

IoT Based Security System for Smart City Applications

PROJECT REPORT

Submitted in the fulfilment of the requirements for

the award of the degree of

Bachelor of Technology

in

Electronics and Communication Engineering

By

Jaba. Sri Venkata Siva Naga Tarun

[201FA05017]

Swetha. Jampani

[211LA05020]

Adapa. Ravi Teja

[211LA05035]

Under the Esteemed Guidance of

Mr. K. Lova Raju

Assistant professor

Department of ECE



VIGNAN'S
Foundation for Science, Technology & Research
(Deemed to be University)
-Estd. u/s 3 of UGC Act 1956

(ACCREDITED BY **NAAC** WITH “**A⁺**” GRADE)

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

(ACCREDITED BY **NBA**)

**VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY AND
RESEARCH**

(Deemed to be University)

Vadlamudi, Guntur, Andhra Pradesh, India -522213

May 2024

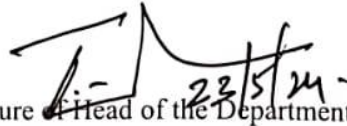
CERTIFICATE

This is to certify that the project report entitled "**IoT Based Security System for Smart City Applications**" that is being submitted by Jaba. Sri Venkata Siva Naga Tarun [201FA05017], Swetha. Jampani [211LA05020] and Adapa. Ravi Teja [211LA05035], in fulfilment for the award of B. Tech degree in Electronics and Communication Engineering, Vignan's Foundation for Science Technology and Research University, is a record of bonafide work carried out by them under the guidance of Mr. K. Lova Raju of ECE Department.



Signature of the Faculty Guide

K. Lova Raju, M.Tech., (Ph.D.), MIEEE.
Assistant Professor



Signature of Head of the Department

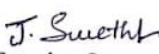
Dr. T. Pichaiah, M.E, Ph.D., MIEEE, FIETE
Professor & HoD

DECLARATION

We hereby declare that the project work entitled "**IoT Based Security System for Smart City Applications**" is being submitted to Vignan's Foundation for Science, Technology and Research (Deemed to be University) in fulfilment for the award of B. Tech degree in Electronics and Communication Engineering. The work was originally designed and executed by us under the guidance of Mr. K. Lova Raju at Department of Electronics and Communication Engineering, Vignan's Foundation for Science Technology and Research (Deemed to be University) and was not a duplication of work done by someone else. We hold the responsibility of the originality of the work incorporated into this Project Report.

Signature of the candidates


Jaba. Sri Venkata Siva Naga Tarun (201FA05017)


Swetha. Jampani (211LA05020)


Adapa. Ravi Teja (211LA05035)

ACKNOWLEDGEMENT

The satisfaction that comes from successfully completing any task would be incomplete without acknowledging the people who made it possible, whose ongoing guidance and encouragement have been essential to the achievement.

We are greatly indebted to **Mr. K. Lova Raju**, my revered guide and Associate Professor in the Department of Electronics and Communication Engineering, VFSTR (Deemed to be University), Vadlamudi, Guntur, for his valuable guidance in the preparation of this project report. He has been a source of great inspiration and encouragement to us. He has been kind enough to devote considerable amount of his valuable time in guiding us at every stage. This is our debut, but we are sure that we are able to do many more such studies, under the lasting inspiration and guidance given by respectable guide.

We would also like to thank to **Dr. T. Pitchaiah**, Head of the Department, ECE for his valuable suggestion.

We would like to specially thank, **Dr. N. Usha Rani**, Dean, School of Electrical, Electronics and Communication Engineering for her help and support during the project work.

We thank our project coordinators **Dr. Satyajeet Sahoo, Dr. Arka Bhattacharyya, Mr. Abhishek Kumar** and **Mr. M. Vamsi Krishna** for continuous support and suggestions in scheduling project reviews and verification of the report. Also, thank to supporting staff of ECE Department for their technical support for timely completion of project.

We would like to express our gratitude to **Dr. P. Nagabhusan**, Vice-Chancellor, VFSTR (Deemed to be University) for providing us the greatest opportunity to have a great exposure and to carry out the project.

Finally, we would like to thank our parents and friends for the moral support throughout the project work.

Name of the Student

Jaba. Sri Venkata Siva Naga Tarun (201FA05017)

Swetha. Jampani (211LA05020)

Adapa. Ravi teja (211LA05035)

Abstract

This research introduces an Internet of Things (IoT) based security system designed for heterogeneous applications. IoT based security system on Raspberry Pi 3 was developed to improve the effectiveness of intruder detection. The developed prototype was tested under a few conditions to determine the accuracy of intruder detection and compare the results with a system that uses a PIR motion sensor for intruder detection. From the results obtained, the developed IoT based security system using PIR motion sensor-based security system. The system employs a diverse range of technologies to ensure robust security measures. Through the integration of IoT devices, it addresses the unique challenges posed by varying application environments. The proposed system leverages a network of interconnected sensors, actuators, and intelligent algorithms to detect and respond to security threats effectively. Its adaptability across different application domains makes it a versatile solution. This abstract highlight the innovative approach and potential impact of the IoT-based security system in safeguarding diverse and heterogeneous contexts. Finally, PIR Sensor and Pi camera plays major role in the project which detects the person and sends the picture to the owners Telegram as a Notification.

,

Major Design (Final Year Project Work) Experience Information

Student Group	Jaba. Sri Venkata Siva Naga Tarun (201FA05017)	Swetha. Jampani (211LA05020)	Adapa. Ravi Teja (211LA05035)
Project Title	IoT based security system for Smart City applications.		
Program Concentration Area	Internet of Things, Embedded System Design, and Sensors.		
Program Concentration Area	Smart City applications.		
Constraints – Examples			
Economic	Fixed budget (limited), Cost-effective systems.		
Environmental	Designed for sustainable smart city applications through real-time monitoring, Notifications for intrusion detection.		
Sustainability	Promotes sustainable through efficient resource use, and security enabling		
Manufacturability	Components commercially available and system designed to be manufacturable		
Ethical	Followed ethical principles in design and testing		
Health and Safety	Safe system design, no hazardous materials		
Social	Improved waste management practices , environmental and benefits		
Political	None		
Other	Real time monitoring, Alerts enabling for security issues, Security purpose		
Standards			
1. ISO/IEC 30141	Internet of Things Reference Architecture standard. It is used to implement an interoperable and secure IoT-based Security system for smart city Application.		
2. IEEE 802.11b/g/n	Wi-Fi communication protocol standards followed for wireless connectivity between the NodeMCU and the internet/cloud Server		
Previous Course Required for the Major Design Experience	1. Electronic Devices and Circuits 2. Embedded Systems 3. Internet of Things		


Supervisor


Project coordinator


Head of the department ECE

TABLE OF CONTENTS

Chapter 1	page no
INTRUDER DETECTION SYSTEM	
1.1 Introduction.....	1
1.2 Motivation.....	2
1.3 Objectives.....	3
1.4 Tools and Standards	4
 Chapter 2	
LITERATURE SURVEY	
2.1 Literature Review.....	5
 Chapter 3	
PROPOSED WORK AND COMPONENTS DESCRIPTION	
3.1 Block diagram of proposed system	7
3.2 Components Description.....	9
3.2.1 Raspberry Pi 3b+.....	13
3.2.2.PIR Sensor.....	14
3.2.2.1 Working Principal of PIR Sensor.....	16
3.2.3 Pi Camera	19
3.2.4 Solenoid lock.....	20
3.2.5 Buzzer.....	20
3.2Telegram bot.....	22
3.3.1 Steps to create telegram bot.....	23
3.4 VNC Viewer.....	27
 Chapter 4	
SOFTWARE TOOLS USED	
4.1 RASPBIAN OS	30
4.2 Commands used in project.....	31
4.3 Steps to install Raspbian OS	33

Chapter 5

RESULTS

5.1 Output.....	35
5.2 conclusion.....	37

Chapter 6

ADVANTAGE,APPLICATION AND FUTURE SCOPE

6.1 Advantages	38
6.2 Applications.....	41
6.3 Future Scope.....	42

Chapter 7

REFERENCE

References.....	43
-----------------	----

Chapter 8

APPENDIX

Appendix.....	46
---------------	----

LIST OF FIGURES

TITLE	PAGE NO
3.1 Block diagram of proposed work.....	7
3.2.1 Raspberry pi 3B+.....	11
3.2.2 PIR Sensor.....	13
3.2.2.1 Working of PIR Sensor.....	15
3.2.2 PIR Sensor.....	16
3.2.3 Pi Camera.....	17
3.2.4 Solenoid lock.....	20
3.2.5 Buzzer.....	21
3.3 Telegram bot.....	26
5.1.1:- Motion Detected and door locked.....	35
5.1.2:- Motion Detected and door unlocked.....	35
5.1.3:- Motion Detected and Image Alert to Telegram Bot.....	36

