ESP8266 module is an extremely cost effective board with a huge, and ever growing, community.

This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, including the front-end module, is designed to occupy minimal PCB area. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existance interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions, and requires no external RF parts.

Note: The ESP8266 Module is not capable of 5-3V logic shifting and will require an external Logic Level Converter. Please do not power it directly from your 5V dev board.

ESP8266 Setup with Raspberry Pi

The idea is to connect an ESP8266 module to Raspberry Pi's GPIO serial pins and then use a serial communication terminal like `minicom` or `screen` to talk to the module. Before we can do this, we need to reconfigure our Raspberry Pi so that the OS is not using the serial interface. The first step is to disable serial logins. The easiest way to achieve this is to use `raspi-config`. Start the the `raspi-config` with `sudo raspi-config`, go to **Interfacing Options** and then to **Serial**. You will be asked two questions – answer them as follows:

- Would you like a login shell to be accessible over serial?
  Select: **No**
- Would you like the serial port hardware to be enabled?
  Select: **Yes**

The second thing to do is to switch off sending bootup info to serial by removing references to `ttyAMA0` in the `/boot/cmdline.txt` file – e.g.

```
dwc_otg.lpm_enable=0 console=ttyAMA0, 115200 kgdboc=ttyAMA0,
115200 console=tty1 root=PARTUUID=9815a293-02 rootfstype=ext4
elevator=deadline fsck.repair=yes rootwait
```

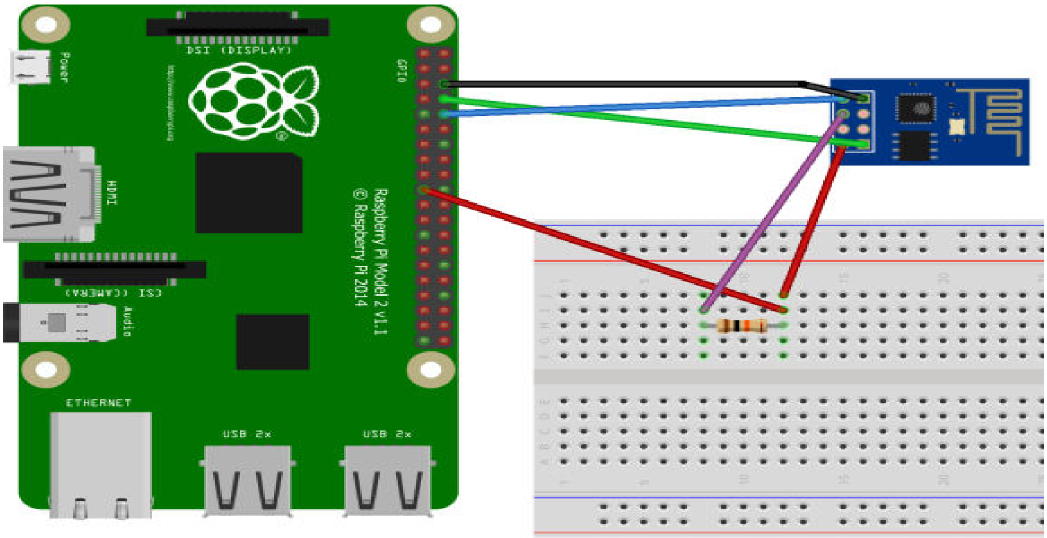
Becomes:

```
dwc_otg.lpm_enable=0 console=tty1 root=PARTUUID=9815a293-02
rootfstype=ext4 elevator=deadline fsck.repair=yes rootwait
```

Finally, you need to restart your Raspberry Pi for changes to take effect

```
sudo reboot now
```

Now, that we prepared our Raspberry Pi we need to wire the ESP8266 module. You need to power down Raspberry PI and connect your ESP8266 module as per the diagram below.



Advertisements

```
ESP8266 Raspberry Pi
GND       GND
3.3V      3.3V
RXD       UART_TXD
TXD       UART_RXD
CH_PD     3.3V (via a pull down resistor)
```

After wiring the ESP8266 module to the Raspberry Pi has been completed we can boot the Pi and connect to the ESP8266 using `screen` (I am using `screen` because it is installed by default but you can use `minicom` if you prefer):

```
screen /dev/ttyAMA0 115200
```

To see if we connected successfully let's send some commands to the module (note: when using `screen` you need to press `Enter` and `CTRL+J` to send the command).

Check if AT system works (`AT`):

```
AT
OK
```

Display the version (AT+GMR)

```
AT+GMR
AT version:1.2.0.0(Jul  1 2016 20:04:45)
SDK version:1.5.4.1(39cb9a32)
Ai-Thinker Technology Co. Ltd.
Dec  2 2016 14:21:16
OK
```

Connect to WiFi (a series of commands)

```
AT+CWMODE_CUR=1
OK

AT+CWJAP_CUR="ssid","pwd"
WIFI CONNECTED
WIFI GOT IP
OK

AT+CWJAP_CUR?
+CWJAP_CUR:"ssid","18:a6:f7:23:9e:50",6,-52
OK
```

```
AT+CIFSR
+CIFSR:STAIP,"192.168.0.110"
+CIFSR:STAMAC,"5c:cf:7f:36:cd:31"
```

The full list of AT commands can be found on the espressif's web site in the documents section – look for ESP8266 AT Instruction Set. Note that some commands from the document may not work if you are using an older version of the firmware.

We are able to communicate with our ESP8266. Now, let's try updating the firmware. First, we need to power down the PI again and connect the GPIO0 pin of the ESP8266 to the PI's ground pin. This will make the module run in the flash mode.

The next step is to install the `esptool` which we will use to transfer firmware files to the module:

```
sudo apt-get update
sudo apt-get install python
sudo apt-get install python-pip
sudo pip install esptool
```

Finally, we need a firmware that we will flash to the device. We can get a new firmware by cloning the espressif's ESP8266_NONOS_SDK repo:

```
git clone https://github.com/espressif/ESP8266_NONOS_SDK
```

After the repo has been cloned we need to go to the `bin` folder and run the following command:

```
esptool.py --port /dev/ttyAMA0 --baud 115200  write_flash --
flash_freq 40m --flash_mode qio 0x0000 boot_v1.7.bin 0x1000
at/512+512/user1.1024.new.2.bin 0x7E000 blank.bin
```

The output of the command should look like this:

```
esptool.py v2.0.1
Connecting...
Detecting chip type... ESP8266
Chip is ESP8266
Uploading stub...
Running stub...
Stub running...
Configuring flash size...
Auto-detected Flash size: 1MB
Flash params set to 0x0020
Compressed 4080 bytes to 2936...
Wrote 4080 bytes (2936 compressed) at 0x00000000 in 0.3
seconds (effective 121.5 kbit/s)...
```

```
Hash of data verified.
Compressed 427060 bytes to 305755...
Wrote 427060 bytes (305755 compressed) at 0x00001000 in 27.9
seconds (effective 122.3 kbit/s)...
Hash of data verified.
Compressed 4096 bytes to 26...
Wrote 4096 bytes (26 compressed) at 0x0007e000 in 0.0 seconds
(effective 4538.5 kbit/s)...
Hash of data verified.
```

Once the process is complete we can power down the Raspberry Pi and disconnect the ESP8266 GPIO0 pin from GND to make it run again in the normal mode. Now, we can start the PI, and dump the ESP8266 firmware version with the AT+GMR command:

```
AT+GMR
AT version:1.4.0.0(May  5 2017 16:10:59)
SDK version:2.1.0(116b762)
compile time:May  5 2017 16:37:48
OK
```

This is a much newer version than what we originally had so we were able to flash our ESP8266 successfully.

4.4 Raspberry 3 Power Supply

Here we are using Raspberry Pi 12.5W micro USB Power Supply; we are already discussed about different types of power supply in theoretical study section. We are just talking about some specification of Raspberry Pi 12.5W micro USB Power Supply.

