A Comprehensive Anti-Theft Vehicle Protection Framework Using Embedded Electronics And Python

A Project report

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

PINAPALA SAI PAVAN	**(21JR1A04F9)**
LOYA RAVI TEJA	**(21JR1A04G7)**
VENNAPUSA CHINNA LINGAREDDY	**(21JR1A04J0)**
TATANABOINA JOHNY	**(21JR1A04I2)**

Under the guidance of

Mr. K. MALLIKARJUNA RAO M. Tech, (PhD)

Assistant Professor, Dept. of ECE

DEPARTMENT OF
ELECTRONICS AND COMMUNICATION ENGINEERING

KKR & KSR INSTITUTE OF TECHNOLOGY AND SCIENCES
(Autonomous)
Vinjanampadu (V), Vatticherukuru (M), Guntur (Dt), A.P-522017.

APRIL – 2025.

KKR & KSR INSTITUTE OF TECHNOLOGY AND SCIENCES
(Autonomous)
Department of Electronics And Communication Engineering
(Approved by AICTE New Delhi || Permanently Affiliated to JNTUK, Kakinada) ||
Accredited with 'A' Grade by NAAC || NBA Accreditation)
Vinjanampadu (V), Vatticherukuru (M), Guntur (Dt), A.P-522017.

CERTIFICATE

This is to certify that this project report entitled "**A Comprehensive Anti-Theft Vehicle Protection Framework Using Embedded Electronics And Python**" submitted by

21JR1A04F9	**PINAPALA SAI PAVAN**
21JR1A04G7	**LOYA RAVI TEJA**
21JR1A04J0	**VENNAPUSA CHINNA LINGAREDDY**
21JR1A04I2	**TATANABOINA JOHNY**

to **Jawaharlal Nehru University Kakinada**, through **KKR & KSR Institute of Technology and Sciences** for the award of the Degree of Bachelor of Technology in **Electronics And Communication Engineering** is a bonafide record of project work carried out under the supervision of **Mr. K. Mallikarjuna Rao** during the year 2024-2025.

Mr. K. MALLIKARJUNA RAO
SUPERVISOR
(Assistant Professor)

Dr. N. ADINARAYANA
HEAD OF THE DEPARTMENT

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

We here by declare that the project "**A Comprehensive Anti-Theft Vehicle Protection Framework Using Embedded Electronics And Python**" has been carried out by our team and this work has been submitted to **KKR & KSR Institute of Technology and Sciences** (A), Vinjanampadu, affiliated to **Jawaharlal Nehru Technological University, Kakinada** in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Electronics And Communication Engineering.

We further declare that this project work has not been submitted in full or part for the award of any other degree in any other educational institutions.

PINAPALA SAI PAVAN	**(21JR1A04F9)**
LOYA RAVI TEJA	**(21JR1A04G7)**
VENNAPUSA CHINNA LINGAREDDY	**(21JR1A04J0)**
TATANABOINA JOHNY	**(21JR1A04I2)**

ACKNOWLEDGEMENT

We would like to express our profound gratitude towards **Mr. K. MALLIKARJUNA RAO**, Department of **Electronics and Communication Engineering**, who played a supervisory role to utmost perfection, enabled us to seek through our IV-II B. Tech project and for guidance as an internal guide methodically and meticulously.

We express our gratitude towards all the faculty members and non-teaching faculty members, the **Electronics And Communication Engineering**.

We are highly indebted to **Dr. N. ADI NARAYANA** Head of the Department, **Electronics and Communication Engineering** for providing us with all the necessary support.

We render our deep sense of gratitude to **Dr. P. BABU**, Principal and **Dr. K . HARI BABU**, Director Academics for permitting us to carry out our main project works. We would like to express our sincere thanks to **Electronics and Communication Engineering** staff for lending us their time to help us complete the work successfully.

We are very much thankful to the college management **Mr. K . SUBBARAO Mr. K . SHEKAR** for their continuous support and the facilities provided. We would also like to thank our staff, parents, and friends for their enduring encouragement and assistance whenever required.

KKR & KSR INSTITUTE OF TECHNOLOGY AND SCIENCES
(Autonomous)
(Approved by AICTE New Delhi || Permanently Affiliated to JNTUK, Kakinada) || Accredited with 'A' Grade by NAAC || NBA Accreditation)

INSTITUTE VISION AND MISSION

INSTITUTION VISION

To produce eminent and ethical Engineers and Managers for society by imparting quality professional education with an emphasis on human values and holistic excellence.

INSTITUTION MISSION

- To incorporate bench marked teaching and learning pedagogues in the curriculum.
- To ensure the all-around development of students through a judicious blend of curricular, co- curricular, and extra-curricular activities.
- To support the cross-cultural exchange of knowledge between industry and academy.
- To provide higher/continued education and research opportunities to the employees of the institution.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION

VISION OF THE DEPARTMENT

To become a reputed center in Electronics and Communication engineering for quality, competency, and social responsibility.

MISSION OF THE DEPARTMENT

- Strengthen the core competence with vibrant technological education in a congenial environment.
- Promote innovative research and development for the economy, society, and environment.
- Inculcate professional behaviour, and strong ethical values to meet the challenges in collaboration and lifelong learning.

PROGRAM SPECIFIC OUTCOMES (PSO's)

PSO1: Professional Skills:

Apply Electronics, Communications, VLSI, Embedded systems knowledge to arrive cost effective and appropriate solutions.

PSO2: Problem-solving skills:

Able to provide solutions and design Semiconductor Devices, Digital Systems, Microprocessor and Signal processing for the fields of Consumer Electronics, Medical, Defense and Spacecraft Electronics industry

PSO3: Successful career:

Able to use latest hardware and software tools like VHDL, MATLAB, MULTISIM, MENTOR GRAPHICS along with analytical skills.

PSO4: Exposure for research and development:

To analyze latest trends in Communication and apply the knowledge for the improvement in the present technology by doing research through higher education.

PROGRAM EDUCATIONALOBJECTIVES(PEO's)

PEO1:

Develop a strong background in basic science and mathematics and ability to use these tools in their chosen fields of specialization.

PEO2:

Have the ability to demonstrate technical competence in the fields of electronics and communication engineering and develop solutions to the problems.

PEO3:

Attain professional competence through life-long learning such as advanced degrees, professional registration, and other professional activities.

PEO4:

Function effectively in a multi-disciplinary environment and individually, within a global, societal, and environmental context.

PEO5:

Take individual responsibility and to work as a part of a team towards the fulfillment of both individual and organizational goals.

PROGRAM OUTCOMES (PO's)

1. **Engineering knowledge:**

   Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis:**

   Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using the first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions:**

   Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems:**

   Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage:**

   Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society:**

   Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:**

Understand the impact of professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics:**

Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:**

Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication:**

Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:**

Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning:**

Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

COURSE OUTCOMES (CO'S)

CO409.1: Perform a system of examinations to identify problems.

CO409.2: Review the literature/Related work.

CO409.3: Defining the problem & its area of domain.

CO409.4: Proposal of solution for the selected area/methodology.

CO409.5: Analysis of the proposed work & documentation.

CO409.6: Acquire collaborative learning, leadership qualities & presentation skills

COURSE OUTCOMES - PROGRAM OUTCOMESMAPPING

Course Code	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO409.1	2	3		2		2	2					
CO409.2	3	2	3	3		3			2	2	3	
CO409.3	3	3	3	3	2	2						1
CO409.4	3	3	2	2	3		2	2	1	3		
CO409.5	3		3	2	2		2		2		1	1
CO409.6	1					3				2		2
Average	3	3	3	3	3	3	2	2	2	3	2	2

3: High 2: Medium 1: Low

COURSE OUTCOMES - PROGRAM SPECIFIC OUTCOMES MAPPING

CO code	PSO1	PSO2	PSO3	PSO4
CO409.1	3			
CO409.2	3	2	1	
CO409.3	3		2	2
CO409.4	3		2	
CO409.5	3	2		
CO409.6		3		
Average	**3**	**2**	**2**	**2**

High 2: Medium 1: Low

PROGRAM EDUCATIONAL OBJECTIVES –PROGRAM SPECIFIC OUTCOMES MAPPING

	PSO 1	PSO2	PSO3	PSO4
PEO1	2	1	3	1
PEO2		3	2	2
PEO3	1	2	3	3
PEO4	3		2	2
PE05	1	3	2	
Average	1	2	2	2

3: High 2: Medium 1: Low

CO-PO'S MAPPING WITH REASONS

1. **CO409.1** is mapped with PO1, PO2 and PO4, PO6, PO7 as basic knowledge of Engineering and problem Analysis activities are highly essential to conduct examinations on existing systems which have been using in industries as a part of and to define the problem of the proposed system.
2. **CO409.2** is mapped with PO1, PO2, PO3, PO4 and PO6, PO9, PO10, and PO11 for identification, gathering analysis, and classification of requirements for the proposed system, basic knowledge of engineering and Analysis steps along with complex problem analysis through the efforts of teamwork in order to meet the specific needs of the customer.
3. **CO409.3** is mapped with PO1, PO2, PO3, PO4, PO5, PO6 and PO12 as to conduct the literature review and to examine the relevant systems to understand and identify the merits and demerits of each to enhance and develop the proposed as per the need.
4. **CO409.4** is mapped with PO1, PO2, PO3, PO4, PO5, PO7, PO8, PO9, and PO10 because modularization and design of the project are needed after requirements elicitation. For modularization and design of the project, Basic knowledge of Engineering, Analysis capabilities, Design skills, and communication is needed between team members as different modules are designed individually before integration.
5. **CO409.5** is mapped with PO1, PO3, PO4, PO5, PO7, PO9, PO11 and PO12 as to construct the project latest technologies are needed. The development of the project is done individually and in groups with well-defined communication by using engineering and management principles.
6. **CO409.6** is mapped with PO1, PO6, PO10, and PO12 because during and after completion of the project, documentation is needed along with proper methods of presentation through understanding and application of engineering and management principles, which in turn needs well-defined communication between the team members with all the ethical values. Even the project development team defines future enhancements as a part of the project development after identifying the scope of the project.

CO-PSO'S MAPPING WITH REASONS

1. **CO409.1** is mapped with **PSO1** as examining existing systems and identification of the problem is a part of Application Development activity and identification of evolutionary changes in the latest technologies.

2. **CO409.2** is mapped with **PSO1, PSO2,** and **PSO3** as identifying and classifying the requirements is a part of Application development and evolutionary computing changes and also follows ethical principles.

3. **CO409.3** is mapped with **PSO1 and PSO3, PSO4** as a review of literature is a part of application development activity by recognizing the computing technologies and their evolutionary changes.

4. **CO409.4** is mapped with **PSO1 and PSO3** because modularization and logical design is also a part of Application development and follows computing changes using Deep learning technology.

5. **CO409.5** is mapped with **PSO1 and PSO2** as Testing, Development, and Integration of project activities are part of Application development and follow ethical principles.

6. **CO409.6** is mapped with **PSO2** for project documentation and presentation the project team members apply the professional and leadership quality

CONTENTS

Chapter No.	Title	Page No.
	Cover Page	**i**
	Certificate	**ii**
	Declaration	**iii**
	Acknowledgement	**iv**
	Outcomes	**v-xiii**
	Contents	**xiv-xvi**
	List of Tables	**xvii**
	List of Figures	**xviii-xix**
	List of Abbreviations	**xx**
	Abstract	**xxi**
1	**Introduction**	**1-7**
	1.1 Overview	1-2
	1.2 Embedded system implementation	2
	1.2.1 Introduction	2
	1.2.2 User interface	2
	1.3 Embedded system	3-6
	1.3.1 Embedded System Hardware	3-4
	1.3.2 Embedded System Software	5-6
	1.4 Applications	6-7
	1.5 Implementation flow	7
2	**Literature survey**	**8-12**
	2.1 Introduction	8
	2.2 Vehicle Anti-Theft Systems	8-11
	2.2.1 Traditional Security Measures	8-9

	2.2.2 GPS-Based Tracking Systems	9-10
	2.2.3 GSM-Based Alert Systems	10-11
	2.3 Biometric Authentication in Vehicle Security	12
	2.3.1 Fingerprint-Based Authentication	12
3	**Existing work with references**	**13-16**
	3.1 Introduction	13
	3.2 Drawbacks	13-16
4	**Proposed work**	**17-84**
	4.1 Introduction	17
	4.2 Block Diagram	18
	4.3 Flow Chart	18-19
	4.4 Circuit Diagram	20
	4.5 Advantages	20-23
	4.6 Applications	23-27
	4.7 Hardware Description	28-73
	4.7.1 ESP32	28-31
	4.7.2 LCD	32-39
	4.7.3 Finger Print Module	39-48
	4.7.4 GPS	48-53
	4.7.5 GSM/GPRS Module	53-59
	4.7.6 Buzzer	59-60
	4.7.7 Vibration Sensor	60-61
	4.7.8 Relay	62-69
	4.7.9 Mosquito Bat Module	69-71
	4.7.10 DC Gear Motor	71-73
	4.8 Software Description	74-83
	4.8.1 Arduino IDE	74-79

	4.8.2 Libraries	80-83
5	**Testing and Result Analysis**	**84-93**
	5.1 Introduction	84
	5.2 Testing Methodology	84-85
	5.3 Test Cases and Observations	85-87
	5.3.1 Fingerprint Authentication Testing	85
	5.3.2 Vibration Sensor and Shock Detection	85-86
	5.3.3 GPS and GSM Module Testing	86
	5.3.4 Camera and Image Capture Testing	86-87
	5.4 Performance Analysis	87
	5.5 Challenges and Solutions	87-88
	5.6 Project Output Images	88-92
	5.7 Conclusion	92
6	**Conclusions**	**93**
7	**Scope for Future work**	**94**
8	**Bibliography**	**95**
9	**Code**	**96-110**

LIST OF TABLES

Table No	Name of the Table	Page No
Table 4.1(a)	Pin Description of LCD	34-35
Table 4.1(b)	Command codes for LCD	36-37
Table 4.1(c)	Control and display commands of LCD	37-39
Table 4.2(a)	Pin Description of Fingerprint Module	41
Table 4.2(b)	Fingerprint authentication performance	42
Table 4.2(c)	Other Specifications of Fingerprint Module	42
Table 4.2(d)	Interface of Fingerprint Module	43
Table 4.2(e)	Connector Specifications of Fingerprint Module	43
Table 4.2(f)	Flash storage structure	43
Table 4.2(g)	System parameters storage area	44
Table 4.2(h)	Command specifications of Fingerprint module	44-46
Table 4.2(i)	01 command packet format	46
Table 4.2(j)	02 data packet format	46
Table 4.2(k)	08 last data packet	46
Table 4.2(l)	Response packet format	47
Table 4.3	Buzzer Pin Configuration	59
Table 4.4	Components of the Mosquito Module	70
Table 4.5(a)	Components of DC Gear Motor	72
Table 4.5(b)	Technical Specifications	72
Table 4.5(c)	Wiring and Connection	73
Table 5.1	Fingerprint Authentication Testing	85
Table 5.2	Vibration Sensor and Shock Detection	85-86
Table 5.3	GPS and GSM Module Testing	86
Table 5.4	Camera and Image Capture Testing	86-87
Table 5.5	Challenges and Solutions	87-88

LIST OF FIGURES

Figure No	Name of the Figure	Page No
Fig 1.1	Overview of embedded system	2
Fig 1.2	Block diagram of embedded system	3
Fig 1.3	Flow of burning source code to processor	6
Fig 4.1	Block Diagram of the Proposed System	18
Fig 4.2(a)	Flow chart of Authorization using Finger Print	18
Fig 4.2(b)	Flow chart of Unauthorized Access	19
Fig 4.2(c)	Flow chart of Remote Access	19
Fig 4.2(d)	Circuit Diagram of Proposed System and Connections	20
Fig 4.3(a)	ESP Microcontroller	28
Fig 4.3(b)	WiFi ESP-WROOM-32 Processor	31
Fig 4.4(a)	LCD – Front View	33
Fig 4.4(b)	LCD – Back View	34
Fig 4.4(c)	LCD Pin Diagram	34
Fig 4.4(d)	Block Diagram of LCD	37
Fig 4.5(a)	Fingerprint Module	40
Fig 4.5(b)	Architecture Of Fingerprint Module	40
Fig 4.5(c)	Communication Architecture of Fingerprint Authentication Unit	41
Fig 4.5(d)	Main board of Fingerprint sensor	47
Fig 4.5(e)	Sensor	48
Fig 4.6(a)	GPS Module	48
Fig 4.6b)	Calculation of GPS Receiver	49
Fig 4.6(c)	Time diff of Satellite Signal & GPS receiver Signal	50
Fig 4.6(d)	Pin Description of GPS Module	51
Fig 4.6(e)	Checking of GPS by connecting to PC	52
Fig 4.6(f)	Output data from GPS receiver module	53
Fig 4.7	GPRS Module(GSM)	53
Fig 4.8	Buzzer	59

Fig 4.9	Vibration Sensor	61
Fig 4.10(a)	Relay Module	62
Fig 4.10(b)	Pin diagram of Relay Module	62
Fig 4.10(c)	Actual Design of simple Relay	63
Fig 4.10(d)	Working of Relay	64
Fig 4.10(e)	Relay Basics	65
Fig 4.10(f)	Energized Relay (ON)	66
Fig 4.10(g)	De-Energized Relay (OFF)	66
Fig 4.11	Mosquito Module	70
Fig 4.12	Dc Gear Motor	72
Fig 4.13(a)	Arduino Ide Dashboard	75
Fig 4.13(b)	File Menu	76
Fig 4.13(c)	Preference Section	76
Fig 4.13(d)	Output Pane	77
Fig 4.13(e)	Tool bar Buttons	77
Fig 4.13(f)	Serial Monitor	78
Fig 4.13(g)	Simple Text Editor	79
Fig 4.13(h)	Output pane Showing Compilation	79
Fig 4.14(a)	Sketch Menu and Libraries	80
Fig 4.14(b)	Different Arduino Boards	81
Fig 4.14(c)	Serial Port Selection in Tools Menu	82
Fig 4.14(d)	Boot Loader	83
Fig 5.1	Final Prototype Image	88
Fig 5.2	Circuit diagram	89
Fig 5.3	Fingerprint Sensor is Ready to Take Input	89
Fig 5.4	Unauthorized Access & Buzzer is Activated	90
Fig 5.5	Unauthorized Access Detected, Image Captured & Mail Sent	90
Fig 5.6	Mail Received for Unauthorized Access	91
Fig 5.7	Vibration Detected and Alert Message Sent	91
Fig 5.8	Alert and Location Recieved	92

LIST OF ABBREVIATIONS

IoT	Internet of Things
ESP32	Espressif Systems 32-bit Microcontroller
DC	Direct Current
GPS	Global Positioning System
GSM	Global System for Mobile Communications
SMS	Short Message Service
Wi-Fi	Wireless Fidelity
LCD	Liquid Crystal Display
PWM	Pulse Width Modulation
RFID	Radio Frequency Identification
AI	Artificial Intelligence
ML	Machine Learning
OTP	One-Time Password
UART	Universal Asynchronous Receiver-Transmitter
GPIO	General-Purpose Input/Output
LED	Light Emitting Diode
USB	Universal Serial Bus
ROM	Read-Only Memory
RAM	Random Access Memory

ABSTRACT

Vehicle theft remains a significant concern, necessitating the development of robust security systems to deter unauthorized access and enhance real-time monitoring. This project, "A Comprehensive Anti-Theft Vehicle Protection Framework Using Embedded Electronics and Python," presents an advanced security solution integrating multiple sensors, a display module, and communication components to ensure vehicle safety.

The system employs an ESP32 microcontroller as the core processing unit, interfacing with various sensors and modules. A fingerprint authentication system is used for authorized access, ensuring only registered users can operate the vehicle. For theft detection, a vibration sensor monitors unauthorized movement or tampering. If suspicious vibrations are detected, the system immediately triggers an alarm, displays a warning on the screen, and initiates countermeasures.

A camera module is deployed to capture images of the potential intruder, which are then stored and sent via email using Wi-Fi connectivity. A dedicated display module (such as an OLED or LCD) provides real-time system status, including access verification, security alerts, and sensor data, ensuring users are informed of any suspicious activities.
To enhance security, a mosquito module is activated upon detecting unauthorized access, deterring intruders using ultrasonic sound waves.

For vehicle immobilization and control, a DC motor is engaged when authorized access is granted, allowing the vehicle to start. Additionally, the system integrates GSM and GPS modules to send SMS alerts with the vehicle's real-time location in case of unauthorized access. The GSM module also enables remote vehicle control, allowing users to deactivate or activate the DC motor via SMS commands.

Furthermore, a buzzer and the display module work together to notify nearby individuals and the owner of a security breach, enhancing situational awareness.

By combining biometric authentication, motion detection, image capture, a real-time display interface, remote communication, and real-time alerts, this system ensures comprehensive protection against vehicle theft. This innovative approach leverages IoT and embedded electronics to enhance vehicle safety and user control.