LIST OF ACRONYMS AND ABBREVIATIONS

API'S :- Application Programming Interfaces

CSI:- Camera Serial Interface

CMOS:- Complementary Metal Oxide Semi-conductor

DC:- Direct Current

DIY:- Do it yourself

DSI:- Display Serial Interface

GPIO:- General Purpose Input output

HD:- High-definition

HDMI:- High definition multimedia interface

IDS:- Intruder Detection system

IEC:- International Electrotechnical Commission

IEEE:- Institute of Electrical and Electronics Engineers

IoT:- Internet of Things

ISO:- International Organization for Standardization

IR:- Infrared

LAN:- Local Area Network

MIPI:- Mobile Industry Processor Interface

MP:- Megapixel

OS:- Operating System

PIR:- Passive Infrared

PIXEL:- Pi Improved Xwindows Environment, Lightweight

PoE:- Power over Ethernet

PoE HAT:- Power over Ethernet Hardware Attached on top

RAM:- Random Access Memory

SDRAM:- Synchronous Dynamic Random Access Memory

SSD:- Solid State Drive

SSH:- Secure Shell

TCO:- Total Cost of Ownership

USB:- Universal Serial Bus

VNC:- Virtual Network Computing

VSB:- Virtual smart Barrie

CHAPTER-1

Intruder Detection System

1.1 INTRODUCTION

The increasing prevalence of interconnected devices in diverse applications has brought forth a pressing need for robust security solutions. In response to this demand, we present an innovative IoT-based Security System tailored for Heterogeneous Applications. This system leverages the power of the Internet of Things (IoT) to fortify security measures across a wide spectrum of application domains. With the proliferation of IoT devices, each presenting unique challenges, our solution provides a comprehensive and adaptable framework. In this dynamic landscape, where smart technologies permeate sectors ranging from healthcare to industrial automation, a one-size-fits-all security approach falls short. Our IoT-based Security System acknowledges the heterogeneity inherent in various applications, offering a tailored and effective defense against emerging threats. In this work, a vision-based home security system on Raspberry Pi 3 was developed to improve the effectiveness of motion detection. The developed prototype was tested under a few conditions to determine the accuracy of motion detection and compare the results with a system that uses a PIR motion sensor for motion detection. From the results obtained, the developed vision-based home security system using PIR motion sensor-based security system.

1.2 MOTIVATION

The motivation for implementing an intruder detection system is rooted in the fundamental need for security and protection against unauthorized access and potential threats. Whether in residential, commercial, or industrial settings, the primary goal is to safeguard assets, property, and individuals from theft, vandalism, and other malicious activities. Intruder detection systems serve as proactive guardians, continuously monitoring designated areas and alerting stakeholders to any suspicious behavior or security breaches. Beyond security, these systems contribute to safety by facilitating timely responses to emergencies, enabling evacuation procedures, and reducing risks associated with unauthorized intrusions. Additionally, compliance with regulatory requirements, insurance incentives, and the peace of mind afforded to homeowners and property owners further motivate the adoption of intruder detection systems. By acting as a deterrent to criminal activity and integrating with other security measures, such as access control and surveillance systems, these solutions enhance overall security capabilities and contribute to a safer and more secure environment for individuals, businesses, and organizations.

1.3 OBJECTIVES

The objectives for an intruder detection system can vary depending on the specific needs and requirements of the environment it is deployed in. However, some common objectives include:

- **Detection of Unauthorized Access:** The primary objective of an intruder detection system is to identify and alert users or security personnel to unauthorized access or breaches of security within a protected area. This includes detecting intruders attempting to enter a premises unlawfully or accessing restricted areas.
- **Timely Response to Security Threats:** Intruder detection systems aim to provide timely alerts and responses to security threats. This involves triggering alarms or notifications as soon as suspicious activity is detected, allowing for immediate action to be taken to address the threat.
- **Prevention of Theft and Vandalism:** One of the key objectives of intruder detection systems is to deter theft, vandalism, and other criminal activities by creating a visible deterrent and alerting potential intruders to the presence of security measures.
- **Protection of Assets and Property:** Intruder detection systems help protect valuable assets, property, and equipment from damage, theft, or unauthorized use. By detecting and deterring intruders, these systems help safeguard physical assets and minimize financial losses.
- **Enhancement of Safety and Security:** Intruder detection systems contribute to the overall safety and security of individuals within a premises by providing early warning of potential security threats. This includes protecting occupants from harm and ensuring their well-being in emergency situations.
- **Facilitation of Emergency Response:** In the event of an intrusion or security breach, intruder detection systems facilitate emergency response efforts by providing accurate information about the location and nature of the threat. This enables security personnel or authorities to respond swiftly and effectively to mitigate the threat.

- Compliance with Regulations and Standards: Intruder detection systems may be deployed to meet regulatory requirements or industry standards related to security and safety.
- Compliance with these regulations helps ensure that the premises are adequately protected and reduces the risk of legal liabilities.
- Integration with Other Security Systems: Intruder detection systems often integrate with other security systems, such as surveillance cameras, access control systems, and alarm monitoring services. The objective is to create a comprehensive security ecosystem that enhances overall protection and situational awareness.

By aligning with these objectives, intruder detection systems can effectively enhance security, deter criminal activity, and protect assets and individuals in a wide range of environments.

1.4 TOOLS AND STANDARDS

Tools and standards are:

- ISO/IEC 30141 Internet of Things Reference Architecture standard.
- It is used to implement an interoperable and secure IoT based security system for smart city application.
- IEEE 802.11b/g/n Wi-Fi communication protocol standards followed for wireless connectivity between the NodeMCU and the internet/cloud Server

CHAPTER-2

LITERATURE SURVEY

2.1 Literature Review

It is well known that various home security systems have been proposed to address home and office security problems. The recent development in the 4th industrial revolution has opened new opportunities for cutting edge technologies to be incorporated into design of intrusion detection. In some cases, mobile devices and applications have played some remarkable roles in smart intrusion systems, in terms of tasks and operations. In recent years, intelligent home advanced technologies and the IoT have provide relief to home- owners to have remote access and control to their home through reliable and secure systems. In addition to this, the daily call from home- owners and the national crime rate has pushed several innovations and research in this direction, to alleviate the problems.

As an integral part of smart homes, several intelligent based alarm systems and security systems are already on the shelf with advanced sensors and wireless based networks. These systems are designed to detect intruders and inform respective home- owners, who may then act on the provided information or messages. With little or no intelligent features, these systems have limitations to address the ever- dynamic crimes rate in the 21st Century. What is often observed, is that most of these traditional ways of detecting crime are limited to detect dynamic crime environment and, often, most of them provide false alarms.

In the past, home security was mainly considered to be an alarm that simply triggered when intruders broke into the domain where it was set, whereas, in an intelligent based secure home, or office, it may be armed with smart and intelligent controlled security systems. Attempting to advance home security, various studies have been carried out by many authors on the intruder Detection Systems (IDS).

Based on the contribution of Jaafar, their work focusses on the design of Dynamic Home Automation alert systems with laser interfaces on webpages and which runs on the Window's 10 mobile application with an intelligent controller of Raspberry Pi 2. The proposed system is visually monitored on the web, and only responds to parameters set on the web. The work comprises three indicators, such as laser, light and alarm, and has the capability of sending messages as well via a mobile. devices

The contribution by Anitha presents home security as a useful integration of the IoT, while considering using an inexpensive security system. The system is designed in a such a way as to inform the home or factory owner by sending a notification and demanding that the necessary action be taken. The system intelligence is a microcontroller base, integrated with a magnetic Reed sensor to acquire data, a buzzer sounding alarm, Wi-Fi module and, lastly, ESP8266 to synchronize to the internet.

The contribution by Abhilash centered on design of a home security system that is economically viable, with an energy efficient scheme, and miniaturized as well. In this case, Pi 3 model B is also used as a controller and integrator, and a Passive Infrared Sensor (PIR sensor) and a webcam are connected. The engraved Pi camera is used to capture images, and the OpenCV python library is used to detect and analyze the captured images. The PIR sensor functions as a motion detector, and the computation is done with python libraries. The captured image, as a result of the triggered sensor, would be sent to the configured email account using the IoT, SMTP and MINE technologies.

CHAPTER-3

Proposed work and Components Description

3.1 Block Diagram of proposed work

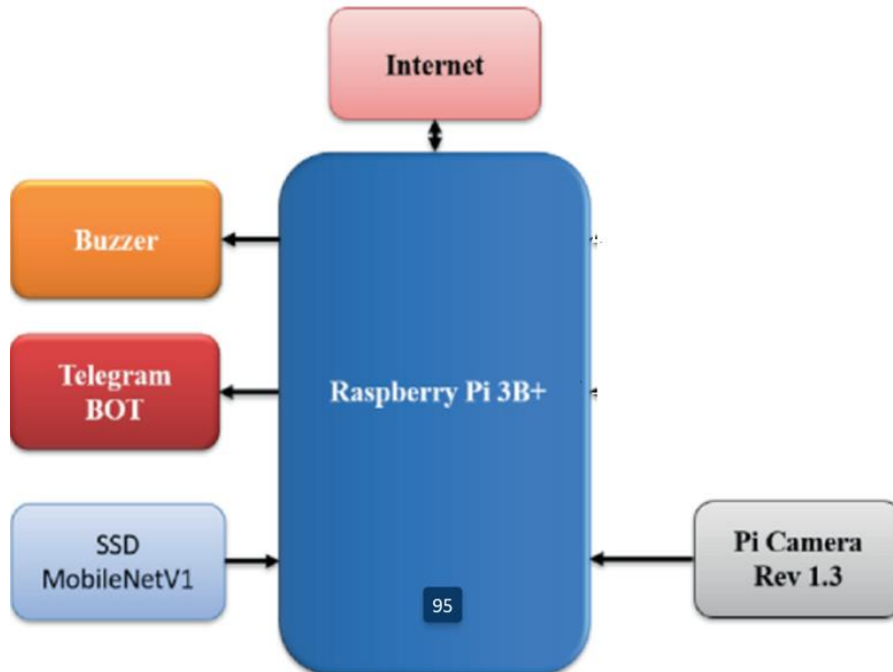


Fig-3.1: Block diagram of proposed work

This design will outline the architecture, components, and functionality of the intruder detection system, selecting appropriate technologies such as motion sensors, door/window sensors, and surveillance cameras. Emphasis will be placed on seamless integration with existing security infrastructure, ensuring compatibility and interoperability with access control systems and alarm monitoring services. Implementation will involve meticulous installation and configuration of components, strategic placement of sensors, and rigorous testing to validate system effectiveness.