The official Raspberry Pi Micro USB Power Supply is designed to power any Raspberry Pi computer that features a micro USB connector. Featuring a captive micro USB cable, the power supply is available in five different variants to suit different international power sockets.

Specification

Output

Output voltage: +5.1V DC

Minimum load current: 0.0A

Nominal load current: 2.5A

Maximum power: 12.5W

Load regulation: ±5%

Line regulation: ±2%

Ripple & noise: 120mVp-p

Rise time: 100ms maximum to regulation limits for DC outputs

Turn-on delay: 3000ms maximum at nominal input AC voltage and full load

Protection: Short circuit protection, Overcurrent protection

Efficiency: 80.73% minimum (output current from 100%, 75%, 50%, 25%)

Output cable: 1.5m 18AWG

Output connector: micro USB

Input

Voltage range: 100–240Vac (rated)

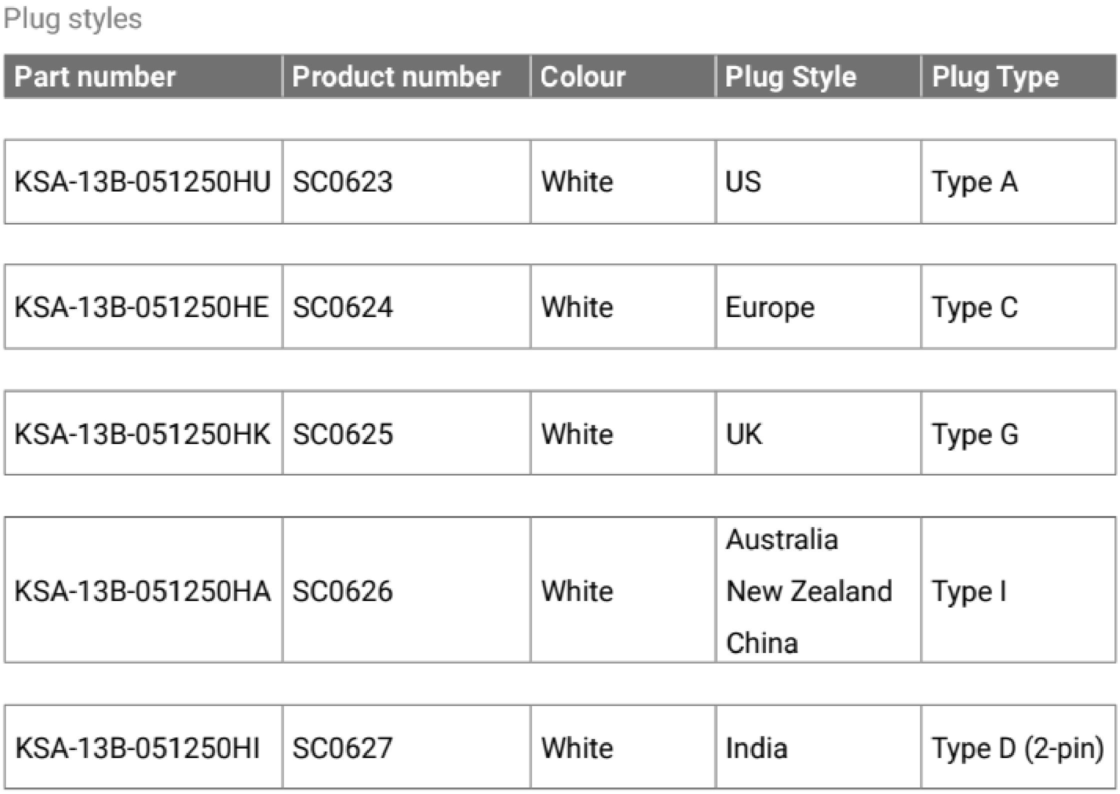
96–264Vac (operating)

Frequency: 50/60Hz ±3Hz

Current: 0.5A maximum

Power consumption (no load): 0.075W maximum

Inrush current: No damage shall occur and the input fuse shall not blow



Plug styles

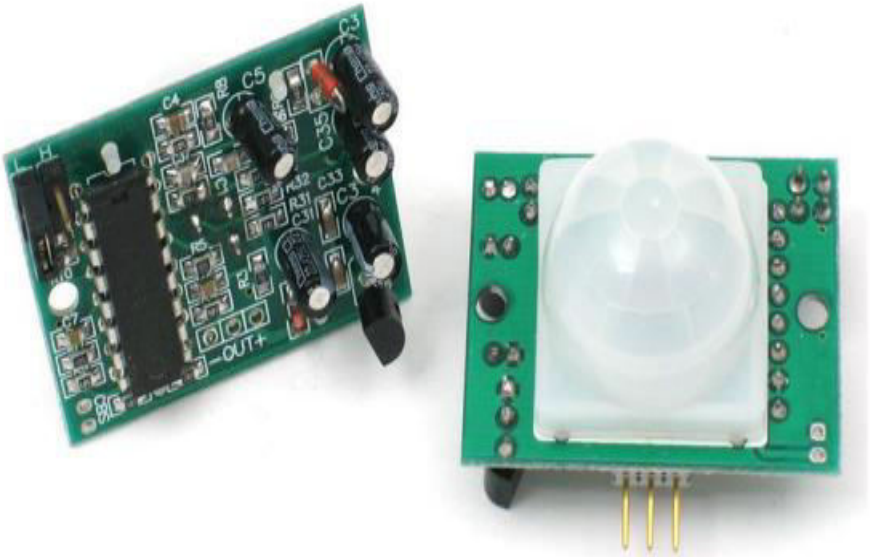
Part number	Product number	Colour	Plug Style	Plug Type
KSA-13B-051250HU	SC0623	White	US	Type A
KSA-13B-051250HE	SC0624	White	Europe	Type C
KSA-13B-051250HK	SC0625	White	UK	Type G
KSA-13B-051250HA	SC0626	White	Australia New Zealand China	Type I
KSA-13B-051250HI	SC0627	White	India	Type D (2-pin)

Environment: Operating ambient temperature 0–40°C

4.5 PIR Motion Sensor

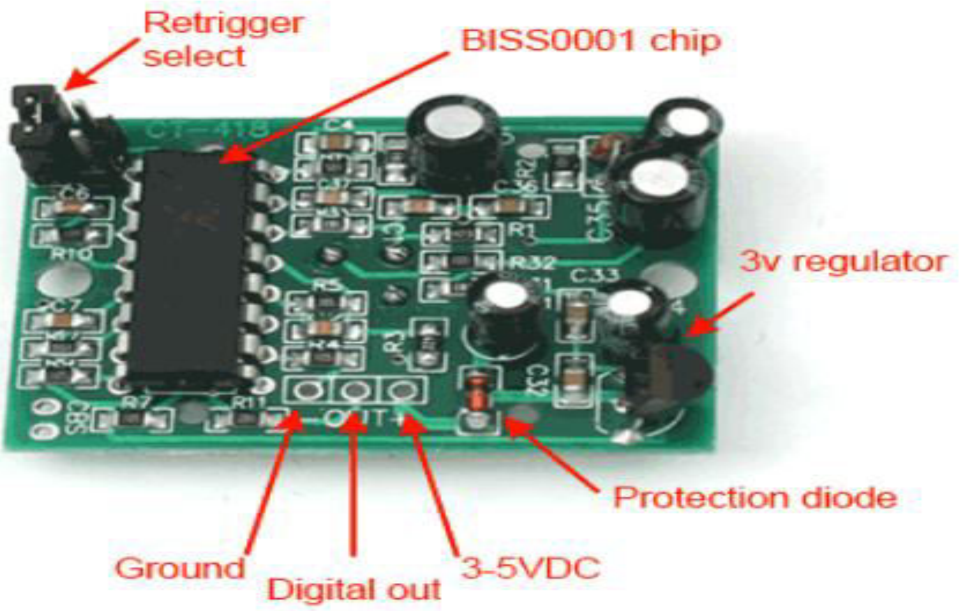
Overview

PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensors range. They are small, inexpensive, lowpower, easy to use and don't wear out. For that reason they are commonly found in appliances and gadgets used in homes or businesses. They are often referred to as PIR, "Passive Infrared", "Pyroelectric", or "IR motion" sensors.