CHAPTER-1

INTRODUCTION

1.1 Overview

With the rise in vehicle theft cases worldwide, there is a growing need for advanced security systems to prevent unauthorized access and theft. Traditional locking mechanisms and alarm systems often fail to provide sufficient protection, making biometric authentication and IoT-based monitoring crucial for modern vehicle security. This project introduces an IoT-enabled Vehicle Anti-Theft System, which integrates a fingerprint sensor for authentication, a vibration sensor for theft detection, a GPS module for real-time tracking, and a GSM module for instant alerts. The ESP32 microcontroller serves as the core processing unit, efficiently managing authentication, sensor inputs, and communication with external devices.

The system primarily relies on a fingerprint sensor as a key security feature, ensuring that only authorized users can start the vehicle. If an unauthorized attempt is detected, the system triggers an alert buzzer and captures an image using a camera connected to a Python-enabled PC. This image is then sent via email to the vehicle owner or a security system for further monitoring. Additionally, a vibration sensor continuously monitors any unexpected movements, such as forced entry, towing, or break-in attempts. If any unusual activity is detected, the system immediately sends an SMS alert containing the vehicle’s GPS location to a registered mobile number via the GSM module, enabling real-time tracking and a quick response.

To further enhance security, the system can be configured to automatically disable the vehicle's ignition system if multiple failed authentication attempts are detected. Integration with cloud-based monitoring platforms allows real-time data logging, making it possible for vehicle owners or law enforcement agencies to track and analyze security breaches. Additionally, the system supports remote access through a mobile application or web interface, enabling users to receive alerts, view live location updates, and even remotely disable the vehicle in case of a theft attempt.

By combining biometric authentication, IoT-based alerts, GPS tracking, and real-time monitoring, this project provides a cost-effective, reliable, and scalable

solution to vehicle security concerns. The ESP32 microcontroller ensures efficient data processing and communication, making it suitable for integration with AI-based predictive security systems that can detect suspicious behaviour patterns. This system not only enhances vehicle security but also offers a smart, automated approach to theft prevention, significantly reducing the risk of vehicle loss and improving overall safety.

1.2 Embedded system implementation

1.2.1 Introduction:

An embedded system is one kind of a computer system mainly designed to perform several tasks like to access, process, and store and also control the data in various electronics-based systems. Embedded systems are a combination of hardware and software where software is usually known as firmware that is embedded into the hardware. One of its most important characteristics of these systems is, it gives the o/p within the time limits. Embedded systems support to make the work more perfect and convenient. So, we frequently use embedded systems in simple and complex devices too. The applications of embedded systems mainly involve in our real life for several devices like microwave, calculators, TV remote control, home security and neighborhood traffic control systems, etc.

1.2.2 User interface:

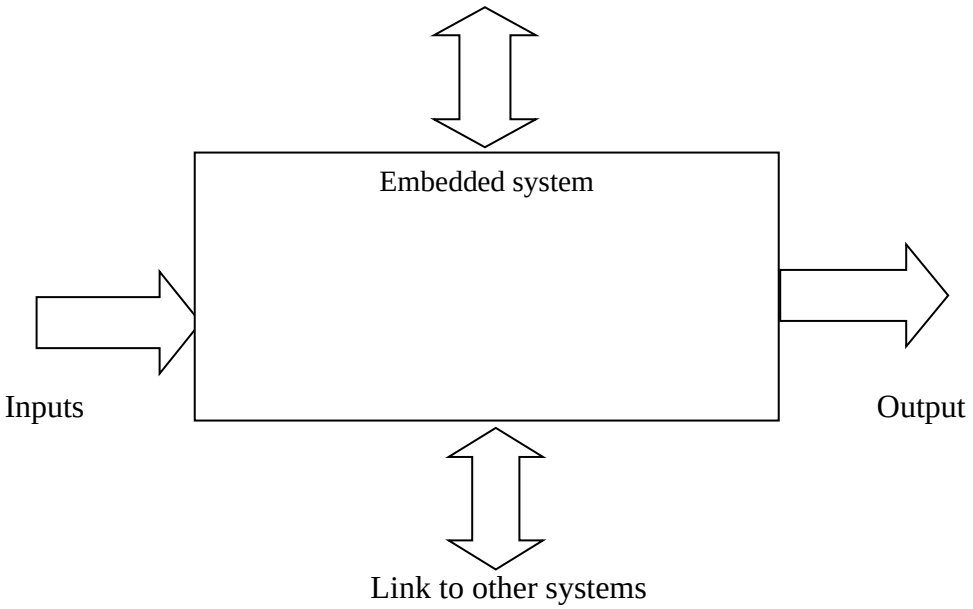

Fig1.1: Overview of embedded system

1.3 Embedded system:

Embedded system includes mainly two sections, they are

1. Hardware
2. Software

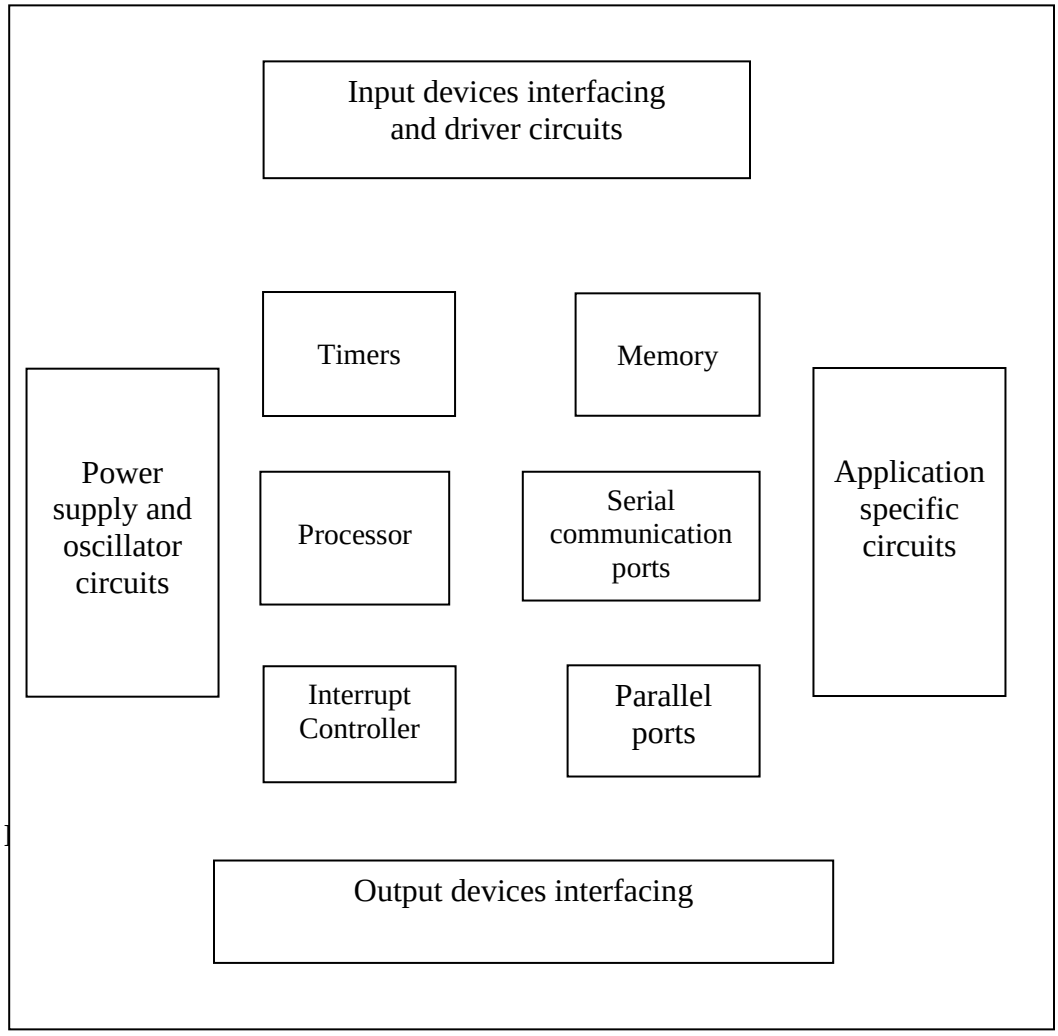

Fig.1.2: Block diagram of embedded system

1.3.1 Embedded System Hardware:

As with any electronic system, an embedded system requires a hardware platform on which it performs the operation. Embedded system hardware is built with a microprocessor or microcontroller. The embedded system hardware has elements like

input output (I/O) interfaces, user interface, memory and the display. Usually, an embedded system consists of:

- Power Supply
- Processor
- Memory
- Timers
- Serial communication ports
- Output/Output circuits
- System application specific circuits

Embedded systems use different processors for its desired operation. Some of the processors used are

1. Microprocessor
2. Microcontroller
3. Digital signal processor

Microprocessor vs. Microcontroller

Microprocessor

- CPU on a chip.
- We can attach required amount of ROM, RAM and I/O ports.
- Expensive due to external peripherals.
- Large in size
- general-purpose

Microcontroller

- Computer on a chip
- fixed amount of on-chip ROM, RAM, I/O ports
- Low cost.
- Compact in size.
- Specific –purpose

1.3.2 Embedded System Software:

The embedded system software is written to perform a specific function. It is typically written in a high level format and then compiled down to provide code that can be lodged within a non-volatile memory within the hardware. An embedded system software is designed to keep in view of the three limits:

- Availability of system memory
- Availability of processor's speed
- When the system runs continuously, there is a need to limit power dissipation for events like stop, run and wake up.

Bringing software and hardware together for embedded system:

To make software to work with embedded systems we need to bring software and hardware together .for this purpose we need to burn our source code into microprocessor or microcontroller which is a hardware component and which takes care of all operations to be done by embedded system according to our code.

Generally we write source codes for embedded systems in assembly language, but the processors run only executable files. The process of converting the source code representation of your embedded software into an executable binary image involves three distinct steps:

1. Each of the source files must be compiled or assembled into an object file.
2. All of the object files that result from the first step must be linked together to produce a single object file, called the re-locatable program.
3. Physical memory addresses must be assigned to the relative offsets within the re-locatable program in a process called relocation.

The result of the final step is a file containing an executable binary image that is ready to run on the embedded system.

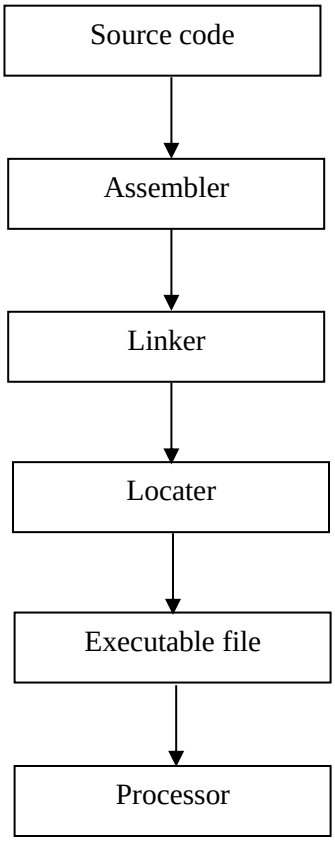

Fig.1.3: Flow of burning source code to processor

1.4 Applications:

Embedded systems have different applications. A few select applications of embedded systems are smart cards, telecommunications, satellites, missiles, digital consumer electronics, computer networking, etc.

Embedded Systems in Automobiles

- Motor Control System
- Engine or Body Safety
- Robotics in Assembly Line
- Mobile and E-Com Access

Embedded systems in Telecommunications

- Mobile computing

- Networking
- Wireless Communications

Embedded Systems in Smart Cards

- Banking
- Telephone
- Security Systems

1.5 Implementation flow:

Stage 1:

Considering the problems of existing methods and giving solution to that problem by considering the basic requirements for our proposed system

Stage 2:

Considering the hardware requirement for the proposed system

For this we need to select the below components:

1. Microcontroller
2. Inputs for the proposed system (ex: sensors, drivers etc..,)
3. Outputs (ex: relays, loads)

Stage 3:

After considering hardware requirements, now we need to check out the software requirements. Based on the microcontroller we select there exists different software for coding, compiling, debugging. we need to write source code for that proposed system based on our requirements and compile, debug the code in that software .

After completing all the requirements of software and hardware we need to bring both together to work our system. For this we need to burn our source code into microcontroller, after burning our source code to microcontroller then connect all input and output modules as per our requirement.

CHAPTER 2

LITERATURE SERVEY

2.1. Introduction

Vehicle theft is a persistent global issue, leading to financial losses and security threats. Over the years, researchers and engineers have explored various technologies, including biometric authentication, GPS tracking, GSM communication, and IoT-based monitoring systems, to mitigate vehicle theft. This literature survey reviews existing works related to vehicle security systems, identifying their advantages, limitations, and potential areas for improvement. The following sections present an in-depth analysis of relevant research papers, categorized by security approach.

2.2 Vehicle Anti-Theft Systems

2.2.1 Traditional Security Measures

Paper 1: Johnson et al. (2015) - Evaluating the Effectiveness of Mechanical Locks and Alarm Systems

- **Objective:** This study investigates the reliability of mechanical security devices, including steering wheel locks, gear locks, and traditional alarm systems.
- **Experimental Approach:** The researchers conducted controlled vehicle break-in tests using commonly available lock-picking tools and techniques. They evaluated response times, success rates, and vulnerability factors.
- **Observations:**
    - Steering wheel locks resisted picking for an average of 8 minutes but were eventually bypassed.
    - Alarm systems successfully alerted owners but failed to deter professional thieves.
    - False alarms were frequent, leading to decreased public responsiveness.
- **Advantages:**
    - Cost-effective and widely available.
    - Basic deterrence against opportunistic theft.

- **Limitations:**
  - Easily bypassed by experienced criminals.
  - False alarms reduce effectiveness over time.
- **Conclusion:** Traditional mechanical and alarm-based security measures provide limited protection. More advanced solutions, such as biometric authentication and real-time tracking, are required for comprehensive security.

2.2.2 GPS-Based Tracking Systems

Paper 2: Smith and Patel (2018) - Real-Time GPS-Based Vehicle Tracking for Theft Prevention

- **Objective:** To evaluate the effectiveness of GPS-based tracking systems in stolen vehicle recovery and prevention.
- **Experimental Approach:**
  - The study involved equipping 50 vehicles with GPS trackers and monitoring their movements over six months.
  - Simulated theft scenarios were conducted, where designated testers attempted to steal and relocate the vehicles.
- **Observations:**
  - 85% of stolen vehicles were successfully recovered within 24 hours.
  - GPS jammers were used in 10% of cases, rendering the system ineffective.
- **Advantages:**
  - Provides real-time location tracking.
  - Assists law enforcement in quick recovery.
- **Limitations:**
  - Susceptible to GPS signal jamming.
  - Relies on external network infrastructure.
- **Conclusion:** GPS tracking improves vehicle recovery rates but must be combined with additional security features to prevent theft from occurring in the first place.

Paper 3: Gupta et al. (2019) - Geofencing-Enhanced GPS Vehicle Security System

- **Objective:** To enhance GPS tracking by implementing geofencing alerts that notify owners when a vehicle moves beyond predefined zones.
- **Experimental Approach:**
  - Developed an embedded GPS system with mobile app integration.
  - Tested system effectiveness in urban and rural areas with different geofencing parameters.
- **Observations:**
  - The system effectively alerted users when vehicles exited restricted areas.
  - False alarms occurred in high-traffic areas with GPS signal fluctuations.
- **Advantages:**
  - Adds an additional layer of theft prevention.
  - Enables real-time monitoring and notifications.
- **Limitations:**
  - Requires stable mobile network connectivity.
  - Incorrect geofencing configuration may lead to unnecessary alerts.
- **Conclusion:** Geofencing is a valuable security addition but must be fine-tuned to reduce false alerts.

2.2.3 GSM-Based Alert Systems

Paper 4: Chen et al. (2020) - SMS-Based Alert Systems for Vehicle Security

- **Objective:** To explore the potential of GSM-based alerts in preventing vehicle theft.
- **Experimental Approach:**
  - Developed an Arduino-based system with GSM modules and motion sensors.
  - Conducted real-world testing by simulating unauthorized access attempts.
- **Observations:**
  - Successfully sent alerts in 98% of cases but had delays in weak network

zones.

  - Dependence on GSM networks was identified as a major limitation.

- **Advantages:**
  - Instant alerts provide a quick response mechanism.
  - Works effectively in most urban environments.
- **Limitations:**
  - Relies on the availability of mobile networks.
  - Cannot physically prevent theft.
- **Conclusion:** GSM alerts are a useful notification tool but should be combined with active theft prevention mechanisms.

Paper 5: Das et al. (2021) - Remote Vehicle Immobilization via GSM

- **Objective:** To implement remote vehicle disabling through GSM-based commands.
- **Experimental Approach:**
  - Built a prototype that interfaced with a car's ignition system via a relay circuit controlled by SMS commands.
  - Tested in multiple real-world environments.
- **Observations:**
  - Successfully disabled the vehicle in 95% of cases.
  - Network latency led to a delay in activation in 12% of cases.
- **Advantages:**
  - Owners can remotely disable stolen vehicles.
  - Works with any GSM-enabled mobile phone.
- **Limitations:**
  - Susceptible to signal interference and jamming.
  - Requires owner intervention to function.
- **Conclusion:** GSM-based remote immobilization enhances security but should be supplemented with automatic disabling mechanisms.

2.3. Biometric Authentication in Vehicle Security

2.3.1 Fingerprint-Based Authentication

Paper 6: Lee and Wong (2016) - Fingerprint-Based Ignition System for Enhanced Security

- **Objective:** To prevent unauthorized vehicle access using fingerprint recognition.
- **Experimental Approach:**
  - Implemented a fingerprint scanner in a prototype ignition system.
  - Conducted access tests with authorized and unauthorized users.
- **Observations:**
  - 90% reduction in unauthorized access attempts.
  - Failure rate increased in dusty and wet conditions.
- **Advantages:**
  - Provides strong authentication and prevents key theft.
- **Limitations:**
  - Higher cost compared to traditional systems.
  - Environmental factors can affect fingerprint recognition accuracy.
- **Conclusion:** Fingerprint authentication significantly enhances security but requires improvements in sensor durability.

CHAPTER 3
EXISTING WORK WITH REFERENCES

3.1 INTRODUCTION

Traditional vehicle security systems rely on mechanical locks, alarm systems, and remote key access. Mechanical locks, such as steering wheel locks and gear shift locks, provide a basic level of protection but are vulnerable to physical tampering. Alarm systems, on the other hand, use motion or impact sensors to trigger loud sirens when unauthorized access is detected. However, thieves have developed techniques to bypass these alarms, reducing their effectiveness. Remote keyless entry and ignition systems are commonly used in modern vehicles, but they are susceptible to relay attacks, where thieves intercept and clone key signals to gain unauthorized access.

To enhance security, GPS-based vehicle tracking systems have been introduced, allowing owners to monitor their vehicle's location in real-time. While effective in theft recovery, these systems do not prevent unauthorized access. Some advanced security solutions integrate RFID (Radio Frequency Identification) or keypad-based access control, but these methods can still be compromised through code interception or duplication. More recent innovations include smartphone-based authentication and face recognition, but these require internet connectivity and are often expensive to implement. Given these limitations, a biometric-based authentication system using fingerprint sensors, combined with real-time alerts and GPS tracking, provides a more secure and reliable solution for vehicle anti-theft systems.

3.2 DRAWBACKS

3.2.1. Vulnerability to Hacking and Signal Interception

Modern security systems, such as remote keyless entry and smart locking mechanisms, are vulnerable to hacking techniques like relay attacks. Thieves use signal amplifiers to intercept and relay key fob signals, allowing them to unlock and start vehicles without needing physical access to the key. This method is particularly effective against vehicles with passive keyless entry systems.

Furthermore, code-grabbing devices can capture the rolling codes used in remote keyless systems, enabling attackers to replay them and gain unauthorized access. With

advancements in hacking tools, criminals can easily clone RFID tags and key fob signals, bypassing traditional security mechanisms.

Even GPS-based security systems are not immune to hacking. GPS jamming and spoofing techniques can interfere with location tracking, preventing vehicle owners from detecting unauthorized movements. Cybercriminals can exploit software vulnerabilities in connected car systems, gaining control over vehicle functions remotely.

Additionally, Wi-Fi and Bluetooth-enabled vehicle security systems can be hacked using brute-force attacks or software exploits. Many vehicle owners neglect firmware updates, leaving their systems vulnerable to cyber threats. Without strong encryption and multi-layered authentication, these security measures can be compromised.

Due to these vulnerabilities, relying solely on electronic security mechanisms without additional layers of protection, such as biometric authentication, increases the risk of vehicle theft.

3.2.2. Limited Effectiveness of Alarm Systems

Traditional car alarms are designed to emit loud sirens when unauthorized access is detected. However, these alarms have become less effective because they often trigger false alerts, leading to alarm fatigue among vehicle owners and bystanders. People tend to ignore car alarms, assuming they are accidental activations rather than real theft attempts.