WiFi:-

IEEE 802.11b/g/n refers to a set of wireless networking standards developed by the Institute of Electrical and Electronics Engineers (IEEE).

1. IEEE 802.11b: This was one of the earliest Wi-Fi standards, introduced in 1999. It operates in the 2.4 GHz frequency band and supports data transfer rates up to 11 Mbps. Despite its relatively slow

speed compared to modern standards, 802.11b had widespread adoption due to its compatibility with existing hardware and its ability to provide wireless connectivity over longer distances compared to previous technologies.

ISO/IEC 30141 is a standard that pertains to "Internet of Things (IoT) – Reference Architecture." This standard, published by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC), provides a comprehensive framework for understanding and implementing IoT systems.

1. **Architecture:** The standard outlines the architectural principles and components of IoT systems, including sensors, actuators, communication protocols, data processing, and application layers. It defines how these components interact to enable seamless connectivity, interoperability, and scalability within IoT ecosystems.

2. **Interoperability:** ISO/IEC 30141 emphasizes the importance of interoperability among diverse IoT devices, platforms, and applications. It provides guidelines for ensuring compatibility and seamless communication between different IoT systems, regardless of manufacturer or technology used.

3. **Security and Privacy:** Security and privacy considerations are fundamental to IoT deployments. The standard addresses these concerns by outlining best practices for implementing robust security measures, such as encryption, authentication, access control, and data integrity, to protect IoT devices and data from unauthorized access, manipulation, or disclosure.

4. **Lifecycle Management:** ISO/IEC 30141 covers the entire lifecycle of IoT systems, from design and development to deployment, operation, and maintenance. It provides guidelines for managing IoT devices, data, and infrastructure throughout their lifecycle, including provisioning, configuration, monitoring, and decommissioning.

5. **Scalability and Flexibility:** IoT environments are characterized by their dynamic and heterogeneous nature. The standard promotes scalability and flexibility in IoT architectures, allowing them to

accommodate a wide range of devices, applications, and use cases while adapting to evolving requirements and technologies.

ISO/IEC 30141 serves as a valuable resource for organizations involved in designing, implementing, and managing IoT solutions. By adhering to its guidelines and principles, stakeholders can ensure the reliability, security, and interoperability of their IoT deployments, enabling them to realize the full potential of connected technologies in various domains, including smart cities, industrial automation, healthcare, transportation, and more.

3.2 COMPONENTS DESCRIPTION

3.2.1 Raspberry Pi 3B+

The Raspberry Pi 3B+ is a single-board computer developed by the Raspberry Pi Foundation. Released in March 2018, it's an updated version of the Raspberry Pi 3 Model B. Some of its key features include:

- **Processor:** It features a Broadcom BCM2837B0 chipset, which includes a quad-core ARM Cortex-A53 processor running at 1.4GHz.
- **Memory:** The Raspberry Pi 3B+ typically comes with 1GB of RAM.
- **Connectivity:** It offers built-in dual-band Wi-Fi (2.4GHz and 5GHz) and Bluetooth 4.2, making it easier to connect to networks and peripherals without additional hardware.

- Ethernet: It has a Gigabit Ethernet port, which is an improvement over the previous model's 100Mbps Ethernet.
- USB Ports: There are four USB 2.0 ports for connecting peripherals such as keyboards, mice, and external storage.
- GPIO Pins: Like other Raspberry Pi models, it includes GPIO (General Purpose Input/Output) pins, allowing users to interface with external hardware and sensors.
- MicroSD Card Slot: It uses a microSD card for storage, where the operating system and user data are typically stored.
- Video Output: It supports HDMI output for connecting to displays.
- Power: The Raspberry Pi 3B+ can be powered via a micro-USB port.

Overall, the Raspberry Pi 3B+ is a versatile and affordable platform that is widely used for various projects, including DIY electronics, home automation, media centers, and educational purposes. Its popularity stems from its low cost, small form factor, and the extensive community support and resources available online.



Fig-3.2.1 Raspberry pi 3B+

The Raspberry Pi 3B+ offers several advantages over its predecessors and some other models in the Raspberry Pi lineup:

- **Improved Performance:** With a faster quad-core processor running at 1.4GHz compared to the 1.2GHz processor in the Raspberry Pi 3 Model B, the 3B+ offers better performance for multitasking and demanding applications.
- **Enhanced Connectivity:** The inclusion of dual-band Wi-Fi (2.4GHz and 5GHz) and Bluetooth 4.2 allows for faster wireless connections and better compatibility with a wider range of peripherals. This makes it more versatile for IoT (Internet of Things) and connectivity-focused projects.
- **Gigabit Ethernet:** The Raspberry Pi 3B+ features a Gigabit Ethernet port, offering significantly faster wired network speeds compared to the 100Mbps Ethernet port on earlier models. This makes it more suitable for network-intensive tasks and applications.

- Power-over-Ethernet (PoE) Support: Some variants of the Raspberry Pi 3B+ come with PoE support, allowing it to be powered through an Ethernet cable, which simplifies power management in certain setups, especially in industrial or IoT deployments.
- Thermal Management: The Raspberry Pi 3B+ incorporates improved thermal management, including a metal heat sink and power management firmware, which allows it to maintain higher performance levels under sustained workloads compared to earlier models.
- Backward Compatibility: Despite the improvements and additions, the Raspberry Pi 3B+ maintains backward compatibility with previous Raspberry Pi models, ensuring that existing projects and accessories remain compatible.
- Availability: Being one of the later models in the Raspberry Pi 3 series, the 3B+ benefits from the maturity of the platform, widespread community support, and a large ecosystem of compatible accessories and software.

Overall, while newer Raspberry Pi models offer additional features and performance enhancements, the Raspberry Pi 3B+ strikes a balance between performance, connectivity, and affordability, making it a popular choice for a wide range of projects and applications.

- Raspberry Pi 3 Model B+ has a 64-bit quad-core processor running at 1.4GHz.
- Dual-band 2.4GHz and 5GHz wireless LAN.
- Power:- 5v/2.5A DC power input.
- RAM:- 1GB SDRAM
- MIPI DSI
- CSI camera port
- PoE capability via a separate PoE HAT

3.2.2 PIR Sensor

A PIR (Passive Infrared) sensor is a type of motion sensor widely used in various electronic devices and systems. It operates based on the principle of detecting changes in infrared radiation emitted by objects within its field of view. The sensor consists of a pyroelectric sensor divided into two halves, each covered by a filter to detect specific infrared wavelengths. When an object moves within the sensor's detection range, it causes a temperature difference between the two halves of the sensor, triggering the sensor to activate. PIR sensors are commonly used in security systems, lighting control, and occupancy detection applications. They offer advantages such as fast response times, low power consumption, and simple operation. However, they have limitations, including their inability to detect motion through obstacles and susceptibility to environmental factors like temperature changes. Despite these limitations, PIR sensors remain popular due to their reliability, affordability, and effectiveness in motion detection.

- A Passive infrared sensor is a type of motion sensor that detects infrared radiation emitted by objects within its field of view.
- It is commonly used in security systems, lighting controls and smart home devices to detect movement and trigger a response, such as turning on lights or initiating an alarm
- The PIR Sensor consists of multiple infrared sensitive detector
- Most PIR Sensors are 3-pin designs that produce a digital output.



Fig-3.2.2 PIR Sensor

3.2.2.1 WORKING PRINCIPAL OF PIR SENSOR

The working principle of a PIR (Passive Infrared) sensor revolves around its ability to detect changes in infrared radiation within its detection range. These sensors typically consist of a pyroelectric sensor, which is divided into two segments. Each segment is equipped with a lens that focuses infrared radiation onto the pyroelectric material. When an object moves in front of the sensor, it emits infrared radiation proportional to its temperature. The two segments of the sensor detect this radiation, and any change in the radiation pattern triggers the sensor. This change occurs because the moving object causes alterations in the temperature gradient across the sensor's segments. These alterations are then converted into an electrical signal, which the sensor interprets as motion. PIR sensors are passive devices, meaning they do not emit any energy themselves but instead respond to changes in their environment. This makes them ideal for motion detection applications, such as security systems, automatic lighting, and occupancy sensing.

PIR sensors, or Passive Infrared sensors, operate by detecting infrared radiation emitted by objects within their detection range. This detection occurs without emitting any energy themselves, hence the term "passive."

Inside a PIR sensor, there's a pyroelectric sensor composed of materials that generate a voltage when exposed to infrared radiation. This sensor is split into two halves, each with a cover made of a special material that filters infrared radiation.

When an object moves within the sensor's field of view, it causes a change in the infrared radiation pattern detected by the two halves of the sensor. This change generates a voltage difference between the two halves, which is then amplified and processed by the sensor's electronics.