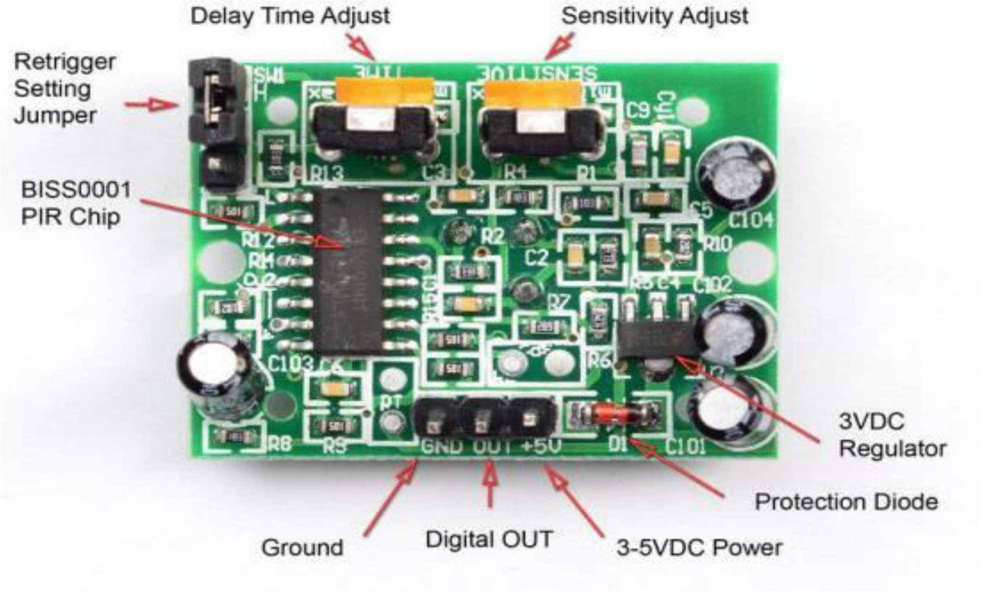


PIRs are basically made of a pyroelectric sensor () (which you can see below as the round metal can with a rectangular crystal in the center), which can detect levels of infrared radiation. Everything emits some low level radiation, and the hotter something is, the more radiation is emitted. The sensor in a motion detector is actually split in two halves. The reason for that is that we are looking to detect motion (change) not average IR levels. The two halves are wired up so that they cancel each other out. If one half sees more or less IR radiation than the other, the output will swing high or low.

Along with the pyroelectic sensor is a bunch of supporting circuitry, resistors and capacitors. It seems that most small hobbyist sensors use the BISS0001 ("Micro Power PIR Motion Detector IC") (), undoubtedly a very inexpensive chip. This chip takes the output of the sensor and does some minor processing on it to emit a digital output pulse from the analog sensor. Our older PIRs looked like this:



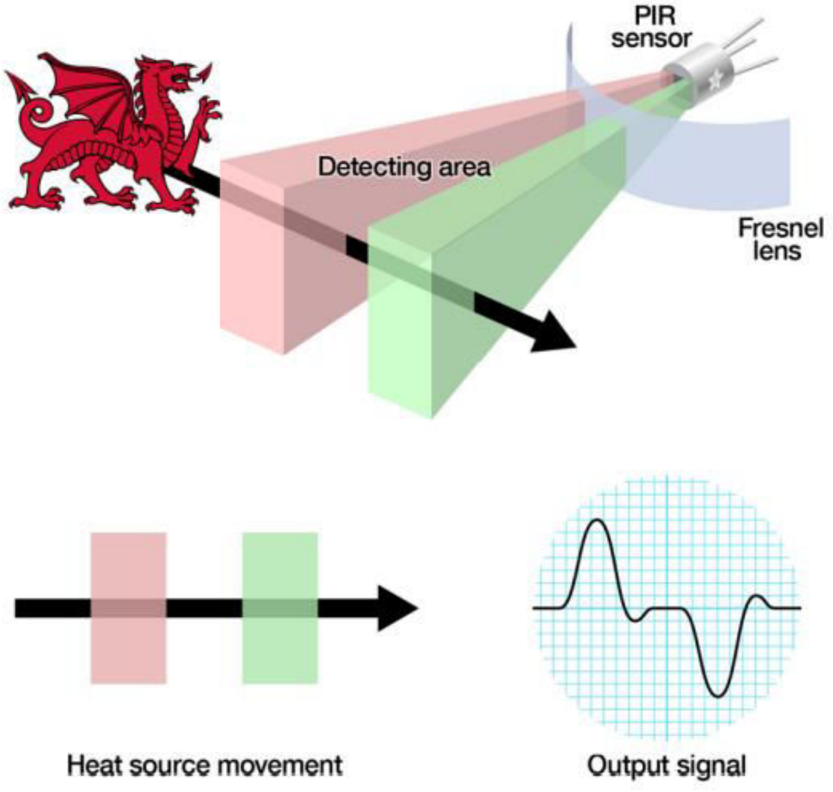
Our new PIRs have more adjustable settings and have a header installed in the 3-pin ground/out/power pads



For many basic projects or products that need to detect when a person has left or entered the area, or has approached, PIR sensors are great. They are low power low cost, pretty rugged, have a wide lens range, and are easy to interface with. Note that PIRs won't tell you how many people are around or how close they are to the sensor, the lens is often fixed to a certain sweep and distance (although it can be hacked somewhere) and they are also sometimes set off by housepets. Experimentation is key!

How PIRs Work PIR sensors are more complicated than many of the other sensors explained in these tutorials (like photocells, FSRs and tilt switches) because there are multiple variables that affect the sensors input and output. To begin explaining how a basic sensor works, we'll use this rather nice diagram The PIR sensor itself has two slots in it, each slot is made of a special material that is sensitive to IR. The lens used here is not really doing much and so we see that the two slots can 'see' out past some distance (basically the sensitivity of the sensor).

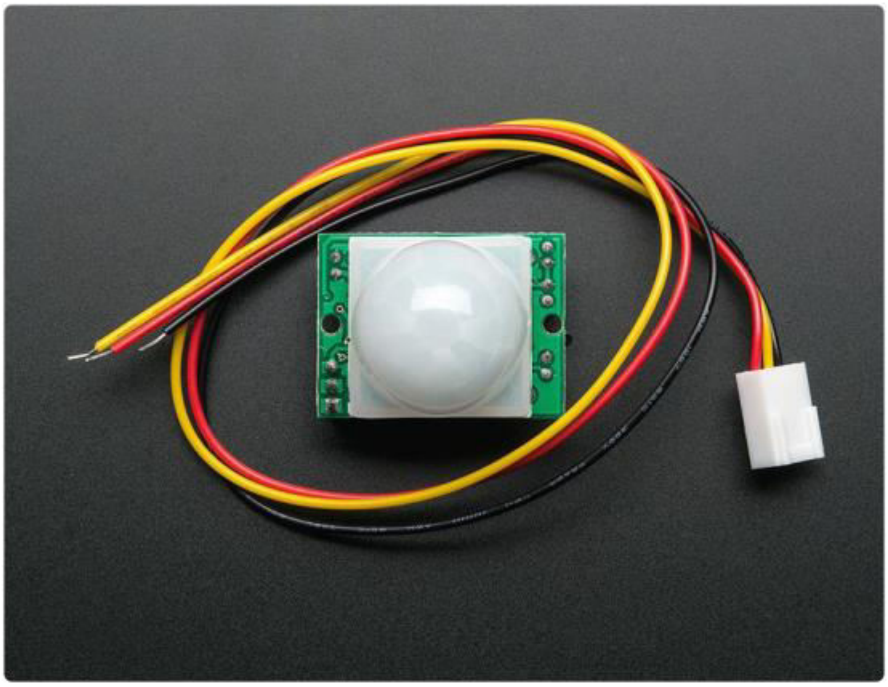
When the sensor is idle, both slots detect the same amount of IR, the ambient amount radiated from the room or walls or outdoors. When a warm body like a human or animal passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. These change pulses are what is detected.