Additionally, experienced thieves can bypass alarm systems by quickly disabling them before they alert anyone. Many modern car thieves use signal jammers to prevent alarm activation, allowing them to enter the vehicle without triggering any noise.

Even when alarms are functional, they rely on human intervention. If the owner or security personnel do not respond immediately, thieves can still escape with the vehicle. Many break-ins occur in isolated or noisy areas, where an alarm may not attract enough attention to deter theft.

Some advanced alarm systems integrate with smartphone notifications, but these still require an internet connection. In areas with weak network coverage, owners may not receive alerts in time to take action. Without additional security layers, alarm systems alone cannot guarantee vehicle protection.

The primary drawback of alarm-based security is that it only reacts after a theft

attempt has begun, rather than preventing unauthorized access in the first place.

3.2.3. Inefficiency of GPS Tracking in Theft Prevention

GPS tracking is widely used for vehicle security, but it has several limitations. While it helps locate stolen vehicles, it does not prevent theft from occurring in the first place. By the time the owner realizes the vehicle is missing, thieves may have already taken measures to disable the tracker.

One major issue is GPS signal interference. Thieves use GPS jammers to block signals, preventing tracking devices from transmitting location data. Some advanced thieves even use metal enclosures or disconnect vehicle batteries to cut off power to tracking systems.

Another problem is delayed tracking response. Many GPS-based security systems update location data at specific intervals, meaning the real-time location may not always be accurate. If a thief moves the vehicle quickly, owners may receive an outdated location report, making recovery difficult.

Additionally, GPS tracking depends on internet connectivity to send alerts. In areas with weak network coverage or inside underground parking lots, GPS signals may be lost, preventing the owner from tracking the stolen vehicle.

While GPS tracking is useful for recovery, it is not a foolproof solution. A more effective security system should prevent theft from happening rather than relying solely on post-theft tracking.

3.2.4. Physical Locks and Mechanical Security Flaws

Mechanical security devices like steering wheel locks, gear shift locks, and tire clamps are commonly used to deter theft. However, they are not foolproof and can be bypassed with the right tools.

Steering wheel locks, for example, can be cut or broken using bolt cutters or saws. Some advanced thieves use liquid nitrogen to weaken metal locks, making them easier to break. Similarly, gear shift locks and tire clamps can be disabled by drilling, prying, or using impact tools.

Another issue is that many mechanical security devices have universal designs, meaning that skilled thieves can find ways to unlock them without much effort. Some

locks can even be picked with basic lock-picking tools, making them ineffective against experienced criminals.

Additionally, mechanical locks require manual operation, meaning the owner must remember to engage them every time they leave the vehicle. In cases where the owner forgets, the vehicle is left unprotected.

Since mechanical security solutions provide only a passive level of protection, they are not sufficient on their own. Biometric authentication, combined with electronic alerts and real-time monitoring, offers a more secure and modern alternative.

3.2.5. Dependency on User Awareness and Human Response

Many existing vehicle security systems rely on the owner's vigilance to be effective. For example, an alarm or notification is useless if the owner does not act immediately. If a theft occurs while the owner is asleep, busy, or in an area with poor network coverage, the response time may be too slow to prevent the theft.

Keyless entry and remote ignition systems require owners to safeguard their key fobs, but many people unknowingly expose themselves to relay attacks by leaving key fobs near doors or windows. Thieves can amplify key signals from outside the house and gain access without the owner realizing it.

Similarly, password-based security systems depend on strong, unique passwords, but many users choose weak or easily guessable passwords, making them vulnerable to hacking. Some vehicle owners even write down passwords or reuse common codes, increasing the risk of unauthorized access.

Another limitation is that many security measures require periodic updates and maintenance. Owners often forget to update firmware for their security systems, leaving them exposed to known vulnerabilities. Even GPS tracking systems require periodic battery checks and network configuration updates to function effectively.

Because existing security methods require active user engagement, they can fail due to human error or negligence. A more robust system should incorporate automated security measures, such as fingerprint authentication and instant alerts, to reduce dependence on user awareness.

CHAPTER 4
PROPOSED WORK

4.1. INTRODUCTION

The Vehicle Anti-Theft System Using Fingerprint Authentication is designed to provide multi-layered security against unauthorized access and theft. The system integrates biometric authentication, vibration detection, GPS tracking, GSM alerts, and real-time image capture for enhanced protection. The ESP32 microcontroller serves as the core processing unit, managing sensor inputs and communication. The fingerprint sensor is used as the primary authentication method, ensuring that only registered users can start the vehicle. In case of unauthorized access attempts, the system triggers multiple security measures, including buzzer alarms and theft alerts. The system also incorporates a vibration sensor, which detects unusual vehicle movement, such as forced entry or towing attempts, and responds with an alert.

If unauthorized access is detected, the system sends an instant GPS location via GSM to a registered mobile number. This enables real-time tracking of the vehicle, allowing owners or authorities to take immediate action. The integration of GSM communication ensures that alerts are received even in areas without internet connectivity. Additionally, a camera module, connected to a Python-based system on a PC, captures an image of the intruder when unauthorized fingerprint access is detected. The captured image is then emailed to the registered owner as further evidence of the attempted theft. This real-time monitoring and reporting system significantly improves theft detection and vehicle recovery chances.

Unlike traditional security measures that rely on alarms and GPS tracking alone, this system actively prevents unauthorized access using fingerprint authentication. Moreover, it provides instant alerts and visual evidence, which enhances the effectiveness of security responses. By combining biometric security, real-time notifications, and image-based intruder detection, this approach overcomes the limitations of existing methods and ensures a proactive, rather than reactive, vehicle security solution.

4.2. BLOCK DIAGRAM

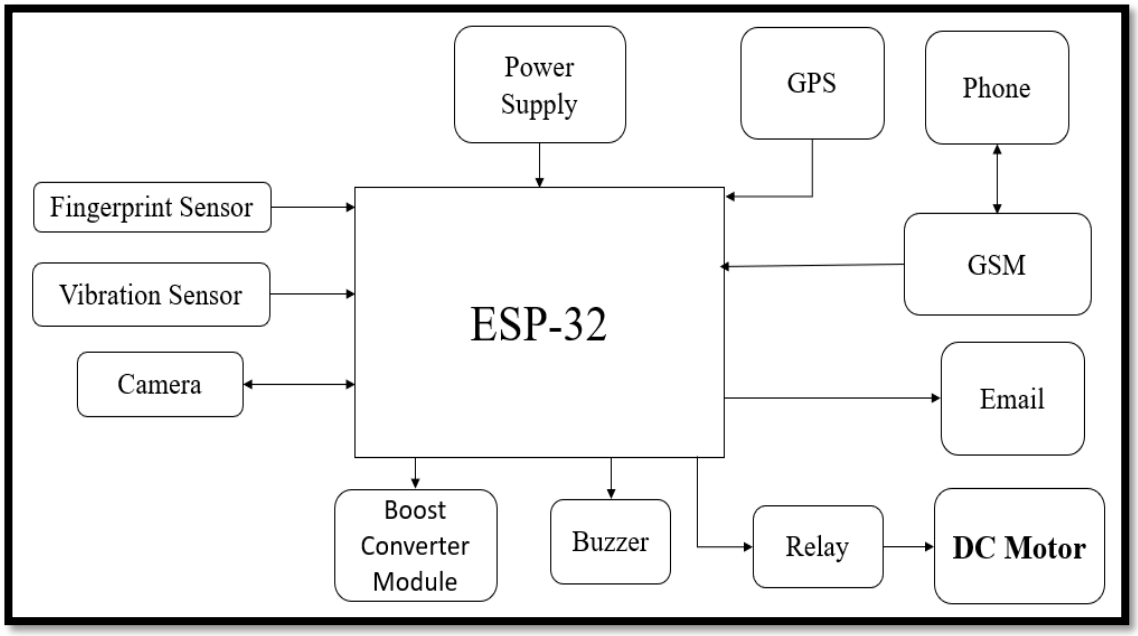

Fig.4.1: Block Diagram of the Proposed System

4.3. FLOWCHART

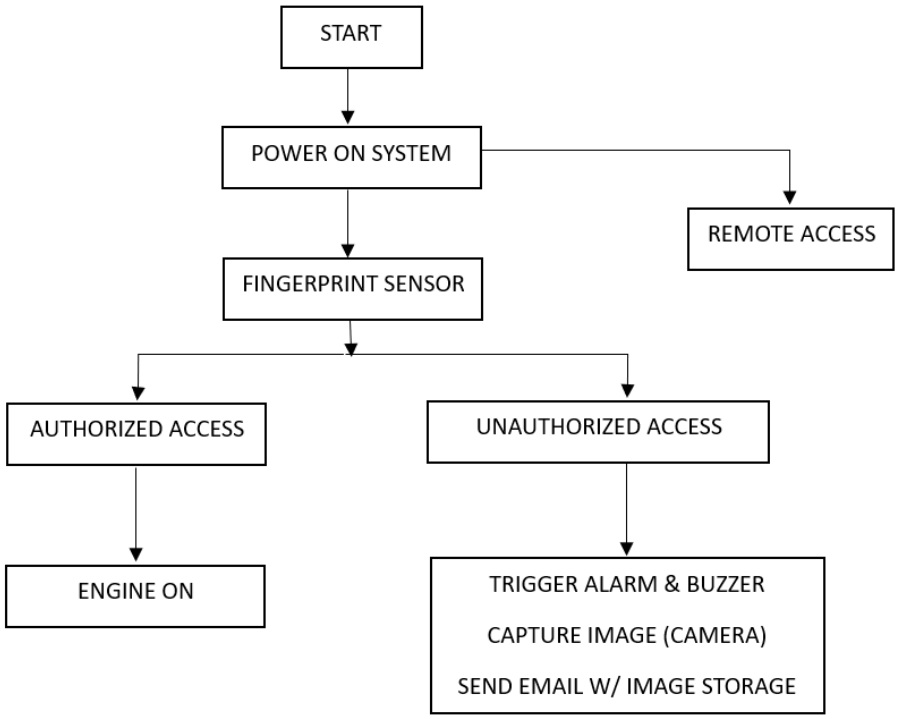

Fig. 4.2(a): Flow chart of Authorization using Finger Print

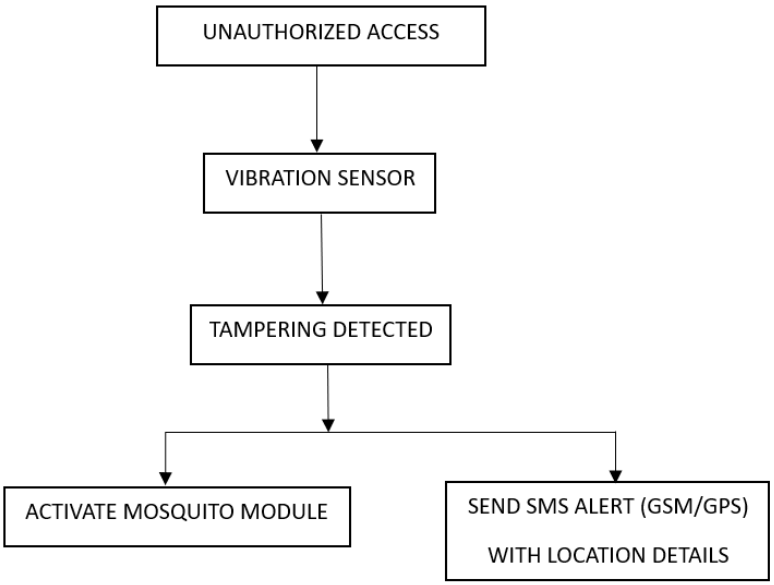

Fig. 4.2(b): Flow chart of Unauthorized Access

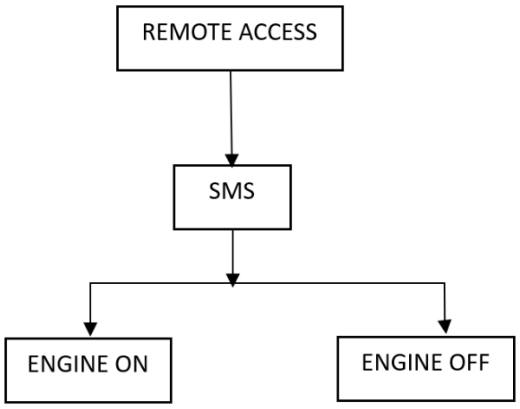

Fig. 4.2(c): Flow chart of Remote Access

Key Components in Flowchart:

1. Fingerprint Authentication: Allows or denies vehicle access.
2. Vibration Sensor: Detects unauthorized movements or tampering.
3. Camera Module: Captures intruder images and sends them via email.
4. Mosquito Module: Activates ultrasonic deterrent upon unauthorized access.
5. GSM/GPS Module: Sends SMS alerts with real-time vehicle location.
6. Remote Control: User can immobilize the vehicle via SMS commands.
7. Buzzer Alarm: Alerts nearby individuals of security breaches.

4.4. CIRCUIT DIAGRAM

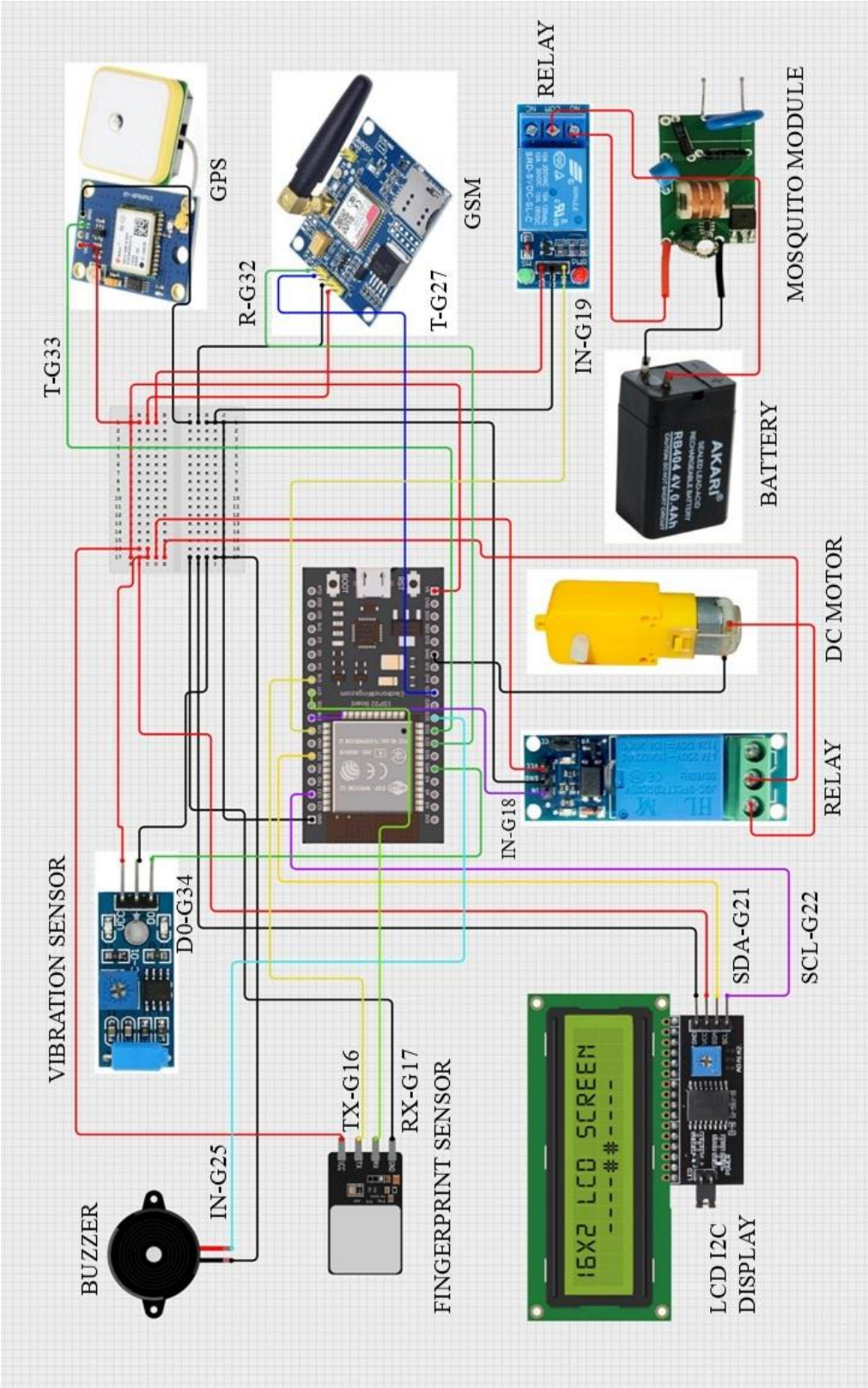

Fig.4.2(d): Circuit Diagram of Proposed System and Connections

4.5. ADVANTAGES

4.5.1. Enhanced Security Through Biometric Authentication

Fingerprint authentication ensures only authorized users can start the vehicle, eliminating risks associated with traditional key-based systems. Unlike passwords or RFID cards, fingerprints cannot be easily stolen, cloned, or lost, making them a highly secure authentication method. The use of a biometric sensor makes unauthorized access nearly impossible, significantly reducing theft incidents.

Moreover, fingerprint-based security is quick and convenient, allowing the driver to unlock and start the vehicle within seconds. Unlike password-based systems that require frequent changes, fingerprints remain unique and unchanged throughout life, making them a hassle-free authentication mechanism. This ensures that only registered owners and authorized users can access the vehicle, preventing unauthorized starts.

Traditional security systems lack real-time authentication, making them vulnerable to relay attacks where thieves use wireless devices to mimic a car key. In contrast, the fingerprint system ensures that the person physically present is the actual authorized driver. This eliminates the possibility of key duplication or hacking, making the vehicle significantly more secure.

4.5.2. Real-Time GPS Tracking and Theft Alerts

The system integrates GPS tracking and GSM communication, ensuring real-time monitoring of the vehicle's location. If a theft attempt is detected, the system automatically sends the GPS coordinates of the vehicle to the registered mobile number via SMS alerts, allowing the owner or authorities to track it instantly. This enhances vehicle recovery chances and prevents prolonged theft incidents.

Unlike conventional security measures that rely on after-theft investigation, this system provides proactive tracking, alerting the owner at the moment of unauthorized access or suspicious movement. This real-time functionality ensures that vehicle owners can take immediate action instead of discovering the theft hours later. The integration of GSM ensures alerts even in remote areas without internet connectivity.

Most traditional GPS trackers are separate modules that require manual activation or are hidden inside vehicles. However, this system automates theft detection and tracking by instantly transmitting the location as soon as unauthorized access is detected.

This minimizes response time and significantly increases the chances of vehicle retrieval.

4.5.3. Immediate Intruder Identification Through Image Capture

A major drawback of existing security systems is the lack of visual evidence of the intruder. This proposed system overcomes this issue by using a camera module connected to a Python-based system on a PC, which captures an image of the unauthorized person. This image is then automatically emailed to the vehicle owner, serving as crucial evidence for law enforcement.

The integration of image capture technology ensures that even if the thief manages to escape, clear facial recognition data is available for tracking them down. Unlike traditional CCTV systems that require manual monitoring, this system works autonomously, ensuring that images are taken at the exact moment of an attempted break-in. This increases the likelihood of identifying and apprehending the culprit.

Moreover, the use of email-based alerts ensures that the vehicle owner can access the evidence from anywhere in the world. Even if the mobile network is down, the captured image remains stored in the email, providing long-term proof of the attempted intrusion. This is a significant advantage over traditional alarm systems, which only notify the user without providing any identification of the perpetrator.

4.5.4. Multi-Layered Security for Theft Prevention

Unlike conventional vehicle security systems that rely on a single layer of protection, this system incorporates multiple layers of security including biometric authentication, vibration sensors, real-time alerts, GPS tracking, and intruder image capture. This makes it highly resistant to theft attempts, as bypassing one security measure will immediately trigger another.

For instance, even if a thief manages to physically break into the vehicle, they cannot start the engine without fingerprint authentication. If they attempt forceful access, the vibration sensor detects movement and triggers an alarm, sending real-time alerts to the owner’s mobile phone. If they continue with their attempt, the camera module captures their image, providing solid evidence for further investigation.

This combination of biometric security, real-time alerts, and physical deterrence (buzzer alarms) ensures that any unauthorized access attempt is instantly detected and countered. Unlike traditional systems that rely solely on alarms or GPS tracking, this

proactive approach actively prevents theft before it happens.

4.5.5. Smart Alert System and Emergency Response

The proposed system integrates a smart alert mechanism that ensures instant notification in case of unauthorized access, vibration detection, or theft attempts. Traditional security systems often have delayed responses, making it difficult for owners to react quickly. In contrast, this system ensures that alerts are sent within seconds, allowing immediate action.

Additionally, the system includes a buzzer alarm, which acts as an immediate deterrent. If unauthorized access or suspicious vibration is detected, the loud buzzer alerts nearby individuals, discouraging thieves from continuing their attempt. Unlike silent tracking systems, this actively prevents theft rather than just monitoring it.