The output of the sensor is a digital signal, indicating whether motion has been detected or not. PIR sensors are commonly used in various applications, including security systems, automatic lighting, and occupancy sensing, due to their reliability, low cost, and ease of use.

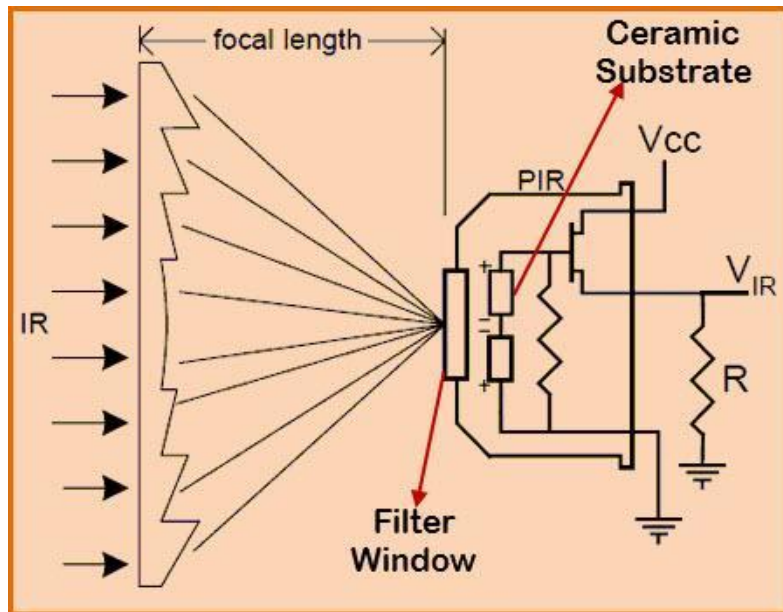


Fig 3.2.2.1 working of PIR Sensor

- It work on the principle that, wherever it detects a change in infrared radiation, it generates a digital output signal.
- Sensor consists of a Fresnel lens Pyroelectric material.
- It is immune to temperature, humidity and noise.
- pyroelectric, material detects the changes in formed radiation and generates an off Signal.



Fig-3.2.2 PIR Sensor

3.2.3 Pi Camera

The Pi Camera, also known as the Raspberry Pi Camera Module, is an accessory designed specifically for use with Raspberry Pi single-board computers. Here's what you need to know about it:

Types: There are primarily two types of Pi cameras:

- Raspberry Pi Camera Module V1: This is the original camera module released by the Raspberry Pi Foundation. It has a 5-megapixel sensor and is capable of capturing still images as well as video.
- Raspberry Pi Camera Module V2: This is an updated version with an 8-megapixel sensor. It offers better image quality and improved low-light performance compared to the V1 module.
- Connection: The Pi Camera Module connects to the Raspberry Pi's CSI (Camera Serial Interface) port, which is a ribbon cable connector specifically designed for interfacing with camera modules.

- **Features:** Both versions of the Pi Camera Module support features like auto-exposure, white balance, and video encoding. They are capable of capturing high-definition (HD) video and still images.
- **Accessories:** Various accessories are available for the Pi Camera, including camera cases, mounts, and lenses. Some third-party accessories offer additional features like adjustable focus or infrared (IR) capabilities for night vision.
- **Software Support:** The Pi Camera Module is supported by the official Raspberry Pi OS (formerly Raspbian) as well as other popular Linux distributions for the Raspberry Pi. There are also libraries and APIs available for programming and controlling the camera module using languages like Python.

Applications: The Pi Camera Module is widely used for various applications, including:

- Photography and videography projects
- Surveillance systems and security cameras
- Remote monitoring and wildlife observation
- Computer vision and machine learning projects
- Video streaming and video conferencing

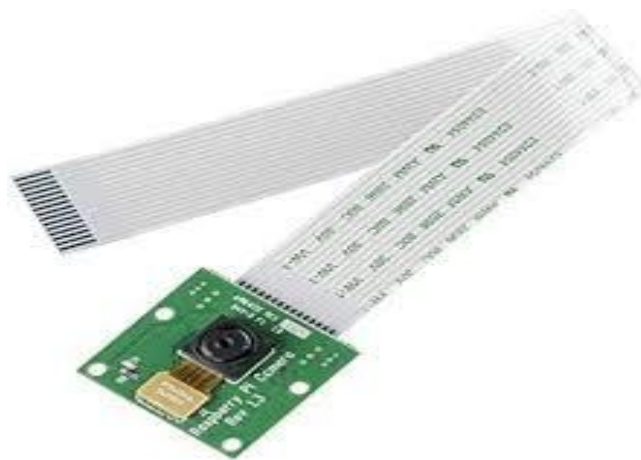


Fig-3.2.3 Pi Camera

The Pi Camera, also known as the Raspberry Pi Camera Module, operates as a peripheral device specifically designed to capture images and video when connected to a Raspberry Pi single-board computer. Its operation revolves around capturing light through a lens, converting it into digital data, and then processing it to produce visual output. When connected to the Raspberry Pi's CSI (Camera Serial Interface) port, the camera module becomes an integral part of the Pi's system, enabling users to capture still images or record video. The camera module consists of a lens, an image sensor, and supporting electronics. The lens focuses incoming light onto the image sensor, which is typically a CMOS sensor capable of capturing either still images or video frames.

Once the light is captured, the sensor converts it into digital data, which is then processed by the camera module's onboard electronics. These electronics handle tasks such as adjusting exposure, white balance, and other image settings. The processed image data is then transferred to the Raspberry Pi's memory via the CSI port, where it can be accessed and further processed by software running on the Pi. Users can control the camera module and capture images or video programmatically using various programming languages like Python, which provides libraries and APIs specifically designed for interfacing with the camera module.

Overall, the Pi Camera module provides an affordable and versatile solution for adding imaging capabilities to Raspberry Pi projects, enabling applications such as photography, videography, computer vision, and remote monitoring.

The Raspberry Pi Camera Module is available in different variations based on their megapixel (MP) count, offering users flexibility for various projects. The original version, known as the 5-megapixel Camera Module (V1), features an Omni Vision OV5647 sensor. It's capable of capturing still images with resolutions up to 2592 x 1944 pixels and supports video recording at 1080p30, 720p60, and 640x480p60/90. The updated version, the 8-megapixel Camera Module (V2), utilizes a Sony IMX219 sensor. This model provides enhanced image quality and resolution, suitable for more demanding applications. Both variations are compatible with Raspberry Pi boards and offer an accessible way to integrate imaging capabilities into projects, ranging from photography and videography to computer vision and surveillance.

3.2.4 Solenoid lock

A solenoid lock is a type of locking mechanism that operates using electromagnetic principles. It consists of a coil of wire, known as a solenoid, which becomes magnetized when an electric current passes through it. This magnetization creates a magnetic field that either attracts or repels a metal component within the lock, thereby securing or releasing the locking mechanism. One of the key components of a solenoid lock is the solenoid itself. The solenoid is typically made of a coil of insulated wire wound around a cylindrical core, such as iron or steel. When an electric current is applied to the coil, it generates a magnetic field around the core, which in turn exerts a force on any nearby ferromagnetic material. In a solenoid lock, this magnetic force is used to either engage or disengage the locking mechanism. When the solenoid is energized, the magnetic field it generates pulls or pushes on a metal component, such as a bolt or plunger, to either lock or unlock the door or device. Solenoid locks are commonly used in various applications, including electronic door locks, vending machines, safes, and cabinets. They offer several advantages over traditional mechanical locks, including faster operation, remote control capabilities, and the ability to integrate with electronic access control systems. One of the key benefits of solenoid locks is their ability to be controlled remotely. By connecting the solenoid lock to a control system, such as a keypad, card reader, or biometric scanner, access to the locked device or area can be restricted to authorized individuals only. This makes solenoid locks ideal for use in security-sensitive environments where access needs to be closely monitored and controlled. Another advantage of solenoid locks is their fast operation. Unlike mechanical locks, which require physical manipulation to engage or disengage, solenoid locks can be activated almost instantly with the flick of a switch or the press of a button. This makes them particularly well-suited for high-traffic areas where speed and convenience are important. Additionally, solenoid locks can be designed to fail-safe or fail-secure, depending on the specific requirements of the application. A fail-safe solenoid lock will automatically unlock in the event of a power failure or system malfunction, allowing for easy egress in emergency situations. Conversely, a fail-secure solenoid lock will remain locked in the event of a power failure, providing an extra layer of security. Solenoid locks are a versatile and reliable locking solution that offer fast operation, remote control capabilities, and customizable security options. Whether used in electronic door locks, vending machines, or safes, solenoid locks provide an effective means of controlling access and securing valuable assets.



Fig-3.2.4 solenoid lock

3.2.5 Buzzer

A buzzer is an electrical component that produces sound when an electrical current is passed through it. Here are some key points about buzzers:

➤ **Types of Buzzers:**

Piezoelectric Buzzers: These buzzers produce sound using the piezoelectric effect, where an electrical current causes a piezoelectric crystal to vibrate, generating sound waves.

Magnetic Buzzers: Magnetic buzzers use an electromagnet and a diaphragm to produce sound. When an electrical current passes through the coil of the electromagnet, it creates a magnetic field that attracts the diaphragm, causing it to move and produce sound.

➤ **Operation:**

When an electrical signal is applied to the buzzer, it energizes the internal components, causing them to vibrate and produce sound waves.

The frequency and amplitude of the sound produced depend on the design and characteristics of the buzzer.