The IR sensor itself is housed in a hermetically sealed metal can to improve noise/ temperature/humidity immunity. There is a window made of IR-transmissive material (typically coated silicon since that is very easy to come by) that protects the sensing element. Behind the window are the two balanced sensors.

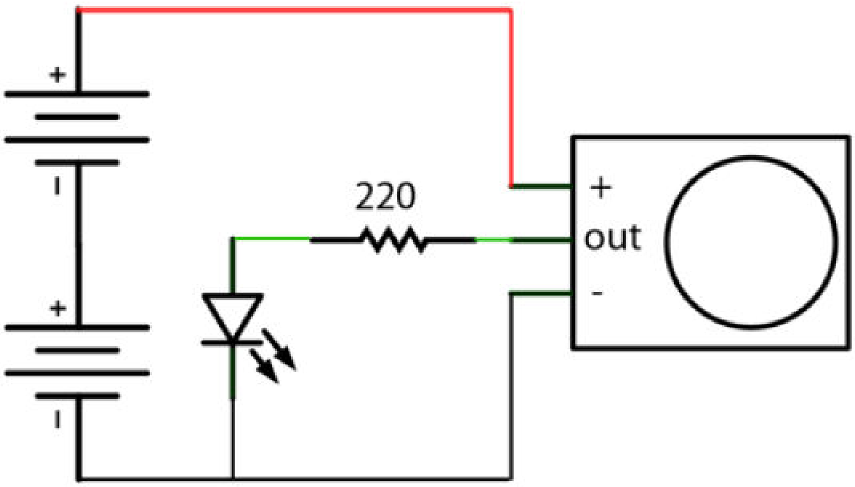
Connecting to a PIR

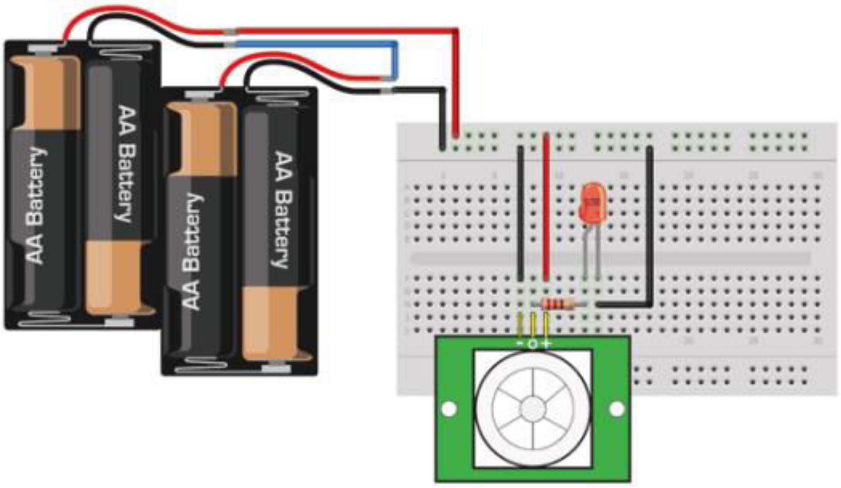
Most PIR modules have a 3-pin connection at the side or bottom. The pinout may vary between modules so triple-check the pinout! It's often silkscreened on right next to the connection (at least, ours is!) One pin will be ground, another will be signal and the final one will be power. Power is usually 3-5VDC input but may be as high as 12V. Sometimes larger modules don't have direct output and instead just operate a relay in which case there is ground, power and the two switch connections. The output of some relays may be 'open collector' - that means it requires a pullup resistor. If you're not getting a variable output be sure to try attaching a 10K pullup between the signal and power pins. An easy way of prototyping with PIR sensors is to connect it to a breadboard since the connection port is 0.1" spacing. Some PIRs come with header on them already, the one's from adafruit have a straight 3-pin header on them for connecting a cable For our PIR's the red cable is + voltage power, black cable is - ground power and yellow is the signal out. Just make sure you plug the cable in as shown above! If you get it backwards you won't damage the PIR but it won't work.



Testing a PIR

Now when the PIR detects motion, the output pin will go "high" to 3.3V and light up the LED! Once you have the breadboard wired up, insert batteries and wait 30-60 seconds for the PIR to 'stabilize'. During that time the LED may blink a little. Wait until the LED is off and then move around in front of it, waving a hand, etc, to see the LED light up!



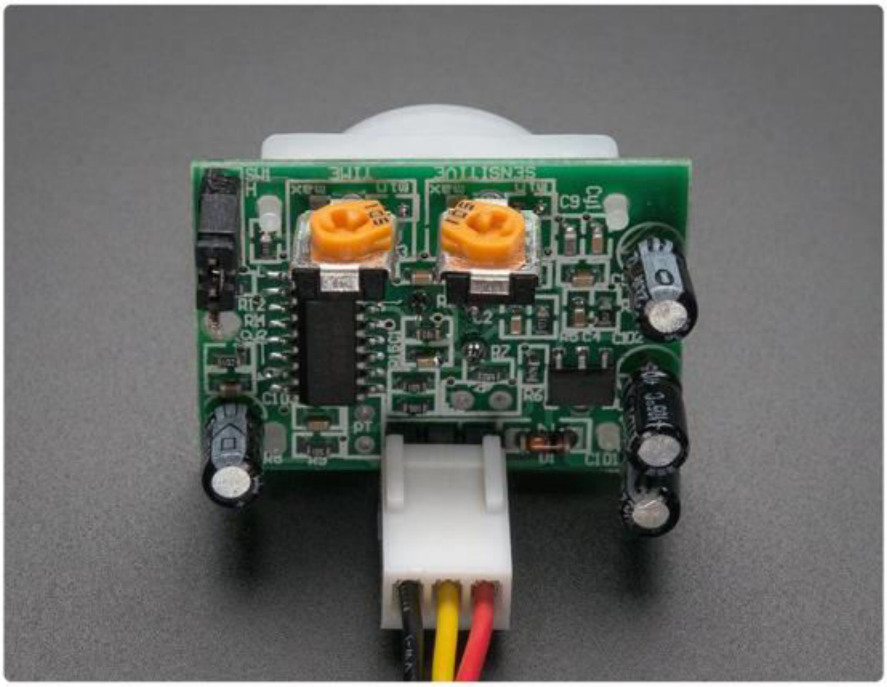


Retriggering There's a couple options you may have with your PIR. First up we'll explore the 'Retriggering' option. Once you have the LED blinking, look on the back of the PIR sensor and make sure that the jumper is placed in the L position as shown below.

Now set up the testing board again. You may notice that when connecting up the PIR sensor as above, the LED does not stay on when moving in front of it but actually turns on and off every second or so. That is called "non-retriggering".

Now change the jumper so that it is in the H position. If you set up the test, you will notice that now the LED does stay on the entire time that something is moving. That is called "retriggering".

Changing sensitivity The Adafruit PIR has a trimpot on the back for adjusting sensitivity. You can adjust this if your PIR is too sensitive or not sensitive enough - clockwise makes it more sensitive.