The GSM module ensures that alerts are received even in the absence of WiFi, making it highly reliable in all environments. Whether the vehicle is parked in a remote area or underground, the system ensures that the owner is always informed of potential security threats. This round-the-clock monitoring significantly reduces the risk of vehicle theft

4.6. APPLICATIONS

4.6.1 Personal Vehicle Security

One of the primary applications of this system is enhancing the security of personal vehicles. Many car owners rely on traditional key-based locking mechanisms, which can be easily bypassed by thieves using duplication techniques or relay attacks. Fingerprint authentication ensures that only the registered owner can unlock and start the vehicle, making theft nearly impossible.

Additionally, in case of an attempted unauthorized entry, the system activates a buzzer alarm, alerting the owner and people nearby. This immediate sound deterrent prevents thieves from proceeding with their attempts. Furthermore, the system sends a real-time location update via GPS and GSM to the registered mobile number, allowing the owner to take swift action.

Another advantage is convenience, as owners do not need to carry physical keys or remember passwords. The biometric system allows quick authentication, reducing the risk of key misplacement or password breaches. Unlike traditional locks, which can be

picked, the fingerprint sensor offers a highly secure and user-friendly alternative for everyday vehicle use.

4.6.2 Commercial Fleet Security

This system is highly beneficial for commercial fleets, where multiple vehicles need to be secured against theft and unauthorized usage. Fleet operators often struggle with managing keys for numerous drivers, leading to security risks and operational inefficiencies. By implementing fingerprint authentication, only authorized drivers can access the vehicles, eliminating unauthorized use.

Additionally, the integration of GPS tracking ensures that fleet managers can monitor their vehicles in real-time, reducing chances of misuse. If an unauthorized individual attempts to start the vehicle, the system immediately notifies the fleet manager via SMS and captures an image of the person for further verification. This enhances accountability among drivers and prevents internal theft.

Furthermore, in case of a stolen fleet vehicle, the system's location tracking and image capturing capabilities assist law enforcement in quick recovery. Unlike conventional fleet management systems, which only track location, this multi-layered security approach ensures both prevention and recovery of stolen assets.

4.6.3 Public Transport Vehicles

Public transport systems, such as buses, taxis, and rental cars, are often vulnerable to unauthorized access and misuse. Drivers may leave keys unattended, leading to potential security risks. This fingerprint-based authentication system ensures that only registered transport employees can start the vehicle, eliminating unauthorized access.

Moreover, public transport providers can remotely monitor vehicle movement using the integrated GPS module. This ensures that buses and taxis follow their designated routes without deviation. If a vehicle is hijacked or stolen, its location can be immediately tracked and recovered, reducing financial losses for transport companies.

The addition of image capture upon unauthorized access further enhances security by identifying potential intruders. If an unauthorized person attempts to access the vehicle, an automatic email alert with the intruder’s image is sent to the transport company's security team, ensuring quick response and necessary action.

4.6.4 High-End and Luxury Car Protection

Luxury cars are often targeted by thieves due to their high market value and advanced technology. Traditional security systems in luxury vehicles rely on keyless entry or RFID-based access, which can be hacked using relay attacks. Implementing biometric authentication eliminates this vulnerability by ensuring physical presence verification of the owner.

Moreover, in case of an attempted break-in, the system's buzzer and real-time GPS tracking notify the owner instantly. This is crucial for luxury car owners who frequently park their vehicles in high-risk locations. The immediate alerts allow them to respond quickly, either by informing authorities or remotely disabling the vehicle.

Additionally, high-end car owners often require valet services, where strangers access their vehicles. This system allows them to register temporary access fingerprints, which expire after a certain time, ensuring that valets or service personnel cannot retain permanent access to the vehicle.

4.6.5 Rental Car Security

Car rental services often face issues related to vehicle theft, unauthorized access, and tracking difficulties. By integrating fingerprint authentication, only registered renters can start the car, ensuring that stolen or duplicated rental keys cannot be misused.

Additionally, rental companies can use the GPS tracking feature to monitor vehicle usage and prevent clients from violating rental agreements, such as driving outside designated areas. If a renter fails to return the vehicle, its real-time location can be tracked, ensuring easy recovery.

The image capture feature is particularly useful in identifying fraudulent users. If an unauthorized person attempts to start the car, the system automatically takes a picture and sends it to the rental company's email. This helps in legal documentation and prevents identity fraud.

4.6.6 Emergency Vehicle Security

Ambulances, fire trucks, and police cars require strict security measures to prevent unauthorized access. This system ensures that only verified personnel can start these emergency vehicles, reducing the chances of misuse or vehicle theft.

In the case of emergency response vehicles, quick authentication is crucial. Fingerprint-based access allows instant vehicle activation without the need for physical keys, ensuring a faster response time during critical situations. This can be life-saving in emergencies where every second counts.

Additionally, if an emergency vehicle is stolen or hijacked, the GPS tracking system ensures real-time monitoring, allowing authorities to locate and recover it quickly. The buzzer alert system can also be used to notify nearby security personnel in case of suspicious activity.

4.6.7 Military and Defence Vehicle Security

Military and defence vehicles require high-level security to prevent unauthorized access and potential national security threats. Traditional key-based security systems are vulnerable to theft, duplication, and unauthorized use. The fingerprint authentication system ensures that only authorized personnel can access these critical vehicles.

Additionally, the GPS tracking feature allows military officials to monitor vehicle movements, preventing unauthorized usage or deviations from designated routes. If a military vehicle is stolen or hijacked, the system provides real-time tracking and alerts, ensuring a quick response to recover the vehicle.

Moreover, the system's intruder identification feature enhances security by capturing images of unauthorized personnel attempting to access military vehicles. This aids in investigations and reinforces national security by preventing unauthorized access to defence assets.

4.6.8 Corporate and Official Vehicles

Many corporate organizations provide company-owned vehicles to their employees for official use. Ensuring that these vehicles are used only for business purposes is a major concern for fleet managers. The fingerprint authentication system restricts access to only designated employees, preventing unauthorized use.

Moreover, the GPS tracking module allows companies to monitor vehicle usage in real-time, ensuring that employees do not use company cars for personal travel. This prevents fuel misuse and ensures compliance with company policies.

Additionally, in case of a stolen company vehicle, the integrated system immediately alerts the corporate security team, providing real-time location updates and intruder identification images, allowing swift recovery of the vehicle.

4.6.9 Government Vehicles and Law Enforcement

Government vehicles, especially those used by law enforcement, must be highly secure to prevent theft and unauthorized use. The fingerprint authentication system ensures that only authorized government officials can operate these vehicles, preventing misuse.

Additionally, the system's real-time tracking capabilities help law enforcement monitor patrol vehicles and ensure that they remain within designated areas. In the case of a stolen police vehicle, the instant GPS location updates enable quick tracking and recovery.

The image capture feature provides crucial evidence in case of unauthorized access. If an unknown person attempts to enter a government vehicle, their image is automatically sent to the police department's security system, aiding in investigation and legal actions.

4.6.10 Smart Cities and IoT-Enabled Transport

As cities move toward smart transportation systems, integrating biometric authentication in public and private vehicles enhances overall security. This system can be linked to IoT networks, ensuring seamless monitoring and tracking of all vehicles in a smart city infrastructure.

The combination of fingerprint authentication, GPS tracking, and real-time alerts ensures that smart city transport systems remain secure, efficient, and theft-proof. Public transport authorities can monitor vehicle movement, restrict unauthorized access, and enhance commuter safety.

Moreover, the automated image capture and email alert system can be integrated with citywide surveillance, helping law enforcement identify potential criminals and prevent vehicle-related crimes in urban areas.

4.7 HARDWARE DESCRIPTION

4.7.1. ESP32:

The ESP32 microcontroller by Espressif Systems is a powerful, flexible, and cost-effective solution for IoT and embedded systems. Designed to cater to a wide range of applications, it combines dual-core processing power, wireless connectivity, and extensive peripheral support, making it an industry favorite. With its small size and energy efficiency, ESP32 is an ideal choice for low-power and battery-operated devices.

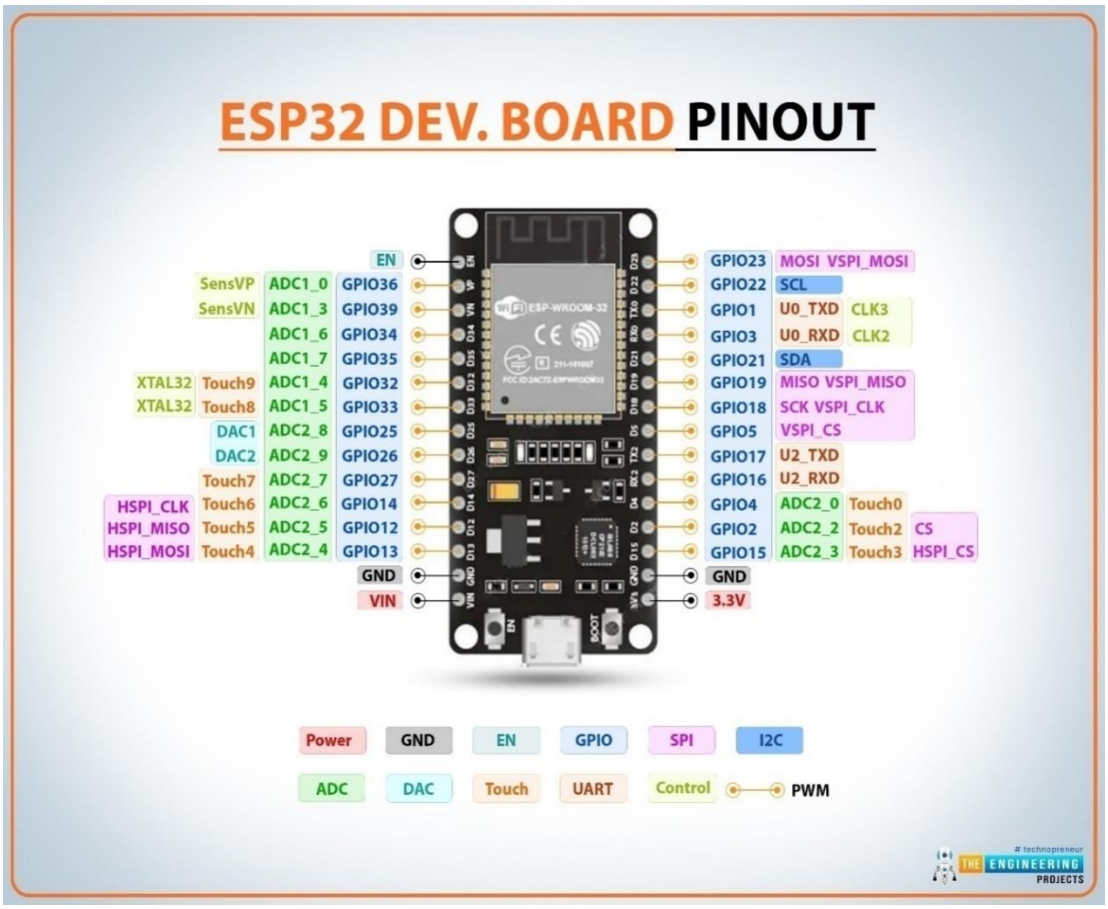

Fig. 4.3(a): ESP Microcontroller

Processor and Performance

At its core, the ESP32 is powered by a dual-core Tensilica Xtensa LX6 processor, clocked up to 240 MHz. This architecture ensures high performance for computational tasks while maintaining low power consumption. The dual-core setup allows one core to handle system tasks while the other manages user applications, ensuring smoother performance for multitasking scenarios.

Memory Capabilities

The ESP32 offers 520 KB of SRAM for instructions and data. In addition, certain variants like the ESP32-WROVER come with 8 MB of external PSRAM, which extends its capability to handle larger and more complex applications, such as machine learning models or image processing. Its flash memory typically ranges from 4 MB to 16 MB, enabling storage of extensive firmware and application data.

Wireless Connectivity

The standout feature of ESP32 is its wireless communication support, including both Wi-Fi and Bluetooth. Its Wi-Fi module complies with IEEE 802.11 b/g/n standards, allowing it to connect seamlessly to 2.4 GHz networks. In addition, the ESP32 supports both Bluetooth Classic and Bluetooth Low Energy (BLE), making it versatile for diverse communication protocols, including IoT devices, wearables, and home automation systems.

Pinout and GPIO Functionality

ESP32 features 36 GPIO pins, many of which are multiplexed to support various functions such as PWM, ADC, DAC, UART, I2C, and SPI. It includes 18 ADC channels, enabling high-resolution analog input, and 2 DAC channels for analog output. Special pins like GPIO 0 and EN (Enable) are used for boot and reset functionality. Its GPIO pins operate at 3.3V logic levels, which ensures compatibility with most modern sensors and modules.

Power Supply and Grounding

The ESP32 is designed to operate at 3.3V, with a typical power input provided via its VIN pin (5V) or micro-USB port. This is regulated down to the operating voltage internally. Multiple GND (Ground) pins are provided for stable grounding in circuit designs. Its low-power modes, such as deep sleep and hibernation, make it suitable for energy-efficient applications, extending the operational life of battery-powered systems.

Peripherals and Interfaces

The ESP32 supports various peripherals, including UART, SPI, I2C, I2S, and PWM, which make it suitable for interfacing with sensors, actuators, and other modules.

These interfaces are widely used in IoT applications, allowing easy integration of components like temperature sensors, displays, and motor drivers. The support for advanced communication protocols simplifies complex IoT system designs.

Board Variants and Customization

ESP32 is available in several variants, such as ESP32-DevKitC, ESP32-WROOM, and ESP32-WROVER, catering to different requirements. For example, the WROVER model is equipped with external PSRAM, which is suitable for high-memory applications. These boards often include pin headers, onboard LEDs, and USB-to-serial converters for easy development and debugging.

Programming and Development

Programming the ESP32 is straightforward, especially with support from the Arduino IDE. By adding the ESP32 board manager URL in Arduino's preferences, developers can select and program ESP32 boards directly. The board is also compatible with platforms like PlatformIO and Espressif's own ESP-IDF (IoT Development Framework), giving developers flexibility in their tools and frameworks.

Applications of ESP32

The versatility of ESP32 lends itself to a wide array of applications, including IoT devices, smart home systems, industrial automation, and robotics. It's also used in wearables and health monitoring systems due to its compact size and power efficiency. With its rich set of features and ease of integration, ESP32 continues to be a cornerstone of modern embedded systems.

Key Features of ESP32

Processor: Dual-core Tensilica Xtensa LX6, clocked up to 240 MHz.

Wireless Connectivity:

Wi-Fi: IEEE 802.11 b/g/n (2.4 GHz).

Bluetooth: BLE and Classic Bluetooth.

Memory:

Flash Memory: Up to 16 MB (typically 4 MB for most boards).

SRAM: 520 KB on-chip SRAM and an additional 8 MB of external PSRAM (on some modules).

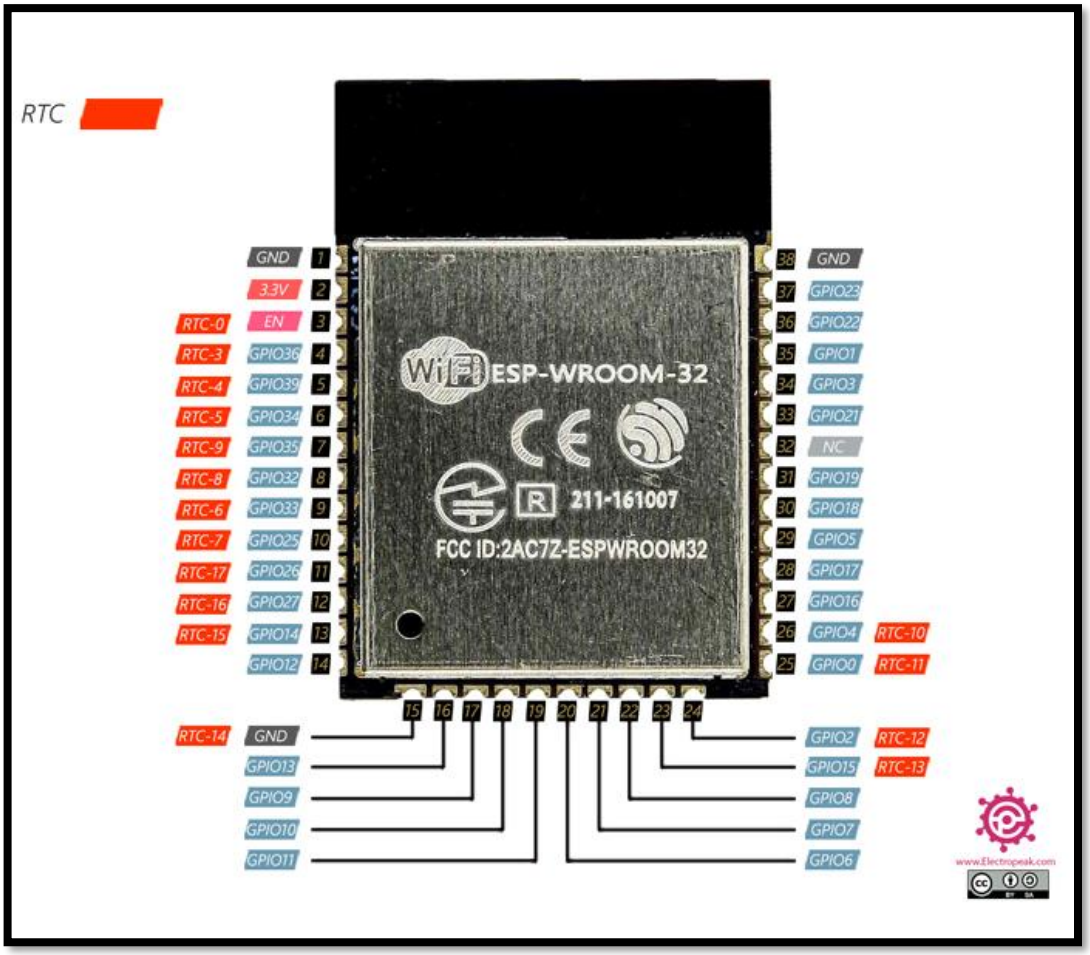

Fig. 4.3(b): WiFi ESP-WROOM-32 Processor

Power Supply:

Operates on 3.3V.

Integrated low-power management for deep sleep modes.

GPIO Pins: 36 GPIO pins that can be configured for various input/output operations.

ADC: 18 channels with a resolution of 12 bits.

DAC: 2 DAC channels for analog output.

I2C, SPI, UART, PWM, I2S Support: Extensive peripherals for communication and control.

Operating Voltage: 3.3V logic level for I/O pins.

4.7.2. LCD:

LCD (Liquid Crystal Display) is the innovation utilized in scratch pad shows and other littler PCs. Like innovation for light-producing diode (LED) and gas-plasma, LCDs permit presentations to be a lot more slender than innovation for cathode beam tube (CRT). LCDs expend considerably less power than LED shows and gas shows since they work as opposed to emanating it on the guideline of blocking light.

A LCD is either made with a uninvolved lattice or a showcase network for dynamic framework show. Likewise alluded to as a meager film transistor (TFT) show is the dynamic framework LCD. The uninvolved LCD lattice has a matrix of conductors at every crossing point of the network with pixels. Two conductors on the lattice send a current to control the light for any pixel. A functioning framework has a transistor situated at every pixel crossing point, requiring less current to control the luminance of a pixel.

Some aloof network LCD's have double filtering, which implies they examine the matrix twice with current in the meantime as the first innovation took one sweep. Dynamic lattice, be that as it may, is as yet a higher innovation.

A 16x2 LCD show is an essential module that is generally utilized in various gadgets and circuits. These modules more than seven sections and other multi fragment LEDs are liked. The reasons being: LCDs are affordable; effectively programmable; have no restriction of showing exceptional and even custom characters (not at all like in seven fragments), movements, etc.

A 16x2 LCD implies 16 characters can be shown per line and 2 such lines exist. Each character is shown in a lattice of 5x7 pixels in this LCD. There are two registers in this LCD, in particular Command and Data.

The directions given to the LCD are put away by the order register. An order is a direction given to LCD to play out a predefined assignment, for example, introducing it, clearing its screen, setting the situation of the cursor, controlling presentation, and so forth. The information register will store the information that will be shown on the LCD. The information is the character's ASCII incentive to show on the LCD.

Data/Signals/Execution of LCD:

Now that was all about the signals and the hardware. Let us come to data, signals and execution.

Two types of signals are accepted by LCD, one is data and one is control. The LCD module recognizes these signals from the RS pin status. By pulling the R / W pin high, data can now also be read from the LCD display. Once the E pin has been pulsed, the LCD display reads and executes data at the falling edge of the pulse, the same for the transmission case.

It takes 39-43µS for the LCD display to place a character or execute a command. It takes 1.53ms to 1.64ms except for clearing display and searching for cursor to the home position.