➤ **Applications:**

Buzzers are commonly used as alerting devices in various electronic systems and appliances.

They are used in alarm systems, timers, doorbells, electronic games, and notification systems.

In industrial settings, buzzers are often used to signal the completion of a process or to alert workers to potential hazards.

➤ Types of Sounds:

Buzzers can produce different types of sounds depending on their design and purpose. These sounds can range from simple beeps to continuous tones or melodies. Some buzzers may have built-in oscillators or can be driven by external oscillators to produce specific frequencies or patterns of sound.

➤ Control:

In many applications, buzzers can be controlled using a microcontroller or other electronic circuitry to produce sounds based on predefined conditions or events.

They can be turned on or off, and the frequency or pattern of the sound can be adjusted programmatically.

Overall, buzzers are simple yet effective components for generating audible alerts and signals in electronic systems, making them essential in a wide range applications.



Fig-3.2.5 buzzer

In an intruder detection system, buzzers play a crucial role in alerting users to potential security breaches or unauthorized access. These systems typically integrate a buzzer as an audible alarm component, supplementing other detection mechanisms such as motion sensors or door/window contacts.

When an intrusion is detected, the buzzer is activated, emitting a loud, attention-grabbing sound to alert occupants or nearby individuals of the security breach. The sound serves as a deterrent to intruders and prompts swift responses from security personnel or authorities. Buzzers used in intruder detection systems are often selected for their reliability, loudness, and ease of integration. They may operate on low voltage for compatibility with the system's power source and typically have simple wiring configurations to facilitate installation.

In addition to their role in security systems, buzzers are also employed in various other applications, such as door entry systems, fire alarms, and industrial warning systems. Their versatility and effectiveness make them a fundamental component in ensuring the safety and security of both residential and commercial premises.

3.3 TELEGRAM BOT

Telegram bots are automated programs designed to interact with users within the Telegram messaging platform. These bots are created and managed by developers using the Telegram Bot API, which provides a framework for building various types of bots with diverse functionalities. Telegram bots have gained popularity due to their versatility and ease of use, offering a wide range of applications and services.

Telegram bots can perform a multitude of tasks, from providing information and entertainment to automating workflows and facilitating transactions. They can be integrated into group chats or used in private conversations, offering personalized experiences tailored to the needs and preferences of individual users.

Developers can create Telegram bots using various programming languages and frameworks, leveraging the extensive documentation and resources provided by Telegram. Bots can be programmed to respond to specific commands or keywords, process user inputs, and generate appropriate responses in the form of text, images, files, or interactive elements such as buttons and menus.

Telegram bots find applications across various domains, including customer support, news delivery, weather updates, language translation, productivity tools, gaming, and e-commerce. They enable businesses to engage with customers, streamline communication processes, and deliver personalized services efficiently.

Furthermore, Telegram provides developers with advanced features such as inline bots, which allow users to interact with bots directly in the chat interface without the need to initiate a separate conversation. Additionally, bots can be integrated with other third-party services and APIs, expanding their capabilities and enhancing their usefulness.

Overall, Telegram bots represent a powerful tool for automation, communication, and innovation, empowering developers to create innovative solutions and enriching the user experience within the Telegram ecosystem. With their growing popularity and continuous development, Telegram bots are poised to play an increasingly significant role in the future of messaging and digital interaction.

3.3.1 STEPS TO CREATE TELEGRAM BOT

Creating a Telegram bot involves several steps, but I'll outline them here for you:

➤ **Set Up a Telegram Account and Install Telegram:**

If you don't already have a Telegram account, you'll need to sign up for one.

Download and install the Telegram app on your mobile device or desktop.

➤ Find BotFather:

BotFather is the official bot provided by Telegram to help users create and manage bots. Search for "@BotFather" in the Telegram app and start a chat with it.

➤ Create a New Bot:

In the BotFather chat, type `/newbot`` and follow the instructions to create a new bot. You'll need to provide a name for your bot and a username (ending in "bot").

➤ Get Your Bot Token:

After creating the bot, BotFather will provide you with a token. This token is essential for authenticating your bot and interacting with the Telegram Bot API. Keep it safe and secure.

➤ Set Up Your Development Environment:

Choose the programming language and framework you want to use to develop your bot. Popular options include Python with libraries like `python-telegram-bot` or Node.js with `Telegraf`.

Install the necessary dependencies and set up your development environment.

➤ Write Bot Code:

Write the code for your bot using the chosen programming language and framework. This code will define how your bot interacts with users and processes their messages.

Use the Telegram Bot API documentation to understand the available methods and functionalities you can implement in your bot.

➤ Deploy Your Bot:

Once your bot code is ready, deploy it to a server or hosting platform. You can use platforms like Heroku, AWS, or Google Cloud Platform for hosting your bot.

Make sure to configure your bot to listen for incoming messages and respond accordingly.

➤ Test Your Bot:

Test your bot by sending messages to it in Telegram. Verify that it responds correctly to commands and messages as expected.

Debug any issues and make necessary adjustments to your bot code.

➤ Promote Your Bot (Optional):

If you want to make your bot accessible to a wider audience, promote it in Telegram bot directories or share it with friends and communities.

➤ Maintain and Update Your Bot:

Regularly maintain and update your bot to improve its functionality, fix bugs, and address user feedback.

Stay informed about changes and updates to the Telegram Bot API and adjust your bot code accordingly.

By following these steps, you can create your own Telegram bot and start interacting with users on the Telegram platform.

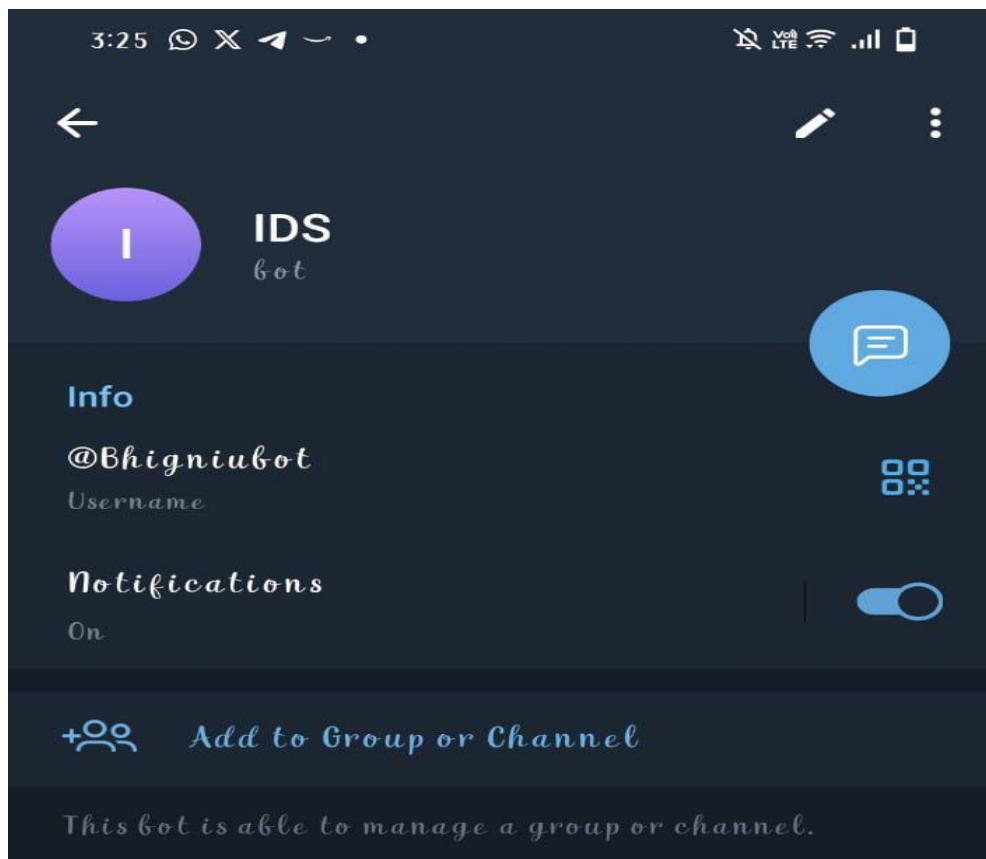


Fig-3.3 Telegram bot

3.4 VNC Viewer

VNC Viewer is a versatile remote desktop application designed for seamless access and control of computers or devices from a remote location. Operating under the VNC (Virtual Network Computing) framework, VNC Viewer, in conjunction with VNC Server, facilitates the remote interaction with graphical desktop environments across various platforms. Its cross-platform compatibility extends to Windows, macOS, Linux, Android, and iOS, ensuring accessibility from a wide range of devices. One of its key strengths lies in establishing secure connections between the viewer and server, employing encryption protocols to safeguard remote desktop sessions. In addition to security, VNC Viewer offers a suite of features including screen scaling, full-screen mode, file transfer capabilities, and clipboard sharing, enhancing the user experience and productivity during remote sessions. These features collectively make VNC Viewer an indispensable tool for remote desktop access, enabling efficient collaboration, troubleshooting, and management of computers or devices from virtually anywhere.

Sure, here are the steps to connect your Raspberry Pi to VNC Viewer:

- Install VNC Server on Raspberry Pi:

Open a terminal on your Raspberry Pi or connect to it via SSH.