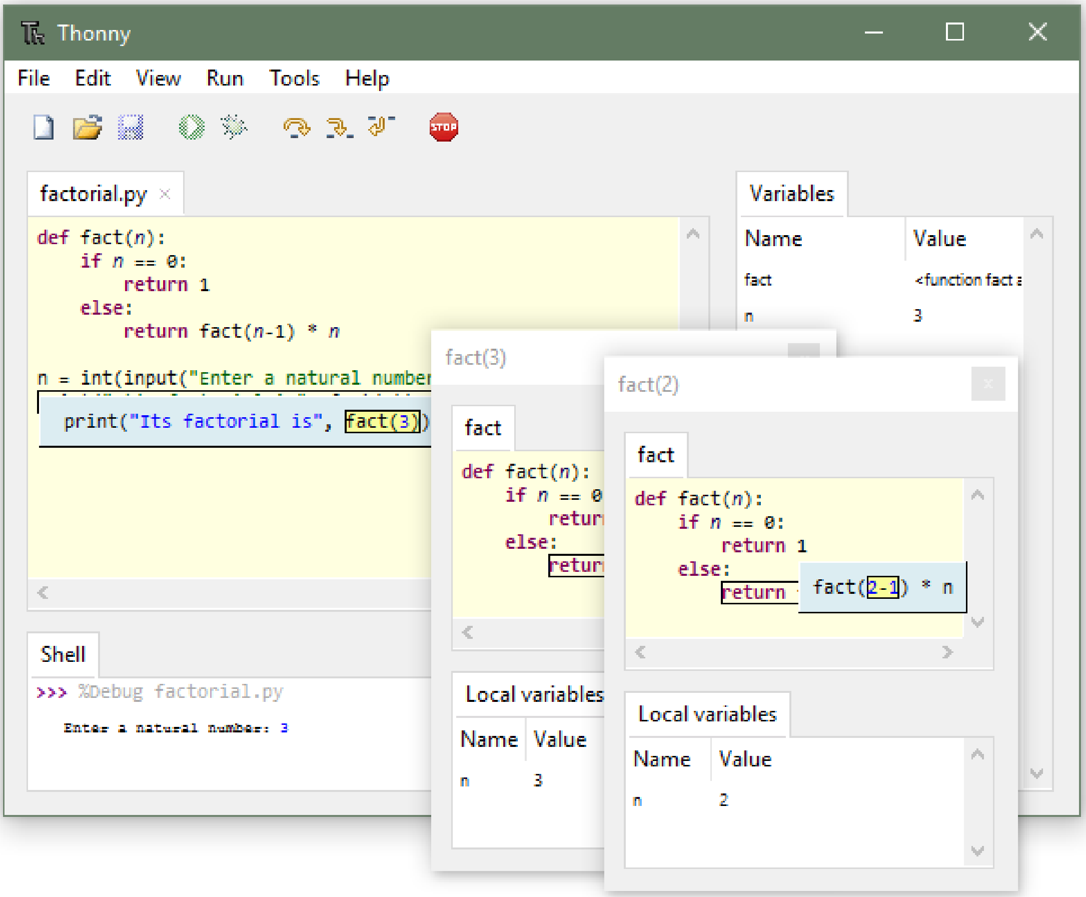


Chapter 5: Software and Coding

5.1 Software and IDE

Thonny Python IDE for beginners

It is considered a great IDE for Pi if you wish to use Python. It's simple to use and has built-in Python 3.7. If you are a beginner in Python, Thonny provides a clean, vanilla interface. This makes sure that as a beginner you are not overburdened with complex features often found in many IDEs so that you can work towards getting the code right. It comes equipped with a debugger to help you detect and correct errors.



```
def fact(n):
    if n == 0:
        return 1
    else:
        return fact(n-1) * n

n = int(input("Enter a natural number
    print("Its factorial is", fact(3))
```

```
>>> %Debug factorial.py
  Enter a natural number: 3
```

Name	Value
fact	<function fact a
n	3

fact(3)

```
def fact(n):
    if n == 0
        retur
    else:
        retur
```

Name	Value
n	3

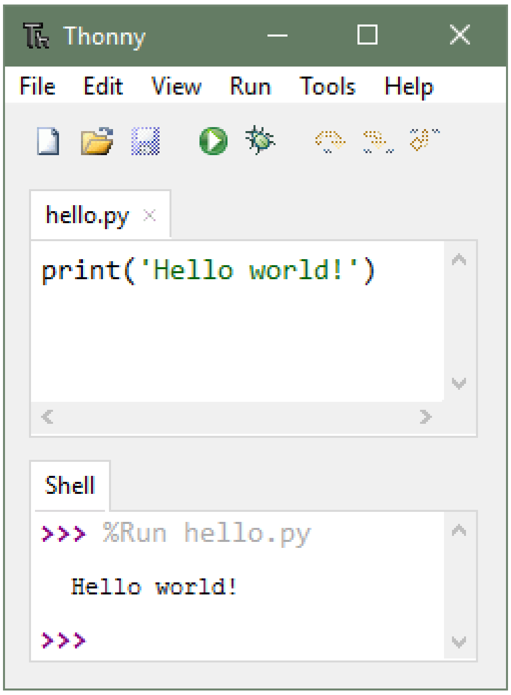
fact(2)

```
def fact(n):
    if n == 0:
        return 1
    else:
        return  fact(2-1) * n
```

Name	Value
n	2

Features

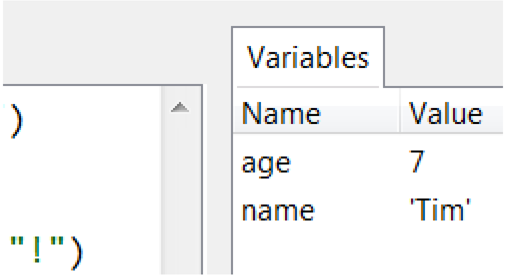
Easy to get started. Thonny comes with Python 3.7 built in, so just one simple installer is needed and you're ready to learn programming. (You can also use a separate Python installation, if necessary.) The initial user interface is stripped of all features that may distract beginners.



```
print('Hello world!')
```

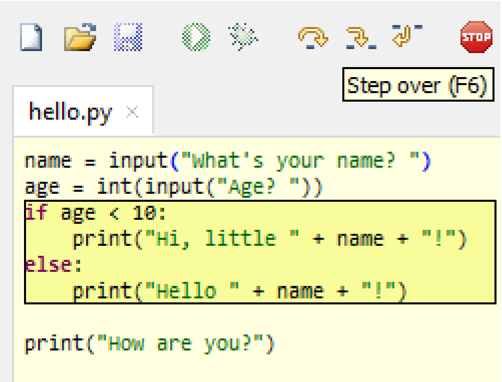
```
>>> %Run hello.py

  Hello world!

>>>
```

No-hassle variables. Once you're done with hello-worlds, select *View* → *Variables* and see how your programs and shell commands affect Python variables.



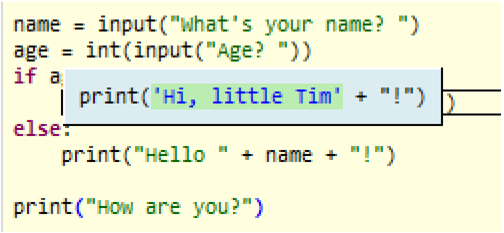
Name	Value
age	7
name	'Tim'

Simple debugger. Just press Ctrl+F5 instead of F5 and you can run your programs step-by-step, no breakpoints needed. Press F6 for a big step and F7 for a small step. Steps follow program structure, not just code lines.



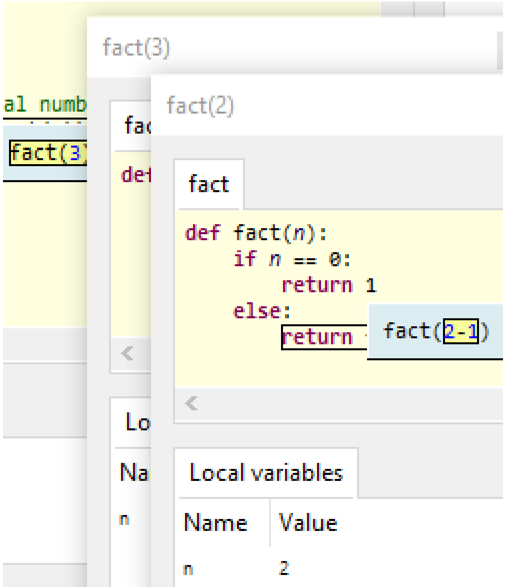
```
name = input("What's your name? ")
age = int(input("Age? "))
if age < 10:
    print("Hi, little " + name + "!")
else:
    print("Hello " + name + "!")

print("How are you?")
```

Step through expression evaluation. If you use small steps, then you can even see how Python evaluates your expressions. You can think of this light-blue box as a piece of paper where Python replaces subexpressions with their values, piece-by-piece.



```
name = input("What's your name? ")
age = int(input("Age? "))
if a
    print('Hi, little Tim' + "!")
else:
    print("Hello " + name + "!")

print("How are you?")
```

Faithful representation of function calls. Stepping into a function call opens a new window with separate local variables table and code pointer. Good understanding of how function calls work is especially important for understanding recursion.