Any attempt to send data before this interval may result in failure in some devices to read data or execute the current data. Some devices compensate for the speed by storing some temporary registers with incoming data.

There are two RAMs for LCD displays, namely DDRAM and CGRAM. DDRAM registers the position in which the character would be displayed in the ASCII chart. Each DDRAM byte represents every single position on the display of the LCD. The DDRAM information is read by the LCD controller and displayed on the LCD screen. CGRAM enables users to define their personalized characters. Address space is reserved for users for the first 16 ASCII characters.

Users can easily display their custom characters on the LCD screen after CGRAM has been set up to display characters.

Images of LCD Display:

Fig. 4.4(a): LCD – Front View

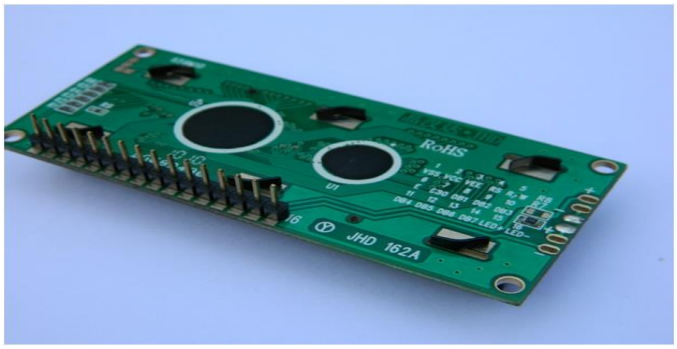

Fig. 4.4(b): LCD – Back View

Pin Diagram:

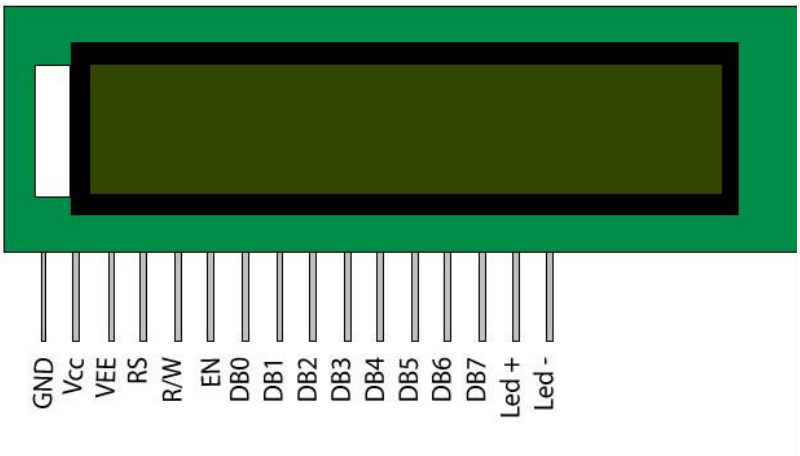

Fig. 4.4(c): LCD Pin Diagram

Table. 4.1(a): Pin Description of LCD

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V_{EE}
4	Selects command register when low; and data register when high	Register Select

5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V_{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

RS (Register select)

A 16X2 LCD has two order and information registers. The determination of the register is utilized to change starting with one register then onto the next. RS=0 for the register of directions, while RS=1 for the register of information.

Command Register

The guidelines given to the LCD are put away by the direction register. An order is a direction given to LCD to play out a predefined assignment, for example, instating it, clearing its screen, setting the situation of the cursor, controlling showcase, and so on. Order preparing happens in the direction register.

Data Register:

The information register will store the information that will be shown on the LCD. The information is the character's ASCII incentive to show on the LCD. It goes to the information register and is prepared there when we send information to the LCD. While choosing RS=1, the information register.

Read and Write Mode of LCD:

As stated, the LCD itself comprises of an interface IC. This interface IC can be

perused or composed by the MCU. A large portion of the occasions we're simply going to keep in touch with the IC since perusing will make it increasingly perplexing and situations like that are exceptionally uncommon.Information such as cursor position, status completion interrupts, etc. can be read if necessary.

LCD Commands:

There are some preset commands in the LCD that we need to send to the LCD via some microcontroller. The following are some important command instructions:

Table. 4.1(b): Command codes for LCD

Sr.No.	Hex Code	Command to LCD instruction Register
1	01	Clear display screen
2	02	Return home
3	04	Decrement cursor (shift cursor to left)
4	06	Increment cursor (shift cursor to right)
5	05	Shift display right
6	07	Shift display left
7	08	Display off, cursor off
8	0A	Display off, cursor on
9	0C	Display on, cursor off
10	0E	Display on, cursor blinking
11	0F	Display on, cursor blinking
12	10	Shift cursor position to left
13	14	Shift cursor position to right
14	18	Shift the entire display to the left

15	1C	Shift the entire display to the right
16	80	Force cursor to beginning (1st line)
17	C0	Force cursor to beginning (2nd line)
18	38	2 lines and 5×7 matrix

Block Diagram of LCD Display:

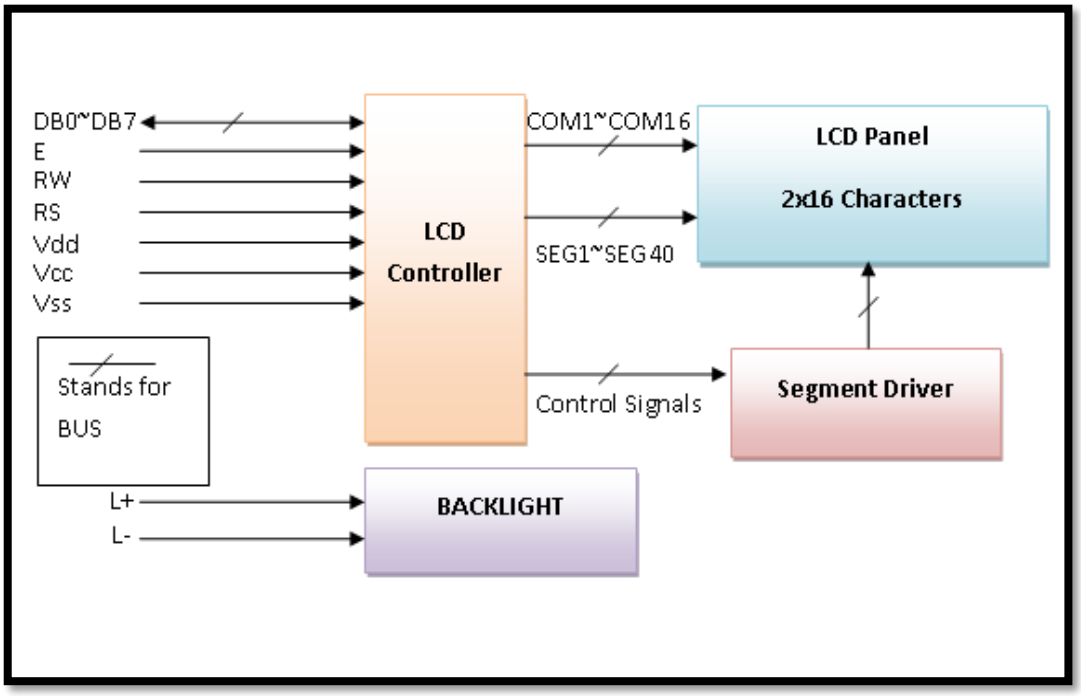

Fig. 4.4(d): Block Diagram of LCD

Table 4.1(c): Control and display commands of LCD

Instruction	Instruction Code										Instruction Code Description	Execution time
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Read Data From RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data from internal RAM	1.53-1.64ms
Write data to RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data into internal RAM (DDRAM/CGRAM)	1.53-1.64ms

Busy flag & Address	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Busy flag (BF: 1→ LCD Busy) and contents of address counter in bits AC6-AC0.	39 µs
Set DDRAM Address	0	0	1	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Set DDRAM address in address counter.	39 µs
Set CGRAM Address	0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0	Set CGRAM Address in address counter.	39 µs
Function Set	0	0	0	0	1	DL	N	F	X	X	Set interface data length (DL: 4bit/8bit), Numbers of display line (N: 1-line/2-line) display font type (F:0→ 5×8 dots, F:1→ 5×11 dots)	39 µs
Cursor or Display Shift	0	0	0	0	0	1	S/C	R/L	X	X	Set cursor moving and display shift control bit, and the direction without changing DDRAM data	39 µs
Display & Cursor On/Off	0	0	0	0	0	0	1	D	C	B	Set Display(D),Cursor(C) and cursor blink(b) on/off control	39 µs
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	SH	Assign cursor moving direction and enable shift	0µs

											entire display.	
Return Home	0	0	0	0	0	0	0	0	1	X	Set DDRAM Address to “00H” from AC and return cursor to its original position if shifted.	43µs
Clear Display	0	0	0	0	0	0	0	0	0	1	Write “20H” to DDRAM and set DDRAM Address to “00H” from AC	43µs

4-bit and 8-bit Mode of LCD:

The LCD can work in two striking modes, the 4-bit mode and the 8-bit mode. We send the information snack through snack in 4 bit mode, first upper chomp, by then lower snack. For those of you who don't have the foggiest idea what a goody is: a chomp is a four-piece gathering, so a byte's lower four bits (D0-D3) are the lower snack, while a byte's upper four bits (D4-D7) are the higher snack.This enables us to send 8 bit data.This connects with us to send 8 bit data.Whereas in 8 bit mode we can send the 8-bit information truly in one stroke since we utilize all the 8 information lines.You need to get it now; yes 8-bit mode is quicker and immaculate than 4-bit mode.In any case, the fundamental shortcoming is that it needs 8 microcontroller-related information lines. This will result in our MCU coming up short on I/O pins, so 4-bit mode is extensively utilized. To set these modes, no control pins are used.

4.7.3. FINGER PRINT MODULE

General view

Fingerprint enrollment, image process, characters acquisition, fingerprint template creation, fingerprint template storage, fingerprint compare (1: 1, 1: N), fingerprint delete. This module can work with different devices based on UAWRT such as PC, SCM and so on. Only easy circuits and fingerprint module can enhance your product into fingerprint authentication power. It is widely used by electronics business,

information security, access control, identity authentication and other security industry.

Fig. 4.5(a): Fingerprint Module

Application solution

1 When FPRS is embedded into your system, the other functions will be controlled by MCU Controller, so developer can realize his own function logic, user interface and communication port through hard ware and software development, such as fingerprint time and attendance and so on.

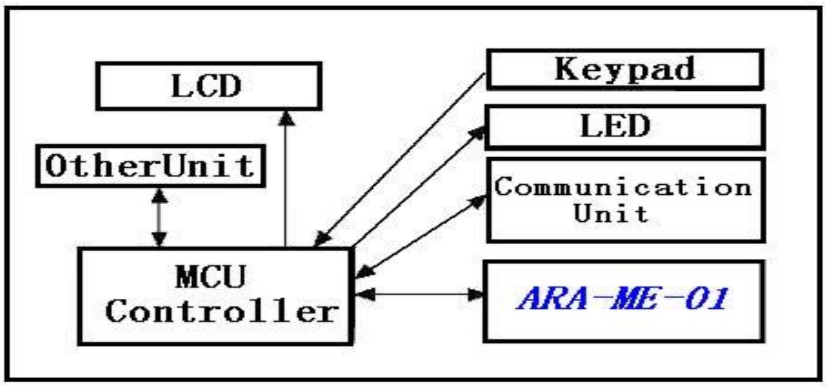

Fig.4.5(b): Architecture Of Fingerprint Module

2 Expand the port such as RS485, Wiegand, even GPRS wireless communication with central controller outside by MCU in the central controller system.

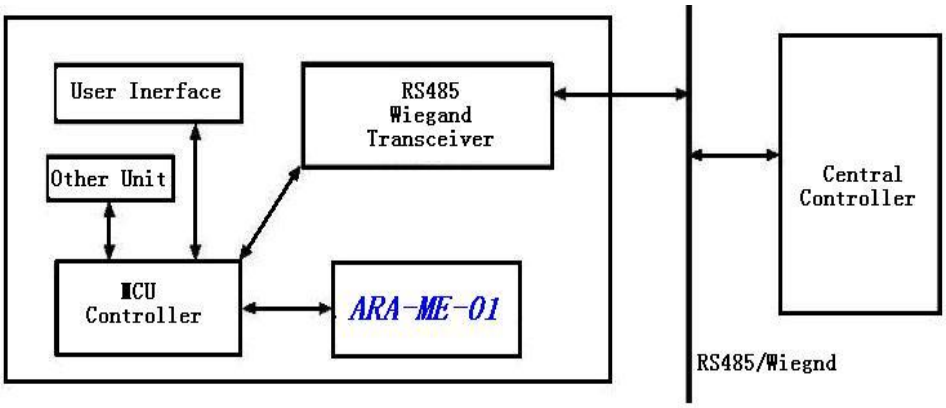

Fig.4.5(c): Communication Architecture of Fingerprint Authentication Unit

Table. 4.2(a): Pin Description of Fingerprint Module

Pin	Name	Spec
1	GND	Grand
2	Reset	Module reset control
3	TXD	Serial sender
4	RXD	Serial receiver
5	PowerEN	Power control
6	VCC	5V

Remark：

1 reset signal is the low level, the level width is bigger than 10ms.

2 high level turn on the power, low level turn off the power.

Software port spec

1. Communication port UART (Universal Asynchronous Serial Port), 9600bps to 115200bps (option), start with 1 bit, stop with 1 bit, no check bit.

2. Communication protocol Module stays in slave mode, and host can direct the module work by different command. All the command of the host and response of the module and data transfer are in standard data pack. Host must pack and analyze the command and the data in standard format.

Table. 4.2(b): Fingerprint authentication performance

Item	spec
Sensor	AES2510
Image	256*288
Resolution	500DPI
Register time	<3秒
Math time （1：1）	<0.1秒
Math time （1：N）	<0.5秒
FRR	<0.1%
FAR	<0.001%
Fingerprints capacity	160 fingerprints

Table. 4.2(c): Other Specifications of Fingerprint Module

item	Symbol	min	classic	max	Measure
Work Voltage	Vn	4.5	5	5.5	V
Work Current	In	50	60	80	mA
Sleep current	Is	9	10	12	uA
Work Temperature	Tn	— 20		70	℃
Humility	Hn	30		120	℉
Module size	Ln	40mm*25mm20mm*25mm			mm

Table.4.2(d): Interface of Fingerprint Module

Type	Description
UART	3.3V CMOS level Baud rates up to 921.6kbps (factory default is 115.2kbps) RS232/422/485 supported via additional level converter
Digital I/O	3.3V CMOS level 8 ports separately configurable 26bit Wiegand I/O supported via additional level converter

Table. 4.2(e): Connector Specifications of Fingerprint Module

Connector	Usage
J1	Host interface port I
J2	Host interface port II (Molex 53261-8090 compatible)
J3	Debug port for factory use only
J4	Sensor interface port, 20 pin FPC/FFC

Command summary

Table.4.2 (f): Flash storage structure

Address	Spec	Size
0x0000-0x7fff	Some codes	32k
0x8000-0x8fff	System parameters storage	4k
0x9000-0xffff	Fingerprint database	28k- the end

Table.4.2(g): System parameters storage area

Flash physical address	Page	Spec
0x8000	0	Reserve
0x8200	1	Parameter list
0x8400	2	User notebook
0x8600	3	Reserve
0x8800	4	Reserve
0x8a00	5	Reserve
0x8c00	6	Reserve
0x8e00	7	Reserve

Table.4.2(h): Command Specifications of Fingerprint module

PS_GetImage	Code: 01H	Function: read image from the sensor and store it in the image buffer.
PS_GenChar	Code: 02H	Function: create fingerprint features document through the image and store in CharBuffer1 or CharBuffer2.
PS_Match	Code: 03H	Function: compare the features documents in the CharBuffer1 and Charbuffer2.
PS_Search	Code: 04H	Function: search the fingerprint database according to the features document in the CharBuffer1 or CharBuffer2.
PS_RegModel	Code: 05H	Function: create fingerprint template according to the features documents in the CharBuffer1 and CharBuffer2.
PS_StoreChar	Code: 06H	Function: store the template into the flash fingerprint database from the features buffer.

PS_LoadChar	Code : 07H	Function: read template to features buffer from flash fingerprint database.
PS_UpChar	Code: 08H	Function: transfer the data in the features buffer to the PC.
PS_DownChar	Code: 09H	Function: load a template from PC to features buffer.
PS_DeletChar	Code: 0CH	Function: delete a template in the flash fingerprint database.
PS_Empty	Code: 0DH	Function: clear flash fingerprint database
PS_WriteReg	Code: 0EH	Function: write SSR
PS_ReadSysPara	Code: 0FH	Function: read basic system parameters
PS_SetPwd	Code: 12H	Function: set device password.
PS_VfyPwd	Code: 13H	Function: verify device password.
PS_GetRandomCode	Code: 14H	Function: get random code
PS_SetChipAddr	Code: 15H	Function: set the chip address
PS_WriteNotepad	Code: 18H	Function: write notepad
PS_ReadNotepad	Code: 19H	Function: read notepad
PS_ValidTe	Code: 1Dh	Function: read the template number from the flash

mpleteNum		

Pack structure

Send command

The packet flag=01 command packet

The packet flag=02 data packet, and data packet followed

The packet flag=08 the last data packet

All the data packet's head is 0xEF01,0xFFFFFFFF

Table.4.2(i): 01 command packet format:

Name	Packet head	Device address	Packet Flag	Packet Length	Command	Parameter 1	….	Parameter N	Checksum
Size	2bytes	4bytes	1 byte	2bytes	1byte	……			2 bytes
spec	0xEF01	xxxx	01						

Table.4.2(j): 02 data packet format:

Name	Packet head	Device address	Packet Flag	Packet Length	Command	Parameter 1	….	Parameter N	Checksum
Size	2bytes	4bytes	1 byte	2bytes	N bytes ……				2 bytes
spec	0xEF01	xxxx	02						

Table.4.2(k): 08 last data packet:

Name	Packet head	Device address	Packet Flag	Packet Length	Command	Parameter 1	….	Parameter N	Checksum
Size	2bytes	4bytes	1 byte	2bytes	N bytes ……				2 bytes
spec	0xEF01	xxxx	08						

·The data packet can't run itself; it must follow the command packet or the response packet.

·Download or upload the data packet in the same format.

·Packet length = the total bytes quantity of from the packet length to the checksum, including checksum, but not including the bytes quantity of the packet length itself.

·Checksum is the summer of all the bytes quantity from the packet flag to checksum, including packet flag, neglect the rounding over 2bytes.

Command response

Response packet/data packet:

The Packet flag = 07 response packet

The packet flag = 02 data packet, and packet followed

The packet flag = 08 the last data packet

Response is to send report to host about the result and how the commands run; data packet can follow the response packet. The host can confirm situation about the command execution and receiving response packet.

Table 4.2(l): Response packet format:

Name	Packet head	Device address	Packet Flag	Packet Length	Confirmation code	Parameter	Checksum
Size	2bytes	4bytes	1 byte	2bytes	1 bytes	N bytes	2 bytes
spec	0xEF01	xxxx	07				

Physical Dimensions

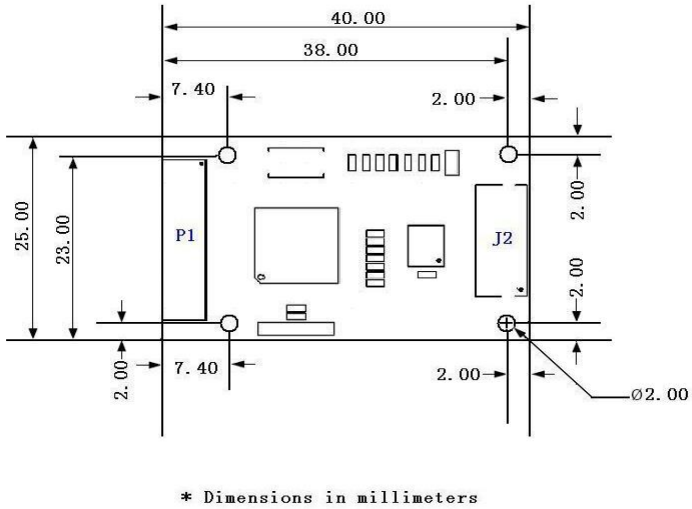

Fig. 4.5(d): Main board of Finger Print Sensor

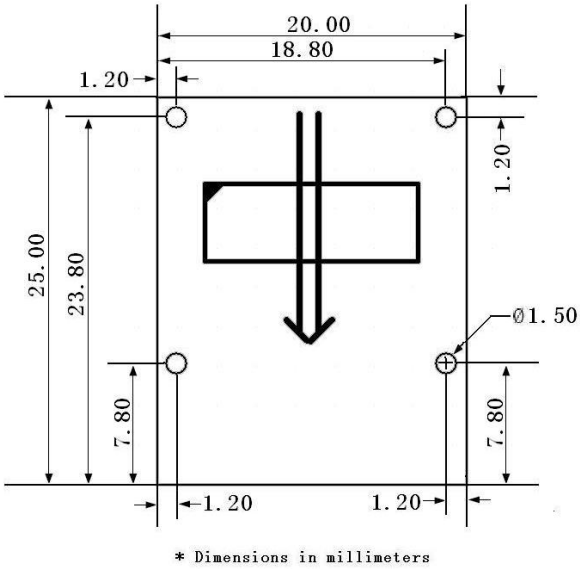

Fig. 4.5(e): Sensor

4.7.4. GPS:

Global Positioning System (GPS) is a satellite-based system that uses satellites and ground stations to measure and compute its position on Earth.