Install VNC Server by running the following command:

```
---
```

```
sudo apt update
```

```
sudo apt install realvnc-vnc-server
```

```
---
```

- Enable VNC Server:

After installation, you need to enable VNC Server. You can do this via the Raspberry Pi Configuration tool or by entering the following command in the terminal:

```
---
```

```
sudo raspi-config
```

```
---
```

Navigate to `Interfacing Options` > `VNC` and select `Yes` to enable VNC Server.

➤ Note Down Raspberry Pi's IP Address:

You'll need the IP address of your Raspberry Pi to connect to it from VNC Viewer. You can find this by running the command `hostname -I` in the terminal.

➤ Download and Install VNC Viewer:

On your computer or mobile device, download and install VNC Viewer from the RealVNC website or app store.

➤ Launch VNC Viewer:

Open VNC Viewer on your computer or mobile device.

➤ Enter Raspberry Pi's IP Address:

In VNC Viewer, enter the IP address of your Raspberry Pi in the address bar and press Enter.

➤ Authenticate:

If prompted, enter the username and password of your Raspberry Pi to authenticate the connection.

➤ Connect:

Click on the "Connect" button or press Enter to establish the connection.

➤ Access Raspberry Pi Desktop:

Once connected, you should see the desktop of your Raspberry Pi in VNC Viewer, allowing you to interact with it as if you were directly connected to a monitor and keyboard.

That's it! You've successfully connected your Raspberry Pi to VNC Viewer, enabling remote access to its desktop environment.

CHAPTER 4

SOFTWARE TOOLS USED

4.1 RASPBIAN OS

Raspbian is the official operating system (OS) designed specifically for the Raspberry Pi single-board computers. Developed by the Raspberry Pi Foundation, Raspbian is based on the Debian Linux distribution and optimized for the unique hardware architecture of the Raspberry Pi.

Raspbian provides a user-friendly environment for beginners while also offering advanced features and customization options for experienced users and developers. It comes pre-installed with a suite of essential software, including a web browser, office productivity tools, programming environments, and multimedia applications, making it suitable for a wide range of tasks and projects.

One of the key features of Raspbian is its lightweight nature, optimized for the limited resources of the Raspberry Pi. This ensures smooth performance even on the Raspberry Pi's modest hardware specifications, making it suitable for various applications such as desktop computing, educational projects, IoT (Internet of Things) devices, and embedded systems.

Raspbian includes the PIXEL (Pi Improved Xwindows Environment, Lightweight) desktop environment, providing a modern and intuitive user interface. PIXEL offers features like a taskbar, menu system, file manager, and customizable desktop settings, making it easy for users to navigate and interact with their Raspberry Pi.

Furthermore, Raspbian benefits from the extensive support and community around the Raspberry Pi ecosystem. Users can access a wealth of resources, tutorials, forums, and software repositories to enhance their Raspbian experience and troubleshoot any issues they encounter.

Overall, Raspbian is a versatile and user-friendly operating system that empowers users to explore the capabilities of the Raspberry Pi and embark on a wide range of projects, from simple experiments

to complex applications. Its accessibility, performance, and rich feature set have contributed to its widespread adoption and popularity among Raspberry Pi enthusiasts and educators worldwide.

4.2 COMMANDS USED IN PROJECT

Intruder detection systems typically utilize a variety of commands and techniques to monitor, detect, and respond to unauthorized access or security breaches. Here are some common types of commands used in these systems:

➤ Arming/Disarming Commands:

- These commands are used to arm or disarm the intruder detection system. Arming the system activates its sensors and triggers alarms in case of any detected intrusion, while disarming the system temporarily disables these sensors, allowing authorized users to move freely without triggering false alarms.

➤ Zone Configuration Commands:

- Intruder detection systems often divide monitored areas into zones, each equipped with sensors or detectors. Zone configuration commands are used to define and configure these zones, specifying parameters such as sensor sensitivity, detection range, and alarm response actions for each zone.

➤ Sensor Testing Commands:

- Sensor testing commands allow users to perform diagnostic tests on individual sensors or detectors to ensure they are functioning correctly. These commands may include sensor calibration, sensitivity adjustment, and self-test routines to verify proper operation.

➤ Event Logging Commands:

- Event logging commands record and store information about detected events, alarms, and system activities in a centralized log or database. This log provides a chronological record of security-related events, enabling administrators to review past incidents, identify patterns, and troubleshoot issues.

- Alerting Commands:

- Alerting commands trigger immediate notifications or alerts in response to detected intrusions or security breaches. These alerts may take the form of audible alarms, visual indicators, notifications sent to security personnel or monitoring centers, or automated messages delivered via email, SMS, or other communication channels.

- System Status Commands:

- System status commands provide real-time information about the operational status of the intruder detection system, including the current arming state, sensor status, alarm conditions, and any active alerts or notifications. Users can use these commands to monitor the system's health and performance and take appropriate action if necessary.

- Remote Control Commands:

- Remote control commands enable authorized users to remotely manage and control the intruder detection system from a central monitoring station or via a web-based interface. These commands may include arming/disarming, zone configuration, event logging, and alerting functions accessible from a remote location.

Overall, these commands form the backbone of an effective intruder detection system, providing users with the tools and capabilities needed to monitor, protect, and secure their premises against unauthorized access and security threats.

4.3 STEPS TO INSTALL RASPBIAN OS

To install Raspbian OS on a Raspberry Pi, follow these steps:

➤ Download Raspbian Image:

- Visit the official Raspberry Pi website (<https://www.raspberrypi.org/downloads/>) and download the latest Raspbian image. Choose either the Raspbian with Desktop version or Raspbian Lite, depending on your preference and project requirements.

➤ Prepare SD Card:

- Insert an SD card (8GB or larger recommended) into your computer's SD card reader.

- Use a disk imaging tool such as Etcher (<https://www.balena.io/etcher/>) to write the Raspbian image to the SD card. Select the downloaded Raspbian image file and the target SD card, then click "Flash" to start the process.

➤ Configure Raspbian (Optional):

- If you're using the Lite version of Raspbian, you may want to configure certain settings before booting up your Raspberry Pi. This can include enabling SSH, setting up Wi-Fi credentials, or configuring network settings. To do this, create a file named "wpa_supplicant.conf" on the boot partition of the SD card and add your Wi-Fi credentials. Additionally, create an empty file named "ssh" (without any extension) to enable SSH.

➤ Insert SD Card into Raspberry Pi:

- Once the Raspbian image is written to the SD card, safely eject it from your computer and insert it into the SD card slot on your Raspberry Pi.

➤ Boot Up Raspberry Pi:

- Connect peripherals such as a keyboard, mouse, display, and power supply to your Raspberry Pi.
- Power on your Raspberry Pi, and it will boot up into the Raspbian desktop environment or command line interface, depending on the version of Raspbian you installed.

➤ Initial Setup:

- Follow the on-screen prompts to complete the initial setup of Raspbian, including language selection, keyboard layout, and password configuration.

➤ Update Raspbian:

- Open a terminal window and run the following commands to update Raspbian to the latest packages:

```
...
```

```
sudo apt update
```

```
sudo apt upgrade
```

```
...
```

➤ Configure Raspbian Settings:

- Explore the Raspberry Pi Configuration tool (accessible via the main menu or terminal) to configure various settings such as display resolution, localization options, interface preferences, and more.

➤ Start Using Raspbian:

- Once Raspbian is installed and configured to your preferences, you can start using your Raspberry Pi for various projects, experiments, and tasks.

By following these steps, you can easily install Raspbian OS on your Raspberry Pi and begin exploring the capabilities of this versatile platform.