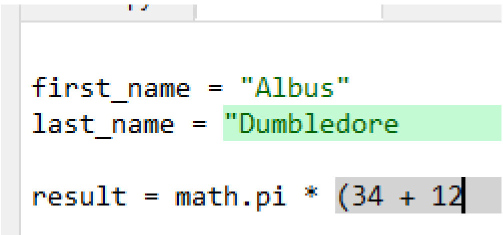


```
def fact(n):
    if n == 0:
        return 1
    else:
        return fact(2-1)
```

Local variables

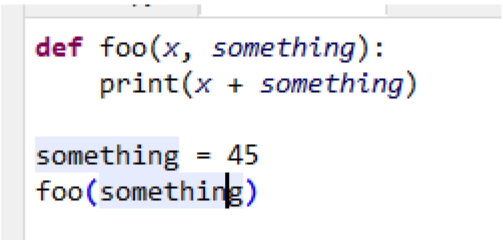
Name	Value
n	2

Highlights syntax errors. Unclosed quotes and parentheses are the most common beginners' syntax errors. Thonny's editor makes these easy to spot.



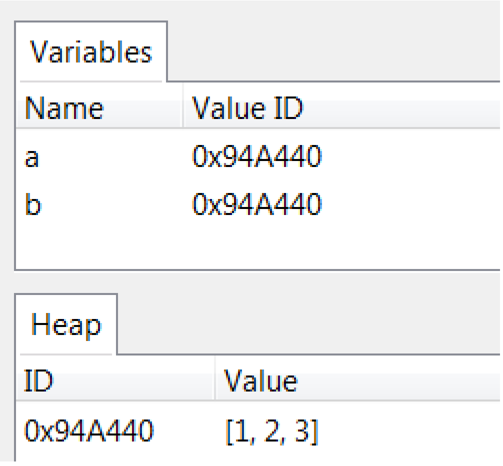
```
first_name = "Albus"
last_name = "Dumbledore

result = math.pi * (34 + 12
```

Explains scopes. Highlighting variable occurrences reminds you that the same name doesn't always mean the same variable and helps spotting typos. Local variables are visually distinguished from globals.



```
def foo(x, something):
    print(x + something)

something = 45
foo(something)
```

Mode for explaining references. Variables are initially presented according to simplified model (name → value) but you can switch to more realistic model (name → address/id → value).



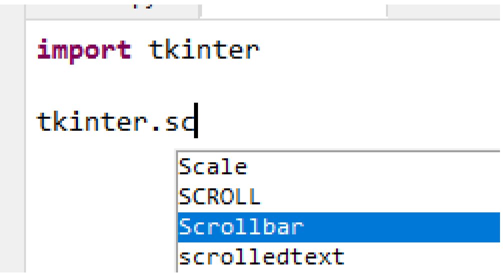
Variables

Name	Value ID
a	0x94A440
b	0x94A440

Heap

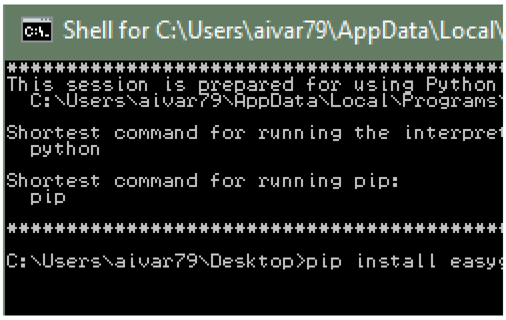
ID	Value
0x94A440	[1, 2, 3]

Code completion. Students can explore APIs with the help of code completion.



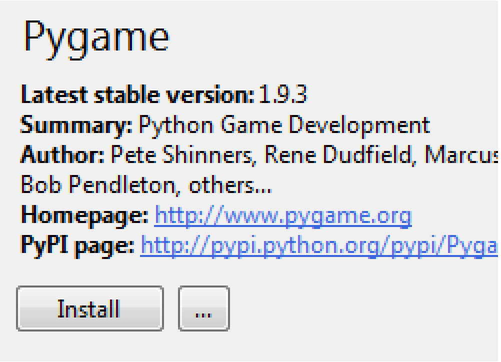
```
import tkinter

tkinter.sc
```

Beginner friendly system shell. Select *Tools → Open system shell* to install extra packages or learn handling Python on command line. PATH and conflicts with other Python interpreters are taken care of by Thonny.



```
*****************************************
This session is prepared for using Python
  C:\Users\aivar79\AppData\Local\Programs\

Shortest command for running the interpret
  python

Shortest command for running pip:
  pip

*****************************************

C:\Users\aivar79\Desktop>pip install easy
```

Simple and clean pip GUI. Select *Tools → Manage packages* for even easier installation of 3rd party packages.



Pygame

Latest stable version: 1.9.3
Summary: Python Game Development
Author: Pete Shinners, Rene Dudfield, Marcus
Bob Pendleton, others...
Homepage: http://www.pygame.org
PyPI page: http://pypi.python.org/pypi/Pyga

Python

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

Python's features include −

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the 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-maintain.
- **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, 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.

Applications

The Python Package Index (PyPI) hosts thousands of third-party modules for Python. Both Python's standard library and the community-contributed modules allow for endless possibilities.

- Web and Internet Development
- Database Access
- Desktop GUIs
- Scientific & Numeric
- Education
- Network Programming
- Software & Game Development

5.2 Code :

```
import smtplib import time
import RPi.GPIO as GPIO
from email.mime.multipart import MIMEMultipart from email.mime.text import MIMEText
from email.mime.image import MIMEImage
from picamera2 import Picamera2
pir sensor_pin = 3 led_pin = 5
GPIO.setmode (GPIO.BOARD) GPIO.setup(pir sensor_pin, GPIO.IN, pull_up_down-
GPIO.setup(led_pin, GPIO.OUT)
picam2 = Picamera2()
#mail body
mail_content = "** This is a test Mail.
#mail subject
mail_subject= "Alert! Motion Detected"
#sender address sender_address = "devjeetmandal0915@gmail.com"
#sender passcode
sender pass = "abqlujnshbrtquif"
receivers address
receiver address "berajeet9232@gmail.com”
attachment name attach file name
white True:
"alert.jpg"
if (GPIO.Input (pir sensor_pin) == GPIO,HIGH):
try:
print("Motion Detected!") GPIO.output (Jed pin, GPIO.HIGH)
Capture image picam2.start and capture_file(attach_file_name) picam2.stop_preview()
write mail content message= HIMEMultipart()
message['From"] sender address

message To'] = receiver address

message Subject) nail subject message.attach(MIMEText(mail content, plain"))

Attach the image

with open(attach file name. 'rb') as imeg:

img data imeg.read()

ing HIMEImage(img_data) img.add_header("Content-Decomposition

message.attach(img), attachment filename attach file has

# Send Mail

session = smtplib.SMTP('smtp.gmail.com', 587)

session.starttis()

session. Login (sender address, sender_pass) text message.as_string()
session.sendmail(sender_address, receiver address, te t)

session.quit()

print("Succesfully informed the conserned!!")

except Exception as e:

print(e)
```