GPS is also known as Navigation System with Time and Ranging (NAVSTAR) GPS.

GPS receiver needs to receive data from at least 4 satellites for accuracy purpose. GPS receiver does not transmit any information to the satellites.

This GPS receiver is used in many applications like smartphones, Cabs, Fleet management etc.

Fig.4.6(a): GPS MODULE

How GPS Works

GPS receiver uses a constellation of satellites and ground stations to calculate accurate location wherever it is located.

These GPS satellites transmit information signal over radio frequency (1.1 to 1.5 GHz) to the receiver. With the help of this received information, a ground station or GPS module can compute its position and time.

How GPS Receiver Calculates its Position and Time

GPS receiver receives information signals from GPS satellites and calculates its distance from satellites. This is done by measuring the time required for the signal to travel from satellite to the receiver.

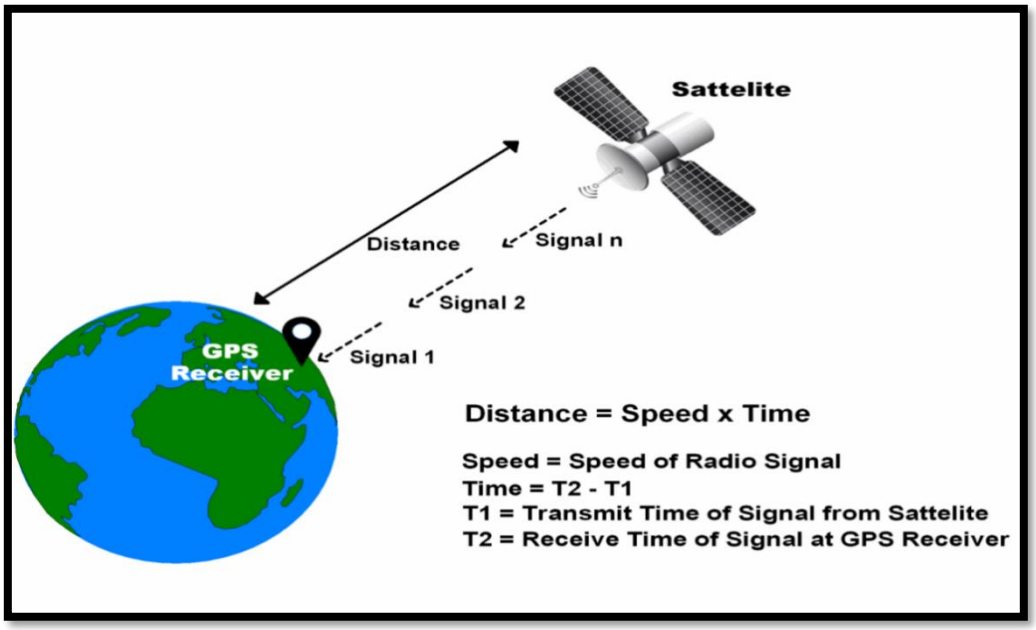

Fig. 4.6(b): Calculation of GPS Receiver

GPS Distance Calculation

Distance = Speed x Time

Where,

Speed = Speed of Radio signal which is approximately equal to the speed of light i.e. $3 * 10^8$

Time = Time required for a signal to travel from the satellite to the receiver.

By subtracting the sent time from the received time, we can determine the travel time.

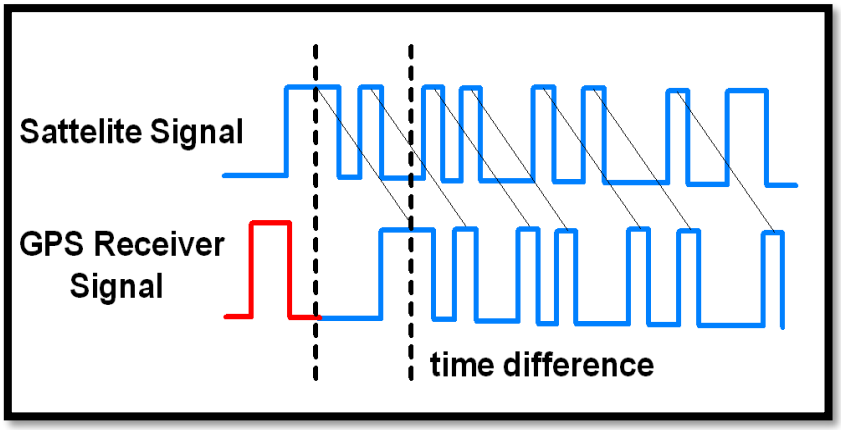

Fig.4.6(c): Time Difference of Sattelite Signal and GPS Receiver Signal

GPS Signal Time Difference

To determine distance, both the satellite and GPS receiver generate the same pseudocode signal at the same time.

The satellite transmits the pseudocode; which is received by the GPS receiver. These two signals are compared and the difference between the signals is the travel time.

Now, if the receiver knows the distance from 3 or more satellites and their location (which is sent by the satellites), then it can calculate its location by using Trilateration method.

Trilateration:

Trilateration is a mathematical technique used by a global positioning system (GPS) device to determine user position, speed, and elevation. By constantly receiving and analyzing radio signals from multiple GPS satellites and applying the geometry of circles, spheres, and triangles, a GPS device can calculate the precise distance or range to each satellite being tracked.

How Trilateration Works

Trilateration is a sophisticated version of triangulation, though it does not use the measurement of angles in its calculations. Data from a single satellite provides a general location of a point within a large circular area on the Earth's surface. Adding data from a second satellite allows the GPS to narrow the specific location of that point down to a region where the two areas of satellite data overlap. Adding data from a third

satellite provides an accurate position of the point on the Earth's surface.

All the GPS devices require three satellites for an accurate calculation of position. Data from a fourth satellite—or even more than four satellites—further enhance the precision of the point's location, and also allows factors such as elevation or, in the case of aircraft, altitude to also be calculated. GPS receivers routinely track four to seven satellites simultaneously and use trilateration to analyze the information.

GPS Receiver

GPS receiver module gives output in standard (National Marine Electronics Association) NMEA string format. It provides output serially on Tx pin with default 9600 Baud rate.

This NMEA string output from GPS receiver contains different parameters separated by commas like longitude, latitude, altitude, time etc. Each string starts with '$' and ends with carriage return/line feed sequence.

E.g.

$GPGGA,184237.000,1829.9639,N,07347.6174,E,1,05,2.1,607.1,M,-64.7,M,,0000*7D

$GPGSA,A,3,15,25,18,26,12,,,,,,,5.3,2.1,4.8*36

$GPGSV,3,1,11,15,47,133,46,25,44,226,45,18,37,238,45,26,34,087,40*72

$GPGSV,3,2,11,12,27,184,45,24,02,164,26,29,58,349,,05,26,034,*7F

$GPGSV,3,3,11,21,25,303,,02,11,071,,22,01,228,*40

$GPRMC,184237.000,A,1829.9639,N,07347.6174,E,0.05,180.19,230514,,,A*64

Fig.4.6(d): Pin Description of GPS Module

GPS Receiver Module

VCC: Power Supply 3.3 – 6 V

GND: Ground

TX: Transmit data serially which gives information about location, time etc.

RX: Receive Data serially. It is required when we want to configure GPS module.

Check GPS Module

Before Interfacing GPS module with PIC18F4550 microcontroller, we can check the output of GPS module. From that string, we can extract information like longitude, latitude, time which is helpful to find location and timing information.

To do this, connect this GPS module to the PC via USB to Serial converter or DB9 connector. Also, it is necessary to keep antenna of GPS module on proper location.

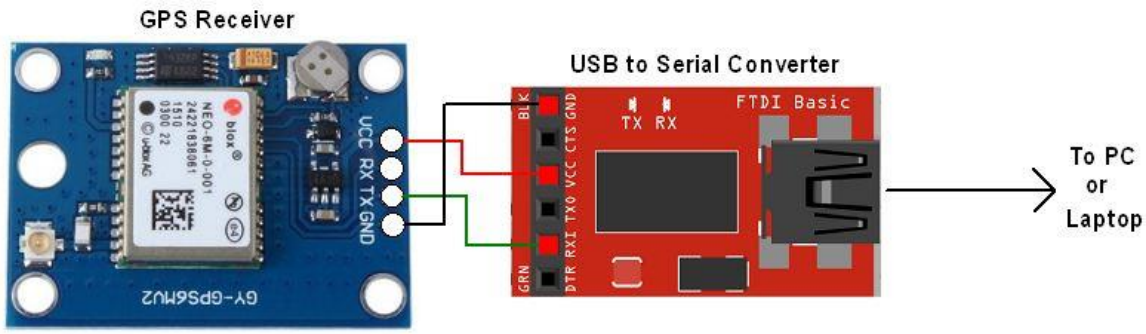

Fig.4.6(e): Checking of GPS by connecting to PC

GPS Serial Interface

1. Now open any serial terminal e.g. Realterm, Hyper terminal, Putty etc. on PC/laptop.
2. Open the PORT with 9600 baud rate.
3. The terminal will show data coming from GPS receiver module.

The output data from GPS receiver module displaying on a serial terminal as follows.

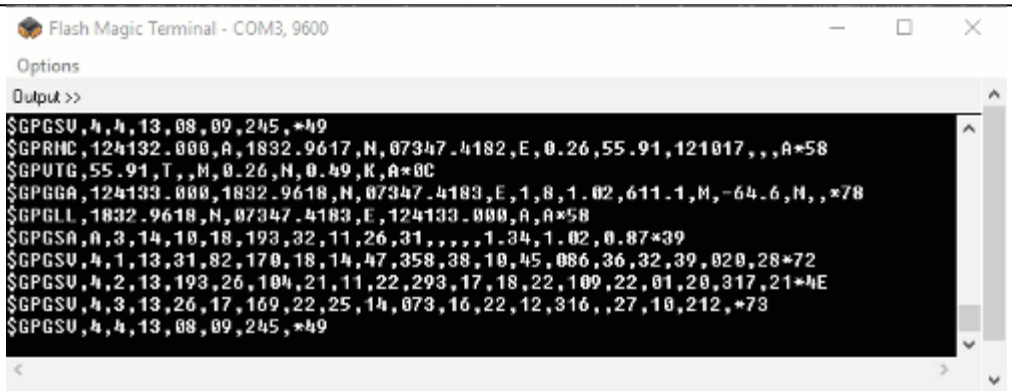

Fig.4.6(f): Output data from GPS receiver module

In the above string, the NMEA string starting with "$GPGGA" is most popularly used. It provides us Time, Longitude, Latitude and Altitude along with directions. This information is helpful to find Time and Location.

4.7.5. GSM/GPRS Module

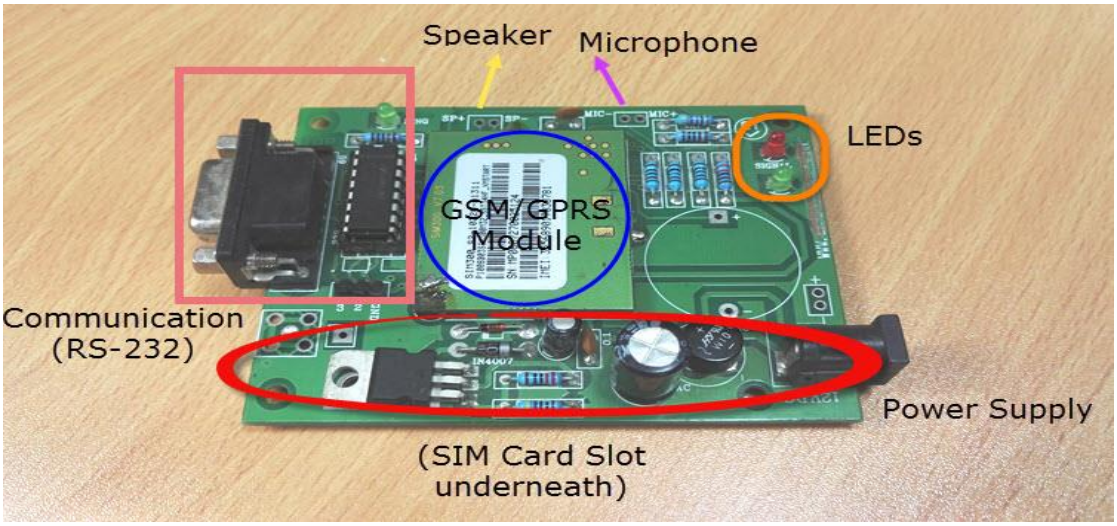

Fig.4.7: GPRS Module(GSM)

GPRS Modules are one of the commonly used communication modules in embedded systems. A GPRS Module is used to enable communication between a microcontroller (or a microprocessor) and the GPRS Network. Here, GSM stands for Global System for Mobile Communication and GPRS stands for General Packet Radio Service.

A GPRS MODEM comprises of a GPRS Module along with some other components like communication interface (like Serial Communication – RS-232), power supply and some indicators. With the help of this communication interface, we

can connect the GSM GPRS Module on the GPRS MODEM with an external computer (or a microcontroller).

GPRS Modules allow microcontrollers to have a wireless communication with other devices and instruments. Such wireless connectivity of microcontroller opens up to wide range of applications like Home Automation, Home Security Systems, Disaster Management, Medical Assistance, Vehicle Tracking, Online Banking, E – Commerce etc. to name some.

What is GPRS?

GPRS or General Packet Radio Service is an extension of the GSM Network. GPRS is an integrated part of the GSM Network which provides an efficient way to transfer data with the same resources as GSM Network.

Originally, the data services (like internet, multimedia messaging etc.) in the GSM Network used a circuit – switched connection. In this type, the access time for the network are long and the charges for the data were based on the connection time. Also, this type of connection is not suitable for transmitting bursts of data.

With the integration of GPRS, a packet – switching based data service, in to the GSM Network, the scene of data services has changes. In GPRS based packet – switching networks, the user device doesn't hold the resources for a continuous time but efficiently uses a common pool.

The access time in GPRS is very small and the main advantage is that it allows bursts of data to be transmitted. Also, the charges for data are based on the usage and not on the connection time.

Difference between a Module, MODEM and Mobile (System)

A GSM/GPRS Module is a device or chip that is actually responsible for the wireless communication with the GSM Network.

A GSM/GPRS MODEM is device that modulates and demodulates the signals from the Wireless Network and allows internet connectivity. A GSM MODEM generally consists of a GSM Module along with some other components like a SIM Card, a device to modulate and demodulate the signals and power supply.

A System, like a mobile phone for example, is a complete device that has a

GSM Module (might be integrated in the processor), a GSM MODEM (even this might be integrated) and other components like processor, screen, keypad, speakers, microphone etc.

Peaking of GSM/GPRS MODEMS, they are a class of Wireless MODEMS, which use the GSM/GPRS Network for transmitting and receiving data. Wireless MODEMS are also available for different cellular networks like CDMA, GSM, UMTS and LTE and different data services like GPRS, EDGE, HSDPA and LTE.

GPRS Module

A GPRS Module is an IC or chip that connects to the GSM Network using a SIM (Subscriber Identity Module) and Radio Waves. The common radio frequencies in which a typical GSM Module operates are 850MHz, 900MHz, 1800MHz and 1900MHz.

Since it is not possible to interface a GPRS Module directly to an external device like a microcontroller, we need a setup like shown in the following image.

It consists of the GPRS Module, slot for inserting a SIM Card, RS-232 Interface for connecting with computer or a microcontroller, signal status LED, power supply and a provision for connecting microphone and speaker.

Each GPRS Module is unique and it can be differentiated by its IMEI Number. IMEI or International Mobile Equipment Identity Number is a 15 – digit unique number associated with mobile phone, satellite phones and other GSM Network devices.

With the help of this GPRS Module, we can do the following tasks.

- Make, receive or reject voice calls
- Send, receive or delete SMS messages in the SIM Card
- Add, read and search the contacts in the SIM Card
- Send and receive data to / from the GPRS Network through GPRS

All the above mentioned tasks can be accomplished with the help of ATtention Commands or AT Commands. AT Commands are parts of Hayes Command Set, which are defined originally for a modem. GSM Network also implements a similar AT like commands for its GSM Modules.

The processor or controller, to which the GSM/GPRS Module is connected to,

is responsible for sending the AT Commands to the module. In response, the GSM Module performs command specific tasks like answering a phone call; send an SMS Message, etc.

Even through the AT Commands may seem generic; it is advisable to refer with the data provided by the manufacturer of the GSM Module for correct and complete list of AT Commands.

Syntax of AT Commands

Generally, AT Commands consists of three parts: the Prefix, the Body or Command and the Termination. The Prefix part of the command consists of either "AT" or "at". The body or command is the actual command given to the GSM Module. The Termination character is by default Carriage Return <CR>.

Before seeing an example, we should know that AT Commands are categorized in to three types: Basic AT Commands, S Parameter AT Commands and Extended AT Commands.

Example for the Basic Commands: ATCMD1<CR>, where AT is the prefix of the command line, CMD1 is the body of the command and <CR> command terminator character.

NOTE: Basic commands never begin with +.

Example for Extended Commands: AT+CMD1<CR>.

NOTE: Extended AT Commands always begin with +.

NOTE: The Carriage Return <CR> will be omitted from the next command example or syntax. You have to assume them as a part of the command even though they are not inserted in the text here.

Types of AT Command Operations

Based on the operation performed by the AT Commands, they are again divided in to four types: Test Commands, Read Commands, Write Commands and Execution Commands. We will now see the definition and syntax of all these command types.

Test Commands: These commands are used to test whether the command exists (supported by the GSM GPRS Module) or not and also checks for the range of a

command's subparameters. When a Test Command is given to the GSM GPRS Module, it returns the list of all the parameters and also the range set of the parameters.

The syntax of a Test Command is ATCMD1=?<CR>

Example for Test Command: AT+CGMI=? (Request manufacturer identification)

Read Commands: Read Commands will return the current value of the parameter. Using these commands, we can read the current settings of the GSM GPRS Module.

The syntax for Read Commands is: ATCMD1?<CR>

Example for Read Command: AT+CSCA? (Query for Service Center)

Write or Set Commands: Set Commands will attempt to change or modify the settings of the GSM GPRS Module by setting a new parameter for the particular command.

The syntax for Set Commands is: ATCMD1=value1,value2,value3…valuen<CR>

An example for Set Command: AT+CMGF=1 (Set Message format to TEXT Mode).

Execution Command: These commands perform an operation like send an SMS, retrieving information about battery charging status etc. They read the non – variable subparameters that are affected by the GSM Module.

Syntax of Execution Commands: ATCMD1<CR>

Example for Execution Commands: AT+CMGS=<number><CR><text message><CTRL-Z> (Sends text message to the number).

Information Responses and Final Codes

After sending the AT Commands to the GSM GPRS Module, we have look for the response. For example, if we send the command as AT+CGMI<CR> to the GSM Module, then the response would be as follows.

<CR><LF>Apple<CR><LF>

<CR><LF>OK<CR><LF>

Here, <CR> is Carriage Return and <LF>is Line Feed.

In a HyperTerminal, if you entered AT+CGMI<CR>, the response will look something like this.

AT+CGMI <– Command entered

Apple <– Information Response

OK <– Final Code

The syntax of the information response and final command is as follows:

<Carriage Return><Line Feed><Information Response / Final Result Code><Carriage Return><Line Feed>

<CR><LF><Response><CR><LF>

NOTE: The sequence of execution of commands will be first commands, then second command, followed by the rest i.e. a sequential execution of commands.

If there is an error anywhere in the execution, an error code is returned by the GSM Module and the execution of further commands is terminated.

Frequently used AT Commands

In this list, you can find out some of the most commonly used AT Commands. For a complete list of AT Commands and their definitions, it is advised to refer the manufacturer data. The <Carriage Return> or <CR> is denoted by this symbol ↲.

To check the communication between the GSM Module and the host (Computer)

AT ↲

OK

To make a voice call

ATD9848032919; ↲

To answer or receive an incoming call

ATA ↲

To redial the last number

ATDL ↲

To disconnect a call

ATH ↲

To set the message mode to text mode

AT+CMGF=1 ↲

OK

To send a text message

AT+CMGS=”9848032919” ↲

Message

CTRL+Z

4.7.6 Buzzer:

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke. Buzzer is an integrated structure of electronic transducers, DC power supply, widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timers and other electronic products for sound devices. Active buzzer 5V Rated power can be directly connected to a continuous sound, this section dedicated sensor expansion module and the board in combination, can complete a simple circuit design, to "plug and play.