CHAPTER 5

RESULTS

5.1 Output:-

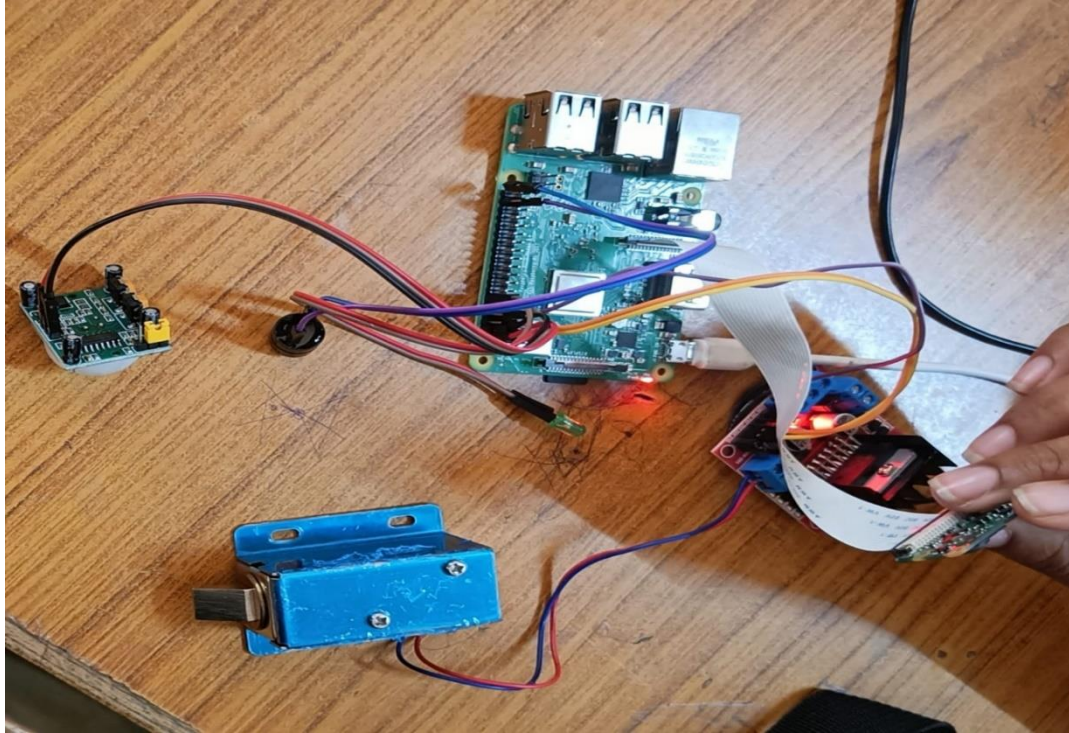


Fig 5.1.1:- Motion Detected and door locked

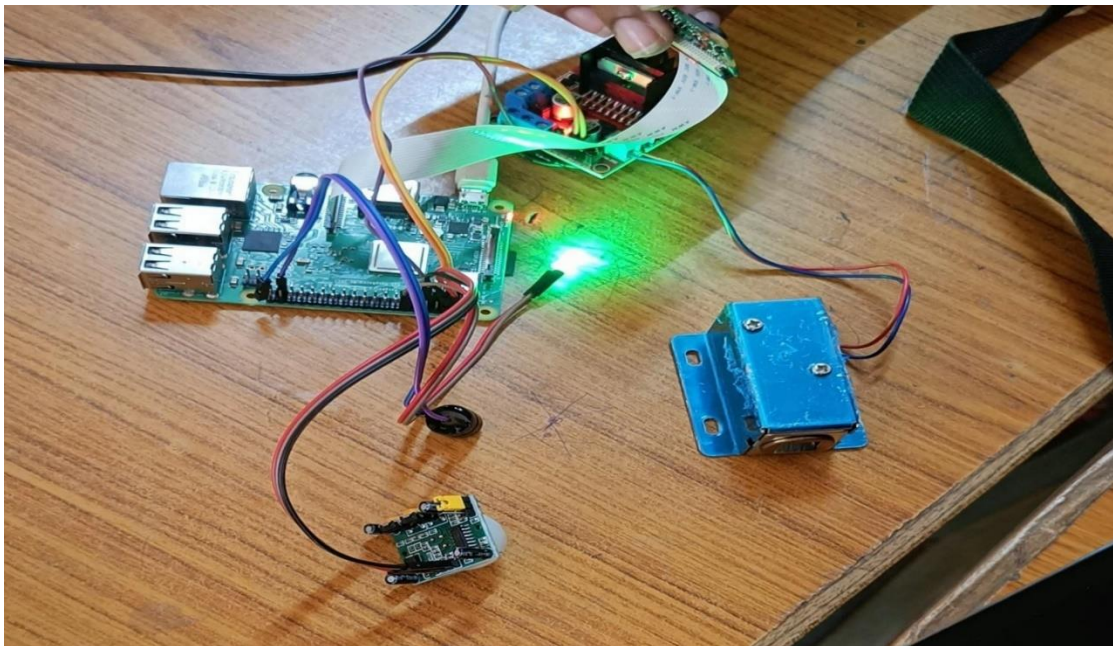


Fig 5.1.2:- Motion Detected and door unlocked



Fig 5.1.3:- Motion Detected and Image Alert to Telegram Bot

When a motion is detected then it enables the pi camera, when pi camera gets enabled it starts taking the detected picture and sends alert notification to Telegram Bot.

Not only sending alert notification when motion is detected buzzer starts beep sound and led glows.

1. Enhanced Public Safety: By deploying interconnected sensors, cameras, and other monitoring devices, cities can effectively detect and respond to security threats in real-time. This enables authorities to prevent crime, accidents, and other emergencies, ultimately leading to safer urban environments for residents and visitors alike.

2. Improved Emergency Response: IoT-enabled security systems facilitate faster and more coordinated emergency response efforts. Through automated alerts, geolocation data, and predictive analytics, first responders can quickly identify the location and nature of incidents, optimize resource allocation, and minimize response times, potentially saving lives in critical situations.

Overall, the results of implementing IoT-based security systems in smart city applications are transformative, leading to safer, more resilient, and sustainable urban environments. However, it's essential to address challenges related to privacy, cybersecurity, and digital inclusion to ensure equitable access and benefit for all citizens.

5.2 CONCLUSION

Concluding an IoT-based security system for smart city applications would involve summarizing its key benefits, potential challenges, and future prospects. Here's a concise conclusion:

In conclusion, the integration of IoT-based security systems in smart city applications represents a pivotal step towards creating safer, more efficient urban environments. By harnessing the power of interconnected devices, real-time data analytics, and automated response mechanisms, these systems empower city authorities to detect and respond to security threats with unprecedented speed and accuracy. From surveillance cameras and environmental sensors to smart streetlights and access control systems, the IoT infrastructure enables comprehensive monitoring and management of public spaces, critical infrastructure, and emergency situations. However, successful implementation requires addressing challenges such as privacy concerns, data security, interoperability, and infrastructure readiness. Moving forward, collaboration between government agencies, technology providers, and community stakeholders will be essential to realize the full potential of IoT-based security solutions in safeguarding citizens, optimizing resource allocation, and fostering sustainable urban development.

CHAPTER 6

ADVANTAGES, APPLICATIONS AND FUTURE SCOPE

6.1 ADVANTAGES

Intruder detection systems utilizing Raspberry Pi offer several advantages:

- **Cost-Effectiveness:** Raspberry Pi boards are affordable and provide a cost-effective solution for building intruder detection systems compared to traditional proprietary systems. This makes them accessible to a wider range of users, including hobbyists, small businesses, and educational institutions.
- **Flexibility and Customization:** Raspberry Pi is a versatile platform that allows for extensive customization and integration with various sensors, cameras, and communication modules. Users can tailor the intruder detection system to their specific needs and requirements, adapting it to different environments and scenarios.
- **Scalability:** Raspberry Pi-based intruder detection systems can be easily scaled up or down to accommodate different sizes of premises or varying levels of security requirements. Additional Raspberry Pi boards can be added to expand coverage or enhance system capabilities, providing scalability and flexibility as needs evolve.
- **Integration with IoT Devices:** Raspberry Pi's compatibility with IoT (Internet of Things) devices and protocols enables seamless integration with smart sensors, actuators, and other IoT devices. This integration allows for enhanced functionality, such as remote monitoring, automated responses, and integration with home automation systems.
- **Community Support and Resources:** The Raspberry Pi community is vast and active, providing a wealth of resources, tutorials, and open-source software for building intruder

detection systems. Users can leverage community forums, GitHub repositories, and online communities to troubleshoot issues, share knowledge, and collaborate on projects.

- **Educational Opportunities:** Raspberry Pi-based intruder detection systems offer valuable educational opportunities for students, hobbyists, and DIY enthusiasts to learn about electronics, programming, and cybersecurity. Building and experimenting with such systems can foster creativity, problem-solving skills, and hands-on learning experiences.
- **Low Power Consumption:** Raspberry Pi boards are energy-efficient and have low power consumption compared to traditional desktop computers or servers. This makes them suitable for deploying intruder detection systems in remote locations, off-grid setups, or environments with limited power availability.
- **Open-Source Software Ecosystem:** Raspberry Pi runs on open-source software, including the Raspbian operating system and a wide range of open-source software libraries and tools. This allows users to leverage existing software solutions, libraries, and frameworks to accelerate development and implementation of intruder detection systems.
- **Remote Monitoring and Control:** Raspberry Pi-based intruder detection systems can be accessed and controlled remotely via network connections. Users can monitor security status, receive alerts, and even arm/disarm the system remotely using web interfaces, mobile apps, or other network-connected devices.
- **Modularity and Expandability:** Raspberry Pi's modular architecture allows users to easily expand and enhance their intruder detection systems by adding additional components or modules. Whether it's integrating new sensors, upgrading hardware, or adding custom functionalities, Raspberry Pi's modular design facilitates system expansion and customization.

- **Low Maintenance Requirements:** Raspberry Pi-based intruder detection systems typically have low maintenance requirements due to their robustness and reliability. Once configured and deployed, these systems can operate autonomously for extended periods with minimal intervention, reducing the need for ongoing maintenance and support.
- **Integration with Cloud Services:** Raspberry Pi can seamlessly integrate with cloud services and platforms, enabling advanced features such as cloud storage, data analytics, and remote access. Users can leverage cloud services to store security footage, analyze sensor data, and implement advanced security features such as machine learning-based anomaly detection.
- **Powerful Processing Capabilities:** Despite its small size, Raspberry Pi boards offer powerful processing capabilities, allowing for real-time data processing, image analysis, and decision-making within the intruder detection system. This processing power enables advanced features such as facial recognition, object tracking, and intelligent alarm triggering.
- **Customizable User Interfaces:** Raspberry Pi-based intruder detection systems can have customizable user interfaces tailored to specific user preferences and requirements. Users can design intuitive dashboards, control panels, or mobile apps to visualize security data, configure system settings, and interact with the system in a user-friendly manner.
- **Low Total Cost of Ownership (TCO):** When considering the initial cost, maintenance requirements, and flexibility of Raspberry Pi-based intruder detection systems, they often have a lower total cost of ownership compared to proprietary solutions. This makes them an attractive option for budget-conscious users and organizations seeking cost-effective security solutions.