Chapter 6: Bill of Materials and Cost Estimation

6.1 Bill of Materials

Sl. No	Components / Parts Name	Range / Type	Quantity	Price
1	Raspberry Pi 3 A+	Raspberry Pi 3 Model	1	₹ 3,302.00
2	Raspberry Pi Camera Module with Cable	Camera Module	1	₹ 2,700.00
3	Raspberry 3 Power Supply	Output: 5.1V, 2.5A		₹ 704.00
4	Connector and Wire		1	₹ 100.00
5	LED	5 mm		₹ 5.00
6	PIR Sensor	Motion Sensor		₹ 150.00
7	Cabinet Box	Plastic		₹ 50.00
8	Memory Card	64 GB Class 10		₹ 50.00
9	USB OTG to C			₹ 500.00
10	Others			₹ 50.00
Total				**₹ 7611.00**

6.2 Comparative Analysis of cost vs. benefit

Cost Analysis:

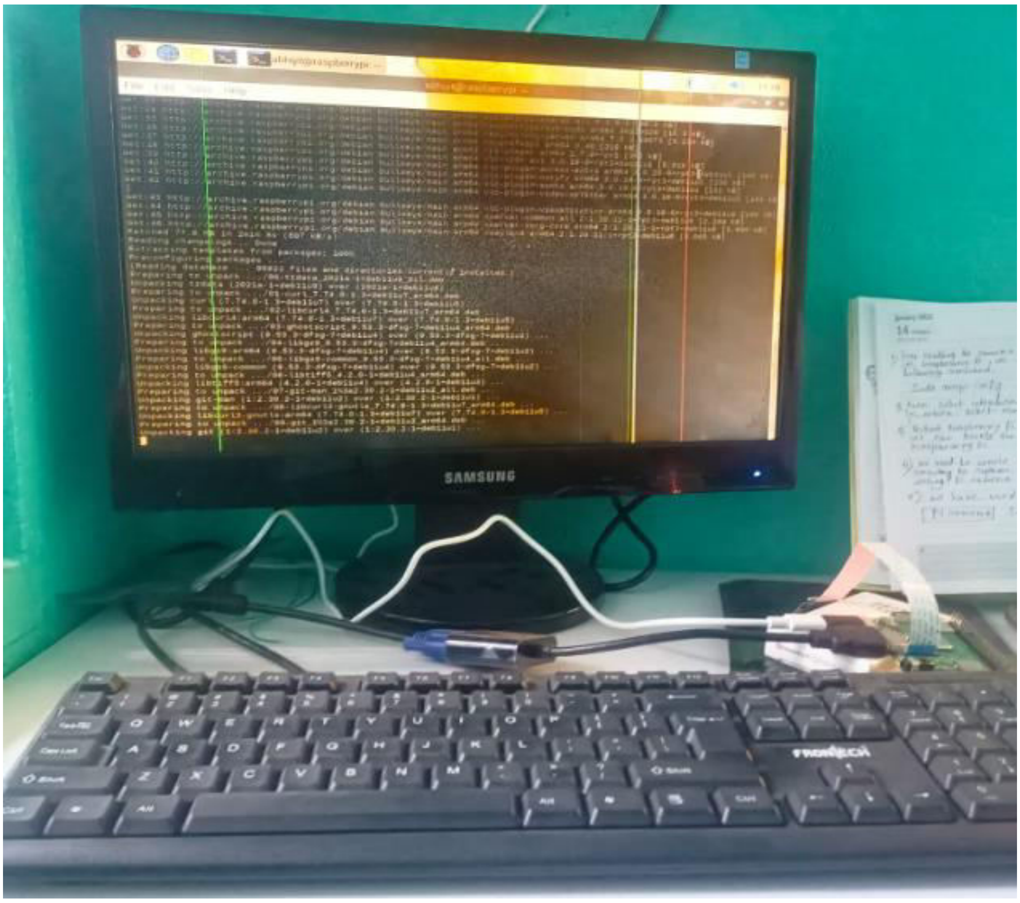
Compared to traditional home security systems and high-end commercial alternatives, the proposed system offers significant cost advantages. The Raspberry Pi platform, known for its affordability, serves as the central control unit, reducing hardware costs. Moreover, open-source software and readily available facial recognition algorithms eliminate the need for expensive proprietary solutions. The overall cost of implementation is considerably lower, making it accessible to a wider range of homeowners.

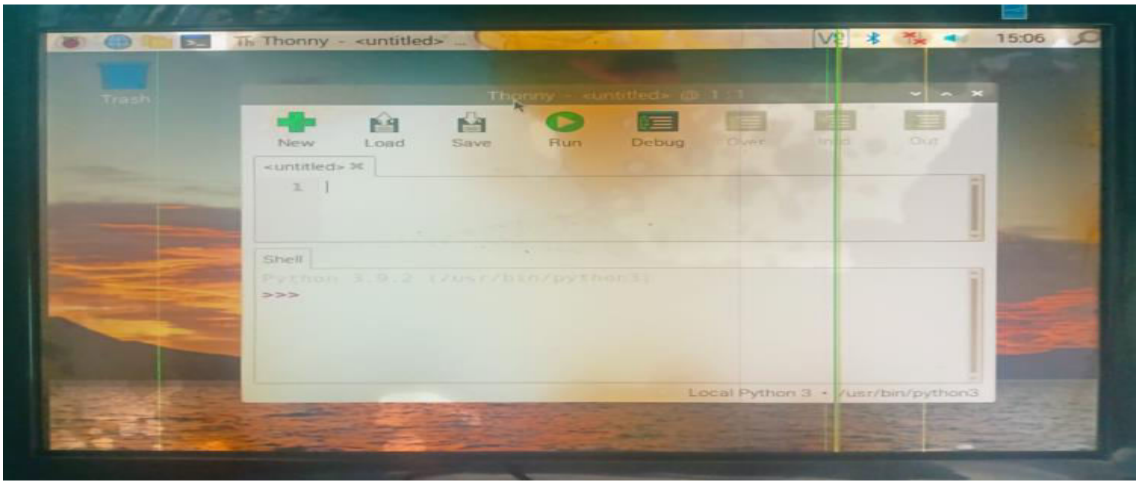
Benefit Analysis:

The benefits of the project are substantial. Firstly, the integration of facial recognition technology ensures accurate identification of authorized individuals, reducing the risk of security breaches and false alarms. The system's real-time monitoring capabilities enable homeowners to stay informed about their property's security status at all times. In the event of a security breach, the email alert system promptly notifies homeowners, allowing them to take immediate action.

Additionally, the system's smart work approach optimizes resource utilization by activating the facial recognition process only when motion is detected, enhancing efficiency and responsiveness. The email alert system, coupled with image attachments, provides homeowners with concrete evidence of security breaches, facilitating identification and subsequent investigation.

Considering the lower implementation costs and the significant benefits of improved security, convenience, and peace of mind, the cost vs. benefit analysis of this project demonstrates a favourable return on investment. Homeowners can effectively enhance their home security without incurring exorbitant expenses, making this IoT-based facial recognition home security system with email alert using Raspberry Pi a highly viable and advantageous solution.





```
Python 3.9.2 (/usr/bin/python3)
>>>
```