Fig.4.8: Buzzer

Table.4.3: Buzzer Pin Configuration

Pin Number	Pin Name	Description
1	Positive	Identified by (+) symbol or longer terminal lead. Can be powered by 5V DC
2	Negative	Identified by short terminal lead. Typically connected to the ground of the circuit

Buzzer Features and Specifications

- Rated Voltage: 6V DC
- Operating Voltage: 4-8V DC
- Rated current: <30mA
- Sound Type: Continuous Beep
- Resonant Frequency: ~2300 Hz
- Small and neat sealed package
- Breadboard and Perf board friendly

How to use a Buzzer

A buzzer is a small yet efficient component to add sound features to our project/system. It is very small and compact 2-pin structure hence can be easily used on breadboard, Perf Board and even on PCBs which makes this a widely used component in most electronic applications.

There are two types are buzzers that are commonly available. The one shown here is a simple buzzer which when powered will make a Continuous Beeeeeeppp.... sound, the other type is called a readymade buzzer which will look bulkier than this and will produce a Beep. Beep. Beep. Sound due to the internal oscillating circuit present inside it. But, the one shown here is most widely used because it can be customized with help of other circuits to fit easily in our application.

This buzzer can be used by simply powering it using a DC power supply ranging from 4V to 9V. A simple 9V battery can also be used, but it is recommended to use a regulated +5V or +6V DC supply. The buzzer is normally associated with a switching circuit to turn ON or turn OFF the buzzer at required time and require interval.

Applications of Buzzer

- Alarming Circuits, where the user has to be alarmed about something
- Communication equipment's
- Automobile electronics
- Portable equipment's, due to its compact size

4.7.7. Vibration Sensor:

A vibration sensor is a device designed to detect and measure oscillations, movements, or vibrations in a system. It converts mechanical motion into an electrical

signal, allowing it to monitor vibrations in various applications, including industrial machinery, security systems, and vehicle safety mechanisms. These sensors come in different types, such as piezoelectric, accelerometer-based, and MEMS (Micro-Electro-Mechanical Systems) vibration sensors, each offering unique advantages depending on sensitivity, accuracy, and response time.

In the context of security and anti-theft systems, vibration sensors play a crucial role in detecting unauthorized activities. For example, in a vehicle anti-theft system, a vibration sensor can detect unusual movements, such as forced entry attempts or unauthorized tampering. When triggered, it can activate an alarm, send alerts to the owner, and even communicate with a microcontroller (like ESP32) to take additional actions, such as capturing images or sending location updates via GPS and GSM. This feature enhances security by providing real-time theft detection and response.

Vibration sensors are widely used in automotive, industrial, and home security applications due to their ability to detect even minor disturbances. In industrial machinery, they help prevent equipment failure by monitoring vibrations that indicate wear and tear. In home security systems, they can detect window or door tampering attempts. Their versatility and reliability make them an essential component in modern electronic security and monitoring systems, improving both safety and efficiency

Fig.4.9: Vibration Sensor

4.7.8. Relay:

What is a relay?

A relay is an electromagnetic switch that is used to turn on and turn off a circuit by a low power signal, or where several circuits must be controlled by one signal.

Most of the high end industrial application devices have relays for their effective working. Relays are simple switches which are operated both electrically and mechanically. Relays consist of an electromagnet and also a set of contacts. The switching mechanism is carried out with the help of the electromagnet. There are also other operating principles for its working. But they differ according to their applications. Most of the devices have the application of relays.

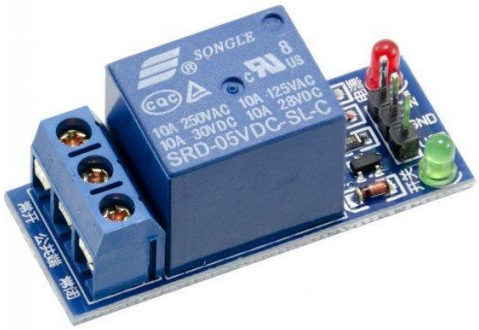

Fig.4.10(a): Relay Module

Pin Diagram:

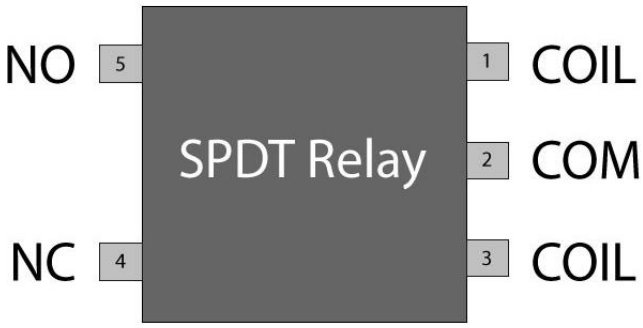

Fig.4.10(b): Pin diagram of Relay Module

Why is a relay used?

The main operation of a relay comes in places where only a low-power signal can be used to control a circuit. It is also used in places where only one signal can be

used to control a lot of circuits. The application of relays started during the invention of telephones. They played an important role in switching calls in telephone exchanges. They were also used in long distance telegraphy. They were used to switch the signal coming from one source to another destination. After the invention of computers they were also used to perform Boolean and other logical operations. The high end applications of relays require high power to be driven by electric motors and so on. Such relays are called contactors.

Relay Design

- There are only four main parts in a relay. They are
- Electromagnet
- Movable Armature
- Switch point contacts
- Spring

The figures given below show the actual design of a simple relay.

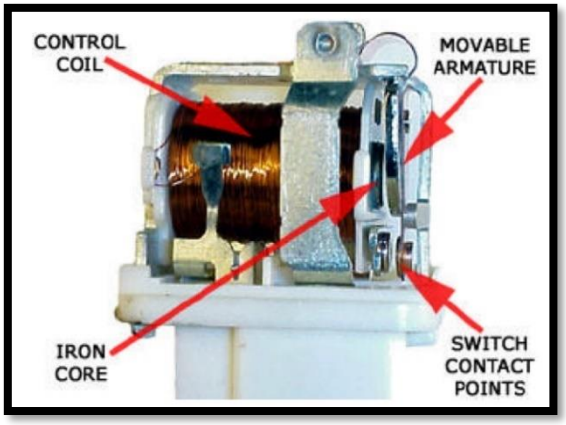

Fig.4.10(c): Actual Design of simple Relay

Relay Construction

It is an electro-magnetic relay with a wire coil, surrounded by an iron core. A path of very low reluctance for the magnetic flux is provided for the movable armature and also the switch point contacts.

The movable armature is connected to the yoke which is mechanically connected to the switch point contacts. These parts are safely held with the help of a spring. The spring is used so as to produce an air gap in the circuit when the relay

becomes de-energized.

How relay works?

The relay function can be better understood by explaining the following diagram given below.

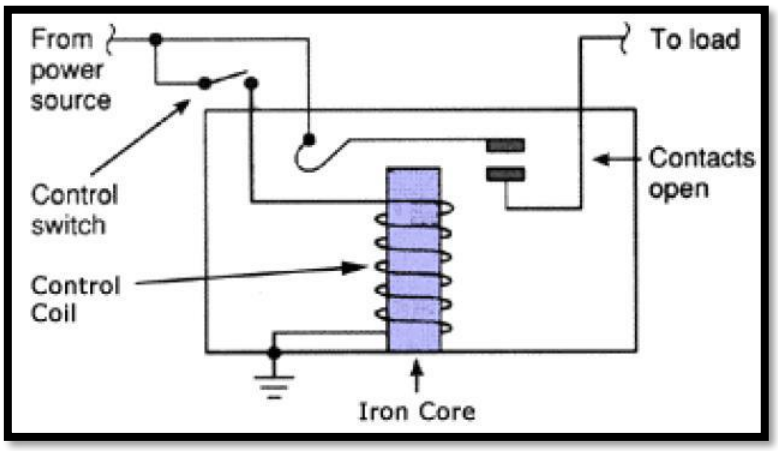

Fig.4.10(d): Working of Relay

Relay Design

The diagram shows an inner section diagram of a relay. An iron core is surrounded by a control coil. As shown, the power source is given to the electromagnet through a control switch and through contacts to the load. When current starts flowing through the control coil, the electromagnet starts energizing and thus intensifies the magnetic field. Thus the upper contact arm starts to be attracted to the lower fixed arm and thus closes the contacts causing a short circuit for the power to the load. On the other hand, if the relay was already de-energized when the contacts were closed, then the contact move oppositely and make an open circuit.

As soon as the coil current is off, the movable armature will be returned by a force back to its initial position. This force will be almost equal to half the strength of the magnetic force. This force is mainly provided by two factors. They are the spring and also gravity.

Relays are mainly made for two basic operations. One is low voltage application and the other is high voltage. For low voltage applications, more preference will be given to reduce the noise of the whole circuit. For high voltage applications, they are mainly designed to reduce a phenomenon called arcing.

Relay Basics

The basics for all the relays are the same. Take a look at a 4 pin relay shown below. There are two colors shown. The green color represents the control circuit and the red color represents the load circuit. A small control coil is connected onto the control circuit. A switch is connected to the load. This switch is controlled by the coil in the control circuit. Now let us take the different steps that occur in a relay.

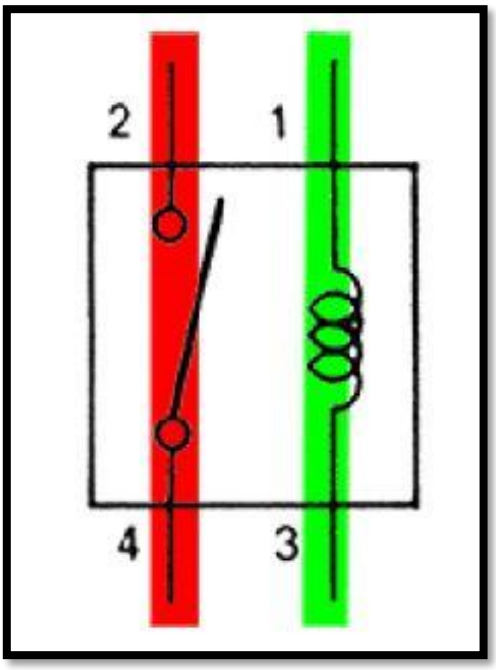

Fig.4.10(e): Relay Basics

Relay operation

- **Energized Relay (ON)**

As shown in the circuit, the current flowing through the coils represented by pins 1 and 3 causes a magnetic field to be aroused. This magnetic field causes the closing of the pins 2 and 4. Thus the switch plays an important role in the relay working. As it is a part of the load circuit, it is used to control an electrical circuit that is connected to it. Thus, when the electrical relay in energized the current flow will be through the pins 2 and 4.

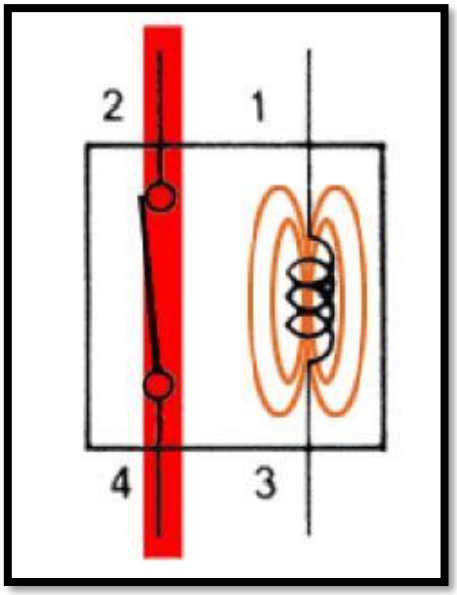

Fig.4.10(f): Energized Relay (ON)

- **De – Energized Relay (OFF)**

As soon as the current flow stops through pins 1 and 3, the relay switch opens and thus the open circuit prevents the current flow through pins 2 and 4. Thus the relay becomes de-energized and thus in off position.

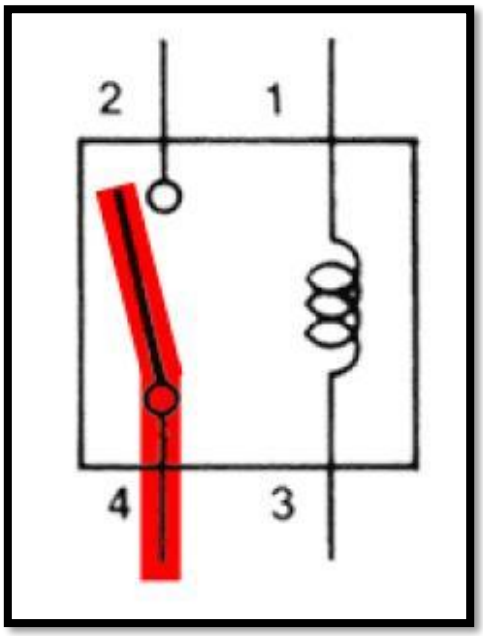

Fig.4.10(g): De-Energized Relay (OFF)

In simple, when a voltage is applied to pin 1, the electromagnet activates,

causing a magnetic field to be developed, which goes on to close the pins 2 and 4 causing a closed circuit. When there is no voltage on pin 1, there will be no electromagnetic force and thus no magnetic field. Thus the switches remain open.

Pole and Throw

Relays have the exact working of a switch. So, the same concept is also applied. A relay is said to switch one or more poles. Each pole has contacts that can be thrown in mainly three ways. They are

- **Normally Open Contact (NO):** NO contact is also called a make contact. It closes the circuit when the relay is activated. It disconnects the circuit when the relay is inactive.
- **Normally Closed Contact (NC):** NC contact is also known as break contact. This is opposite to the NO contact. When the relay is activated, the circuit disconnects. When the relay is deactivated, the circuit connects.
- **Change-over (CO) / Double-throw (DT) Contacts:** This type of contacts are used to control two types of circuits. They are used to control a NO contact and also a NC contact with a common terminal. According to their type they are called by the names break before make and make before break contacts.

Relays can be used to control several circuits by just one signal. A relay switches one or more poles, each of whose contacts can be thrown by energizing the coil.

Relays are also named with designations like

- **Single Pole Single Throw (SPST)**: The SPST relay has a total of four terminals. Out of these two terminals can be connected or disconnected. The other two terminals are needed for the coil to be connected.
- **Single Pole Double Throw (SPDT):** The SPDT relay has a total of five terminals. Out of these two are the coil terminals. A common terminal is also included which connects to either of two others.
- **Double Pole Single Throw (DPST):** The DPST relay has a total of six terminals. These terminals are further divided into two pairs. Thus they can act as two SPST which are actuated by a single coil. Out of the six terminals two of them are coil

terminals.

- **Double Pole Double Throw (DPDT)**: The DPDT relay is the biggest of all. It has mainly eight relay terminals. Out of these two rows are designed to be change over terminals. They are designed to act as two SPDT relays which are actuated by a single coil.

Relay Applications

- A relay circuit is used to realize logic functions. They play a very important role in providing safety critical logic.
- Relays are used to provide time delay functions. They are used to time the delay open and delay close of contacts.
- Relays are used to control high voltage circuits with the help of low voltage signals. Similarly they are used to control high current circuits with the help of low current signals.
- They are also used as protective relays. By this function all the faults during transmission and reception can be detected and isolated.

Application of Overload Relay

Overload relay is an electro-mechanical device that is used to safeguard motors from overloads and power failures. Overload relays are installed in motors to safeguard against sudden current spikes that may damage the motor. An overload relay switch works in characteristics with current over time and is different from circuit breakers and fuses, where a sudden trip is made to turn off the motor. The most widely used overload relay is the thermal overload relay where a bimetallic strip is used to turn off the motor. This strip is set to make contact with a contactor by bending itself with rising temperatures due to excess current flow. The contact between the strip and the contactor causes the contactor to de-energize and restricts the power to the motor, and thus turns it off.

Another type of overload motor is the electronic type which continuously watches the motor current, whereas the thermal overload relay shuts off the motor depending on the rise of temperature/heat of the strip.

All overload relays available to buy comes in different specifications, the most

important of them being the current ranges and response time. Most of them are designed to automatically reset to work after the motor is turned back on.

Relay Selection

You must note some factors while selecting a particular relay. They are

- Protection Different protections like contact protection and coil protection must be noted. Contact protection helps in reducing arcing in circuits using inductors. A Coil protection helps in reducing surge voltage produced during switching.
- Look for a standard relay with all regulatory approvals.
- Switching time Ask for high speed switching relays if you want one.
- Ratings There are current as well as voltage ratings. The current ratings vary from a few amperes to about 3000 amperes. A In case of voltage ratings, they vary from 300 Volt AC to 600 Volt AC. There are also high voltage relays of about 15,000 Volts.
- Type of contact used whether it is a NC or NO or closed contact.
- Select Make before Break or Break before Make contacts wisely.
- Isolation between coil circuit and contacts.

4.7.9 Mosquito Bat Module:

Introduction

The mosquito bat module is a crucial component used in electric mosquito swatters. It functions as a high-voltage generator circuit that produces an electric shock to eliminate mosquitoes and other small insects. The module consists of an oscillator circuit, a step-up transformer, and a voltage multiplier, which together generate a high voltage efficiently.

Features

- High Voltage Output: Converts low-voltage DC input into a high-voltage AC output ranging from 2000V to 4000V.
- Low Power Consumption: Optimized design ensures extended battery life.
- Rechargeable Battery Compatibility: Can be used with lithium-ion, lead-acid, or AA/AAA batteries.
- Safety Mechanisms: Insulated components prevent accidental shocks.

- Durable and Lightweight: Compact design for easy portability.
- Automatic Discharge Protection: Prevents electric shock when not in use.

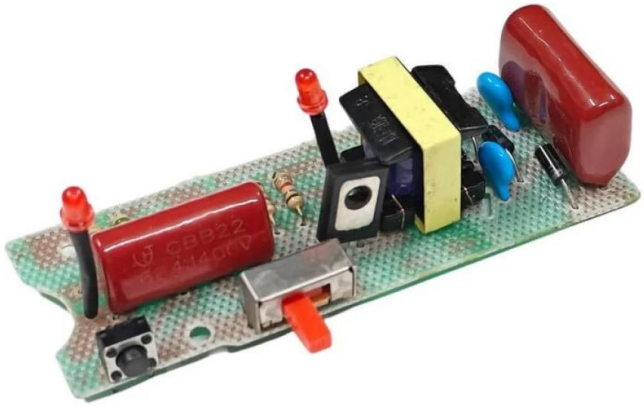

Fig.4.11: Mosquito Module

Working Principle

The mosquito bat module operates by taking a low DC voltage input and converting it into high-frequency AC. This AC voltage is then stepped up using a transformer, and a voltage multiplier further increases the voltage. When a mosquito comes into contact with the metal mesh layers, the high voltage creates an electric field that electrocutes the insect instantly. Some modules also include a discharge circuit to prevent accidental shocks when the device is not in use.

Table 4.4: Components of the Mosquito Module

Component	Function
Oscillator Circuit	Converts DC input into high-frequency AC.
Step-Up Transformer	Increases voltage to several hundred volts.
Voltage Multiplier	Further amplifies voltage up to 4000V.
Grid Electrodes	Metal mesh layers that electrocute insects upon contact.
Battery Pack	Provides power to the module.
Charging Circuit	Allows recharging via USB or AC adapter.

Applications

- Household Use – Commonly used in electric mosquito bats for pest control.
- Commercial Use – Deployed in hotels, restaurants, and offices to prevent mosquito-borne diseases.

- Outdoor Protection – Useful in camping, trekking, and agricultural pest control.
- Livestock Protection – Helps keep insects away from animal enclosures.
- Automated Pest Control – Integrated into larger bug zapper systems.

Advantages

- Eco-Friendly: A chemical-free alternative to mosquito repellents.
- Cost-Effective: Reusable with rechargeable batteries.
- Portable and User-Friendly: Lightweight and ergonomic design.
- Quick Action: Instantly kills mosquitoes upon contact.

Limitations

- Battery Dependency: Requires regular charging.
- Limited Range: Works only upon direct mosquito contact.
- Safety Concerns: High-voltage grids can cause minor shocks if mishandled.

4.7.10 DC Gear Motor:

Introduction

A DC gear motor is a type of electric motor that integrates a gearbox to provide controlled motion with increased torque at lower speeds. These motors are widely used in automation, robotics, and mechanical systems where precise and powerful motion control is required.

Features

- High Torque Output – The integrated gearbox enhances torque while reducing speed.
- Variable Voltage Support – Available in different voltage ratings (6V, 12V, 24V, etc.).
- Speed Control – Adjustable using Pulse Width Modulation (PWM).
- Reversible Motion – Changes rotation direction by reversing the power supply polarity.
- Compact and Efficient – Small design with high durability.
- Long Operational Life – Designed to withstand wear and tear.

Fig.4.12 DC Gear Motor

Working Principle

A DC gear motor works by converting electrical energy into mechanical motion. When voltage is applied to the motor, it generates rotational movement. The integrated gearbox reduces the rotational speed while increasing torque. This controlled motion allows for precise applications in various mechanical and robotic systems. Some gear motors also include encoders for real-time feedback and speed regulation.