Overall, intruder detection systems using Raspberry Pi offer a cost-effective, flexible, and customizable solution for enhancing security and surveillance capabilities in various environments. With their scalability, integration options, and community support, Raspberry Pi-based systems are well-suited for both personal and professional use cases.

6.2 APPLICATIONS

IoT-based security systems offer numerous benefits for smart city applications:

Surveillance: Deploying IoT sensors and cameras enables real-time monitoring of public spaces, enhancing overall security and safety.

Traffic Management: IoT devices can track traffic flow, detect accidents, and optimize traffic signals, leading to smoother traffic patterns and improved safety.

Environmental Monitoring: Sensors can measure air quality, detect pollution levels, and monitor for natural disasters, helping cities respond proactively to environmental hazards.

Smart Lighting: IoT-connected lighting systems can adjust brightness based on factors like pedestrian traffic, saving energy and enhancing safety in dimly lit areas.

Emergency Response: IoT sensors can detect incidents like fires or gas leaks, triggering automatic alerts to emergency services for faster response times.

Asset Tracking: IoT technology can track the location and status of city assets such as vehicles and equipment, reducing theft and improving maintenance efficiency.

Water Management: IoT sensors can monitor water quality, detect leaks, and manage irrigation systems, promoting efficient water usage and reducing waste.

Public Health Monitoring: IoT devices can track public health metrics, such as detecting disease outbreaks or monitoring crowd density during events, aiding in disease prevention and control.

Waste Management: Smart bins equipped with IoT sensors can optimize waste collection routes, reduce overflow, and enable better recycling practices.

Access Control: IoT-based access control systems can manage entry to buildings and public spaces, enhancing security while improving the flow of people.

6.3 FUTURE SCOPE

The future scope for IoT-based security systems is incredibly promising, poised to revolutionize the way we safeguard our homes, businesses, and communities. As technology continues to advance, IoT devices offer unparalleled connectivity and data collection capabilities, enabling a comprehensive and proactive approach to security. These systems can seamlessly integrate various sensors, cameras, and actuators to monitor and control access points, detect intrusions, and respond to emergencies in real-time. Moreover, with the advent of artificial intelligence and machine learning, IoT security systems can analyse vast amounts of data to identify patterns, predict potential threats, and even autonomously adapt security protocols. From smart surveillance cameras and biometric authentication to IoT-enabled locks and perimeter sensors, the possibilities are vast. As the demand for enhanced security solutions grows amidst evolving threats, IoT-based security systems are poised to play a pivotal role.

CHAPTER 7

REFERENCES

- Nashwan Adnan OTHMAN, Ilhan AYDIN "A Face Recognition Method in the Internet of Things for Security Applications in Smart Homes and Cities" in Proc.2018 6th International Istambul Smart Grids and Cities Congress anFair(ICSG).
- T. Ojala, M. Pietikainen and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," Pattern Recognition vol. 29, pp. 51-59, January 1996.
- Raj G Anvekar, Dr.Rajeshwari M Banakar "Design Alternatives For End User Communication In IOT Based System Model" in Proc. 2017 IEEE International Conference on Technological Innovation in ICT For Agriculture and Rural Development(TIAR 2017).
- Monica Chillaron, Larisa Dunai, Guillermo Peris Fajarnes, Ismael Lengua Lengua "Face detection and recognition application for Android" in Proc. IECON2015-Yokohama November 9-12, 2015.
- Gagandeep Singh Nagpal, Gagandeep Singh, Jappreet Singh, Nishant Yadav "Facial Detection and Recognition using OpenCV on Raspberry Pi Zero" in Proc. International Conference on Advances in Computing. Communication Control and Networking (ICACCCN2018).
- Neha Patil, Shrikant Ambatkar and Sandeep Kakde "IoT Based Smart Surveillance Security System using Raspberry Pi" in Proc. International Conference on Communication and Signal Processing, April 6-8, 2017, India.

- Ms. Ashwini Pawar, Prof. V. M. Umale "Internet of Things Based Home Security Using Raspberry Pi." In Proc. 978-15386-5257-2/18/\$31.0002018IEEE.
- Paul Viola. Michael Jones "Rapid Object Detection using a Boosted Cascade of Simple Features" in Proc. 0-7695-12720/01 \$10.00 0 2001 IEEE.
- Vinit Jain, Soniya Chawla "IMPLEMENTATION OF A SMART SAFETY AND SECURITY DEVICE USING RASPBERRY PI, TELEGRAM BOT, PROTA OS AND MANYTHING WEB SERVICE" in Proc. International Journal of Computer Engineering and Applications, Volume XII, Issue II, Feb. 18, www.ijcea.com ISSN 2321-3469.
- Zhang, Y., Sun, J., Wang, Z., and Fang, H. A smart home system based on the ESP32 and the MQTT protocol. IEEE Access, vol. 7, pp. 163806-163813
- S. Liu, Y. Liu, X. Yan, B. Zhang, and X. Chen. Smart home control based on the ESP32 and MQTT protocol. Conference Series, Journal of Physics, 1897(1), 012017.
- Zhao, X., Zhang, L., and C. Liu (2020). Design and implementation of an ESP32-based smart home system, 112- 115 in 2020 2nd International Conference on Electronic Information Technology and Computer Engineering (EITCE).
- Y. Wang, J. Zhang, G. Zhao, Y. Wang, and Y. Jia (2020). Based on ESP32 and IoT, this intelligent smart home solution. IEEE 5th International Conference on Information Technology and Mechatronics Engineering (ITOEC), 436- 439, 2020.

- J. Wu, L. Wu, L. Liu, and Y. Zhang (2020). Based on the ESP32 microcontroller and Amazon Alexa, this smart home system. IEEE Access, vol. 8, pp. 231103-231109.
- S. H. Kim and S. W. Lee, "ESP32-CAM based home security system with mobile push notifications," 2019 IEEE 22nd International Conference on Intelligent Transportation Systems (ITSC), 2019, pp. 2019-2024. doi: 10.1109/ITSC.2019.8917182
- Anjali Shrivastva, (2023). Research Paper for Smart Home Automation System using ESP32 with Blynk, IR Remote & manual control, IOT project DOI 10.17148/IJE REEICE. 2020.9565
- S. Kumar," Ubiquitous Smart home System Using Android Application, "International Journal of Computer Networks & Communications, vol. 6, pp. 33-43, January 2014.
- D Abhilash, Chandrashekar, and S Shalini. 2017. Economical, energy efficient and portable home security system based on Raspberry Pi 3 using the concepts of OpenCV and MIME. In 2017 International Conference on Circuits, Controls, and Communications (CCUBE). IEEE, 60–64.
- A Anitha. 2017. Home security system using internet of things. In IOP conference series: materials science and engineering, Vol. 263. IOP Publishing, 042026.
- Suraj Pawar, Vipul Kithani, Sagar Ahuja, and Sunita Sahu. 2018. Smart home security using IoT and face recognition. In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBE). IEEE, 1–6.

CHAPTER 8

APPENDIX

```
import RPi.GPIO as GPIO

import telepot

import time

import cv2

import cv2

import time

vs = cv2.VideoCapture(0)

time.sleep(2)

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

buz=26

pir=17

led=2

lock1=3

lock2=4

GPIO.setup(pir,GPIO.IN)

GPIO.setup(buz,GPIO.OUT)

GPIO.setup(led,GPIO.OUT)

GPIO.setup(lock1,GPIO.OUT)

GPIO.setup(lock2,GPIO.OUT)

GPIO.output(lock1,0)
```



```
GPIO.output(lock2,0)
```

```
GPIO.output(buz,0)
```

```
GPIO.output(led,0)
```

```
def handle(msg):
```

```
    global telegramText
```

```
    global chat_id
```

```
    global receiveTelegramMessage
```

```
    chat_id = msg['chat']['id']
```

```
    telegramText = msg['text']
```

```
    print("Message received from " + str(chat_id))
```

```
    if(telegramText=='/lock'):
```

```
        GPIO.output(lock1,0)
```

```
        GPIO.output(lock2,0)
```

```
        bot.sendMessage(chat_id, "door locked")
```

```
    if(telegramText=='/unlock'):
```

```
        GPIO.output(lock1,1)
```

```
        GPIO.output(lock2,0)
```

```
        bot.sendMessage(chat_id, "door unlocked")
```

```
def capture():
```

```
    print("Sending photo to " + str(chat_id))
```

```
    bot.sendPhoto(chat_id, photo = open('./image.jpg', 'rb'))
```

```
bot = telepot.Bot('7039353778:AAH6gr1pmjNGGryYjsS1KZ-5_u809qtQuz0')
```

```
chat_id='1073239516'

bot.message_loop(handle)

print('Telegram bot is ready')

bot.sendMessage(chat_id, 'BOT STARTED')

time.sleep(2)

while True:

    (grabbed, frame) = vs.read()

    cv2.imshow('input',frame)

    cv2.waitKey(1)

    if(GPIO.input(pir)==1):

        print("MOTION DETECTED")

        GPIO.output(buz,1)

        cv2.imwrite('image.jpg',frame)

        cv2.waitKey(1)

        capture()

        time.sleep(1)

        GPIO.output(buz,0)

        ii=0

        while(ii<15):

            ii=ii+1

            GPIO.output(led,0)

            time.sleep(0.5)

            GPIO.output(led,1)
```

```
time.sleep(0.5)
```

```
else:
```

```
print("NO MOTION");
```

```
time.sleep(2)
```