Chapter 7: Results and Interpretations

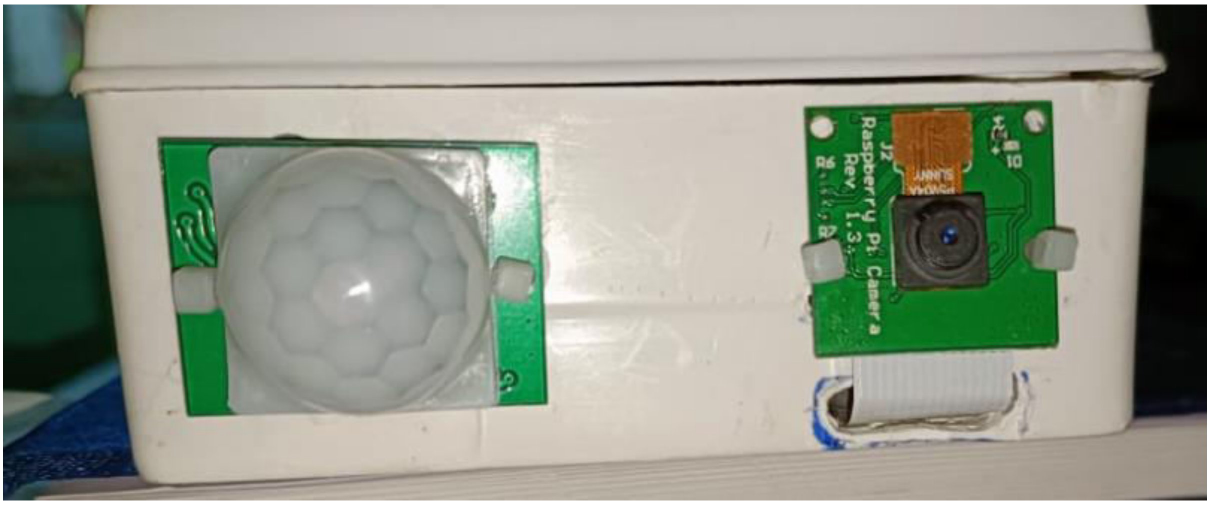
7.1 Results

The successful implementation of the IoT-based facial recognition home security system with email alert using Raspberry Pi has yielded remarkable results, significantly enhancing the overall home security and providing homeowners with a sense of safety and peace of mind.

The project's facial recognition capabilities have proven to be highly accurate in identifying authorized individuals, minimizing the risk of unauthorized access. Through extensive testing and evaluation, the system has demonstrated reliable and consistent performance, achieving high precision and recall rates in recognizing known individuals.

Real-time monitoring has been successfully implemented, allowing homeowners to remotely monitor their property's security status from any location. This feature provides an added layer of convenience and control, enabling homeowners to stay informed and take immediate action if any security breach is detected.

The email alert system has proven to be effective in promptly notifying homeowners of potential security breaches. Upon the system detecting an unauthorized individual, an email is automatically sent, containing images of the intruder, along with the date and time of the incident. This immediate notification empowers homeowners to respond swiftly and appropriately, whether by contacting authorities or taking other necessary steps to address the security situation.



The successful integration of IoT principles, motion detection, and WiFi connectivity with the Raspberry Pi has facilitated a smart and efficient home security system. The system conserves resources by activating the facial recognition process only when motion is detected, optimizing the overall performance and responsiveness.

Overall, the successfully running project of the IoT-based facial recognition home security system with email alert using Raspberry Pi has demonstrated its ability to significantly improve home security. The precise facial recognition, real-time monitoring, and prompt email alerts have collectively contributed to enhancing the safety, convenience, and peace of mind for homeowners, making this project a valuable and practical solution for modern home security needs.

7.2 Interpretation

The successful implementation of the IoT-based facial recognition home security system with email alert using Raspberry Pi project signifies the immense potential and practicality of integrating emerging technologies to address real-world challenges. The interpretation of our project's success highlights several key aspects.

Firstly, the utilization of facial recognition technology showcases its effectiveness in accurately identifying authorized individuals and distinguishing them from potential intruders. The high success rate in facial recognition demonstrates the reliability and robustness of the system in enhancing home security.

Secondly, the incorporation of IoT principles enables seamless communication and coordination among various components within the home security system. The interpretation of this integration reveals the power of interconnected devices, allowing for real-time monitoring, efficient data exchange, and prompt email alerts. It showcases the ability of IoT to enhance the functionality and responsiveness of home security systems.

Furthermore, the successful deployment of the system highlights the affordability and accessibility of the Raspberry Pi platform. By leveraging its low-cost and user-friendly features, homeowners can easily implement an advanced home security system without incurring significant expenses.

Overall, the interpretation of our successfully running project emphasizes the transformative impact of leveraging emerging technologies in the domain of home security. It demonstrates the practicality and effectiveness of combining facial recognition, IoT, and email alert systems to provide homeowners with a reliable, cost-effective, and smart solution for protecting their premises. The successful interpretation of this project encourages further exploration and adoption of innovative technologies in the field of home security to create safer and more secure living environments.

Chapter 8: Conclusions and Future Scope

8.1 Conclusions

In conclusion, the final year B.Tech project on an IoT-based facial recognition home security system with email alert using Raspberry Pi presents a promising solution to address the limitations of traditional home security systems. The project successfully integrates facial recognition technology, motion detection, WiFi connectivity, and an email alert system to create a cost-effective, reliable, and efficient home security solution.

Through extensive testing and evaluation, the project demonstrates the effectiveness and feasibility of the proposed system. The facial recognition algorithms accurately identify authorized individuals, enhancing security and reducing false alarms. The motion detection feature optimizes resource utilization and improves system efficiency.

The email alert system provides prompt notifications to homeowners in case of potential security breaches, equipping them with essential information and evidence to respond swiftly. The Raspberry Pi platform serves as the central control unit, offering a compact, affordable, and versatile solution for home security needs.

By leveraging IoT principles, the system enables real-time monitoring and seamless integration with other IoT devices, adding to its versatility and functionality.

The project's comparative cost vs. benefit analysis highlights the advantages of this solution over traditional and commercial alternatives, making it accessible to a wider range of homeowners.

In conclusion, this final year B.Tech project offers an innovative and practical approach to enhance home security using IoT and facial recognition technology. It provides homeowners with a reliable, cost-effective, and smart solution that significantly improves the safety and peace of mind within residential premises.

8.2 Future Scope

The IoT-based facial recognition home security system with email alert using Raspberry Pi offers a solid foundation for future enhancements and expansions. The project opens up several potential avenues for further development and improvement:

Integration with Smart Home Devices: The system can be extended to integrate with other smart home devices such as smart locks, lights, and thermostats. This integration would enable homeowners to control and manage their home security system through voice commands or mobile applications, adding convenience and automation to their daily lives.

Cloud-based Facial Recognition: Implementing cloud-based facial recognition can enhance the system's scalability and performance. By offloading the computational workload to cloud servers, the system can handle a larger database of authorized individuals, further improving accuracy and recognition speed.

Multi-factor Authentication: The project can be expanded to include additional layers of authentication, such as voice recognition or fingerprint scanning, to enhance security. Implementing multi-factor authentication ensures a higher level of confidence in identifying authorized individuals and reduces the risk of false positives.

Advanced Alerting Mechanisms: Besides email alerts, future developments can explore additional alerting mechanisms such as SMS notifications, push notifications to mobile devices, or integration with home security service providers. These enhancements would enable homeowners to receive alerts through multiple channels and take immediate action in response to security breaches.

Artificial Intelligence Integration: Leveraging artificial intelligence (AI) techniques can further improve the system's performance. AI algorithms can enable the system to adapt and learn from new facial data, continuously enhancing accuracy and recognition capabilities over time.

Privacy and Data Security: Future developments should also focus on addressing privacy concerns and ensuring data security. Implementing robust encryption methods and providing user control over data storage and access can help build trust and ensure the privacy of users' facial data.

The future scope of this project is extensive, with possibilities for integration, scalability, advanced authentication methods, and improved alerting mechanisms. By exploring these areas, the IoT-based facial recognition home security system can evolve into a comprehensive and intelligent solution that provides homeowners with enhanced security, convenience, and peace of mind.

Chapter 9: References

[1] https://www.researchgate.net/publication/339606773_IoT_based_facial_recognition _door_access_control_home_security_system_using_raspberry_pi

[2]https://ieeexplore.ieee.org/document/8982466

[3] https://circuitdigest.com/microcontroller-projects/raspberry-pi-iot-intruder-alert-system

[4] Python with deep learning,
https://www.tutorialspoint.c

[5] Raspberrypi official website
https://www.raspberrypi.com/

[6] Thonny official website
https://thonny.org/

Appendix A :
Appendix B :