Table.4.5(a): Components of DC Gear Motor

Component	Function
DC Motor	Converts electrical energy into mechanical motion.
Gearbox	Reduces speed and increases torque.
Output Shaft	Rotates at a controlled speed.
Encoders (Optional)	Provide feedback for precise speed control.

Applications

- Robotics – Essential for robotic arms and autonomous systems.
- Industrial Automation – Used in conveyor belts, CNC machines, and packaging systems.
- Electric Vehicles – Commonly found in e-bikes, scooters, and remote-controlled vehicles.

- Agricultural Equipment – Helps in irrigation systems and automated farming devices.
- Medical Devices – Utilized in prosthetics, medical robots, and automated instruments.

Table.4.5(b): Technical Specifications:

Parameter	Specification
Operating Voltage	12V DC
No-Load Speed	200 RPM
Torque	10-50 kg.cm (varies by model)
Shaft Diameter	6mm
Current Draw	1A – 5A (depending on load)
Gear Ratio	Available in different ratios (10:1, 30:1, 100:1, etc.)

Wiring and Connection

A DC gear motor can be controlled using an H-Bridge circuit (such as L298N or L293D motor drivers). The wiring setup involves:

Table.4.5(c): Wiring and Connection:

Connection	Description
Motor Terminals	Wires from the motor connect to OUT1 and OUT2 on the motor driver.
Power Supply	The motor driver requires a voltage input (typically 12V) connected to VCC.
Microcontroller PWM Pins	Used for speed and direction control.

Advantages

- High Efficiency – Provides strong torque with low power consumption.
- Precise Control – Allows accurate speed and direction adjustments.
- Long Lifespan – Designed for continuous operation.

Limitations

- Gear Wear Over Time – Gears may require maintenance or replacement.
- Speed Limitations – Maximum speed depends on gear ratio.
- Heat Generation – Prolonged use may cause overheating.

4.8. SOFTWARE DESCRIPTION

4.8.1. Arduino IDE:

Arduino IDE where IDE stands for Integrated Development Environment – An official software introduced by Arduino.cc, that is mainly used for writing, compiling and uploading the code in the Arduino Device. Almost all Arduino modules are compatible with this software that is an open source and is readily available to install and start compiling the code on the go.

Introduction to Arduino IDE:

- Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.
- It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.
- It is easily available for operating systems like MAC, Windows, and Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more.
- Each of them contains a microcontroller on the board that is actually programmed
- and accepts the information in the form of code.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
- This environment supports both C and C++ languages.

How to install Arduino IDE:

You can download the Software from Arduino main website. As I said earlier, the software is available for common operating systems like Linux, Windows, and

MAX, so make sure you are downloading the correct software version that is easily compatible with your operating system.

- If you aim to download Windows app version, make sure you have Windows 8.1 or Windows 10, as app version is not compatible with Windows 7 or older version of this operating system.

The IDE environment is mainly distributed into three sections

- **1. Menu Bar**
- **2. Text Editor**
- **3. Output Pane**

As you download and open the IDE software, it will appear like an image below.

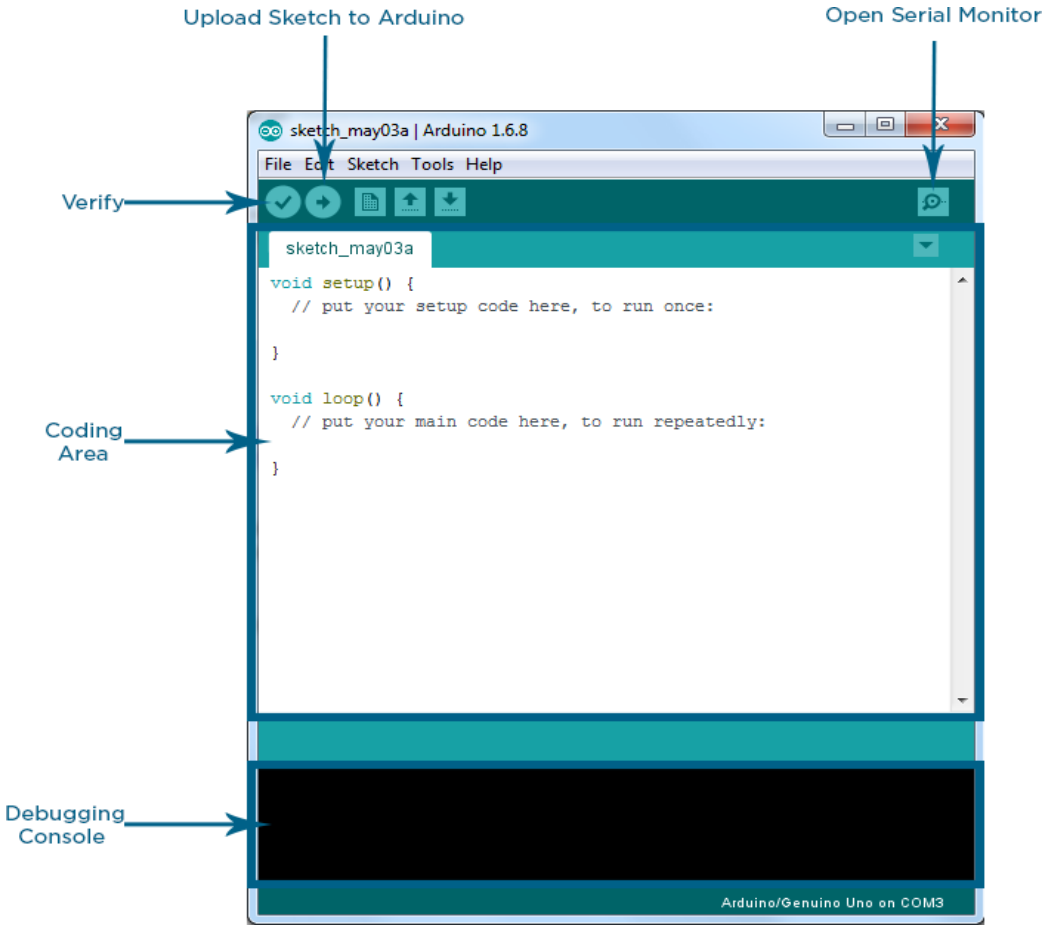

Fig.4.13(a): Arduino Ide DashBoard

The bar appearing on the top is called **Menu Bar** that comes with five different options as follow

- **File** – You can open a new window for writing the code or open an existing one. Following table shows the number of further subdivisions the file option is categorized into.

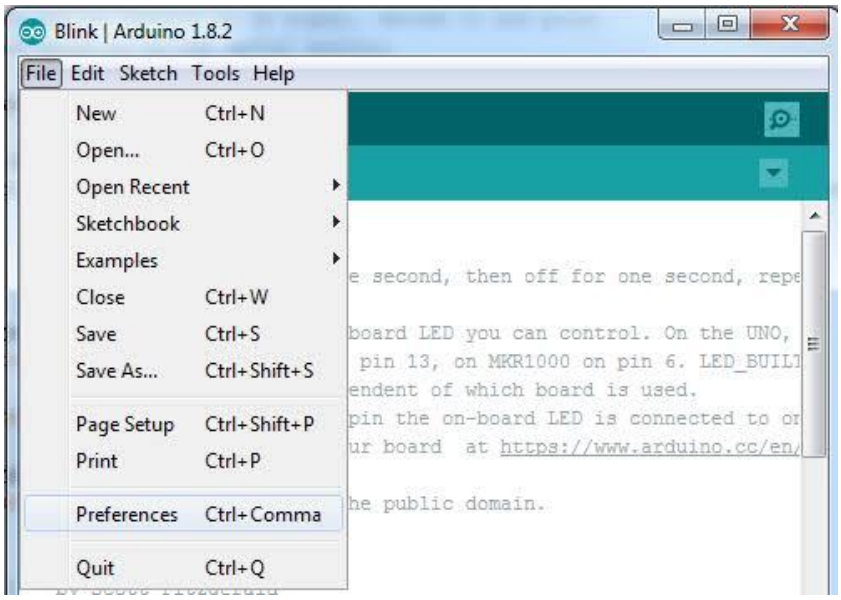

Fig.4.13(b): File Menu

As you go to the preference section and check the compilation section, the Output Pane will show the code compilation as you click the upload button.

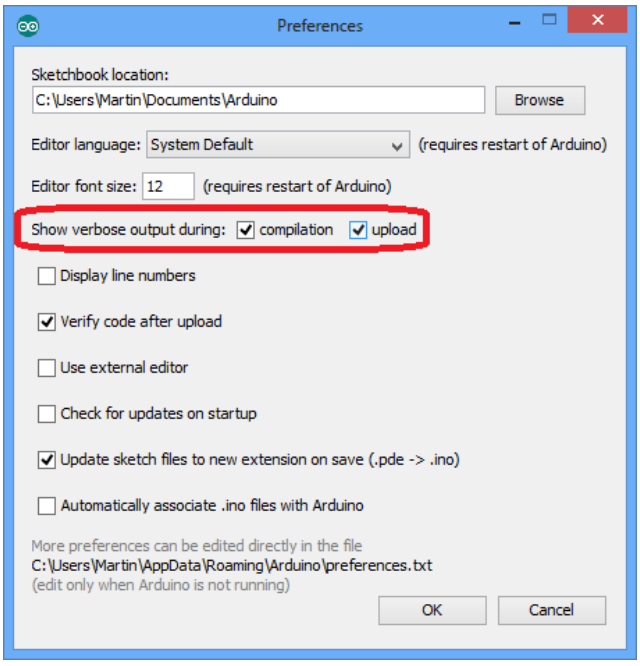

Fig.4.13(c): Preference Section

And at the end of compilation, it will show you the hex file it has generated for the recent sketch that will send to the Arduino Board for the specific task you aim to achieve.

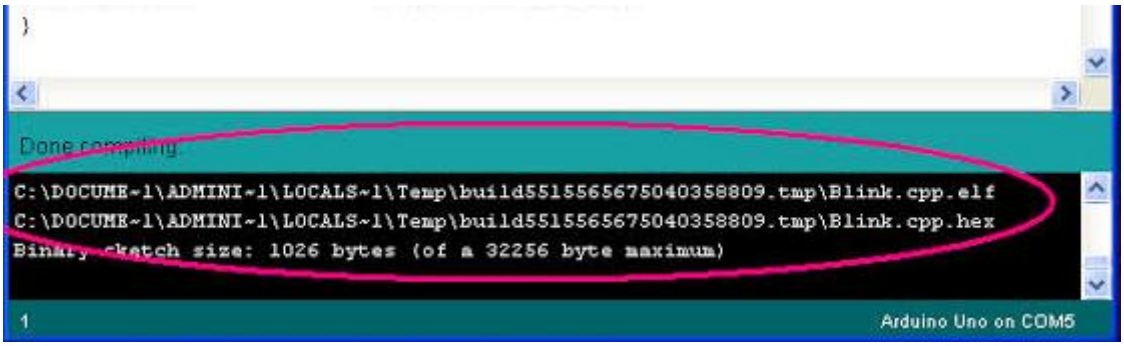

Fig.4.13(d): Output Pane

- **Edit** – Used for copying and pasting the code with further modification for font
- **Sketch** – For compiling and programming
- **Tools** – Mainly used for testing projects. The Programmer section in this panel is used for burning a bootloader to the new microcontroller.
- **Help** – In case you are feeling skeptical about software, complete help is available from getting started to troubleshooting.

The **Six Buttons** appearing under the Menu tab are connected with the running program as follow.

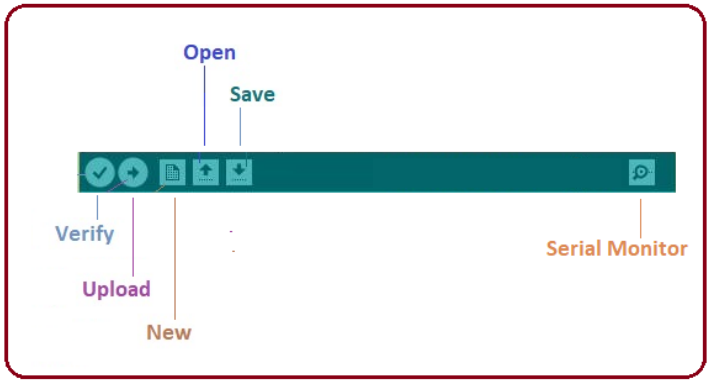

Fig.4.13(e): Tool bar Buttons

- The check mark appearing in the circular button is used to verify the code. Click this once you have written your code.
- The arrow key will upload and transfer the required code to the Arduino board.

- The dotted paper is used for creating a new file.
- The upward arrow is reserved for opening an existing Arduino project.
- The downward arrow is used to save the current running code.
- The button appearing on the top right corner is a Serial Monitor – A separate pop-up window that acts as an independent terminal and plays a vital role for sending and receiving the Serial Data. You can also go to the Tools panel and select Serial Monitor, or pressing Ctrl+Shift+M all at once will open it instantly. The Serial Monitor will actually help to debug the written Sketches where you can get a hold of how your program is operating. Your Arduino Module should be connected to your computer by USB cable in order to activate the Serial Monitor.
- You need to select the baud rate of the Arduino Board you are using right now. For my Arduino Uno Baud Rate is 9600, as you write the following code and click the Serial Monitor, the output will show as the image below.

Fig.4.13(f): Serial Monitor

The main screen below the Menu bard is known as a simple text editor used for writing the required code.

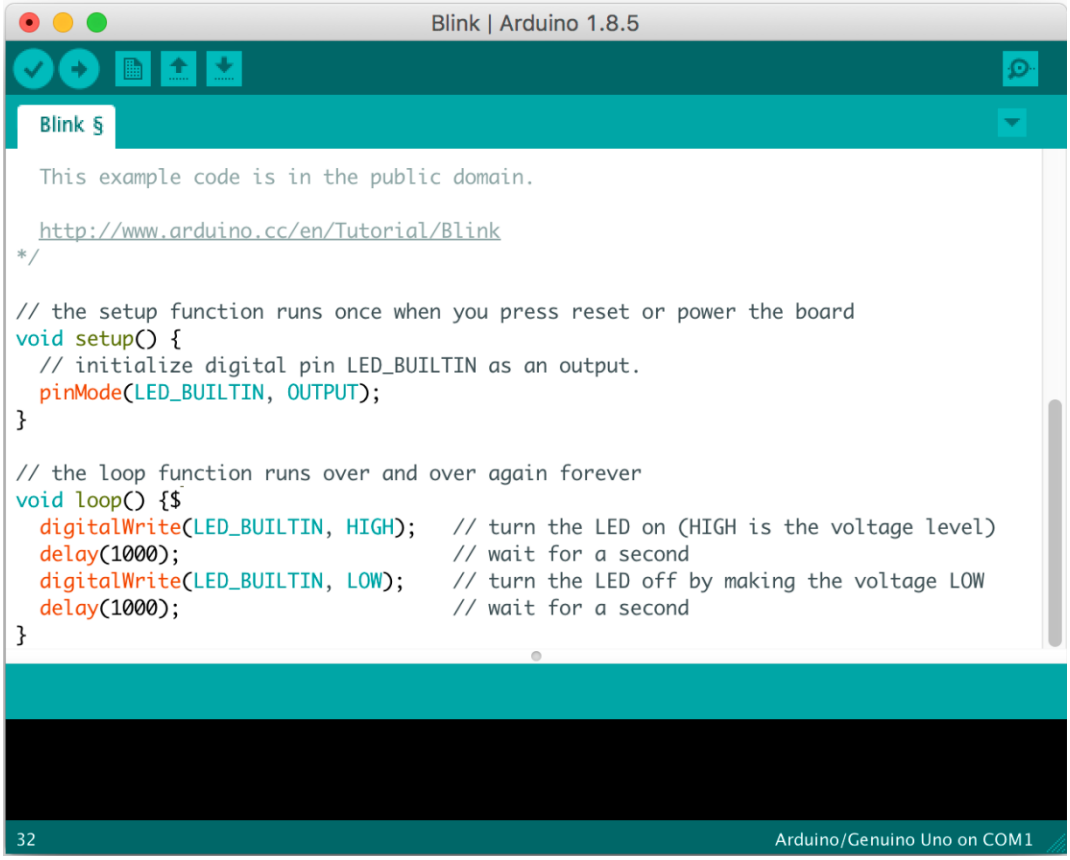

Fig.4.13(g): Simple Text Editor

The bottom of the main screen is described as an Output Pane that mainly highlights the compilation status of the running code: the memory used by the code, and errors occurred in the program. You need to fix those errors before you intend to upload the hex file into your Arduino Module.

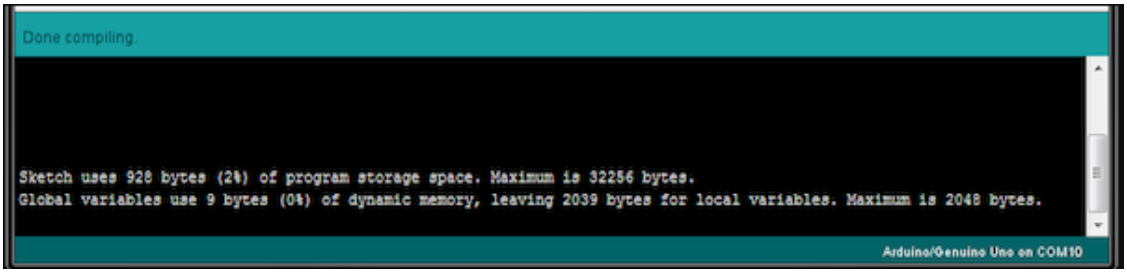

Fig.4.13(h): Output Pane Showing Compilation

More or less, Arduino C language works similar to the regular C language used for any embedded system microcontroller, however, there are some dedicated libraries used for calling and executing specific functions on the board.

4.8.2. Libraries:

Libraries are very useful for adding the extra functionality into the Arduino Module. There is a list of libraries you can add by clicking the Sketch button in the menu bar and going to Include Library.

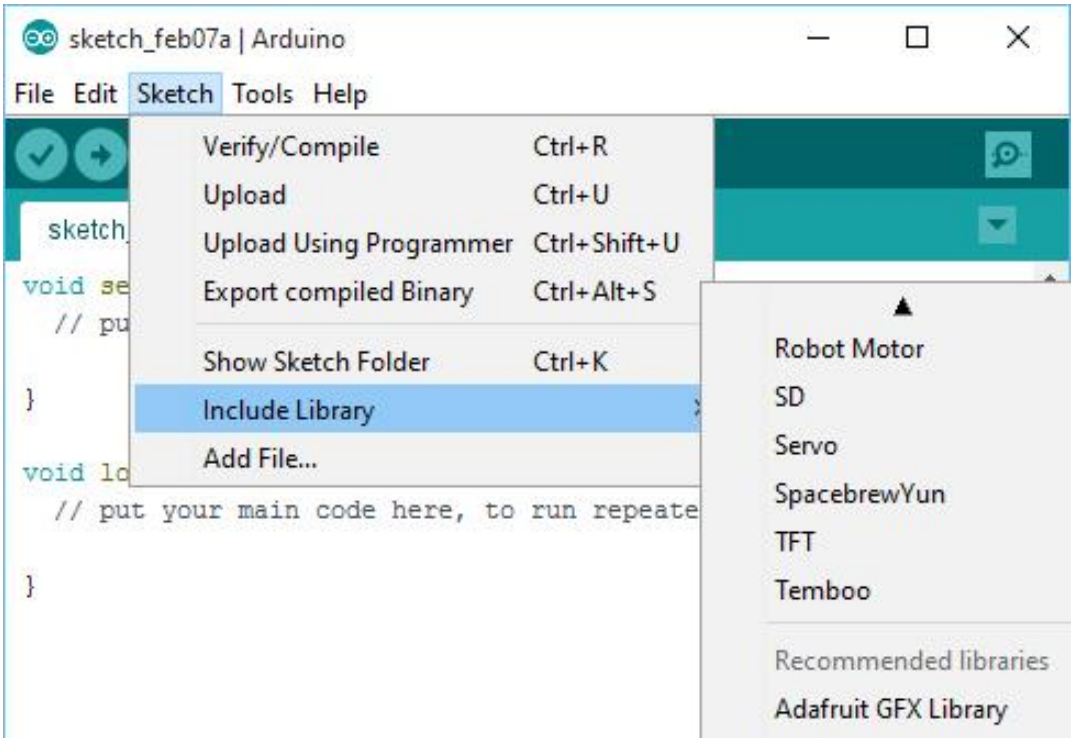

Fig.4.14(a): Sketch Menu and Libraries

As you click the Include Library and Add the respective library it will on the top of the sketch with a #include sign. Suppose, I Include the EEPROM library, it will appear on the text editor as

#include <EEPROM.h>.

Most of the libraries are preinstalled and come with the Arduino software. However, you can also download them from the external sources.

Making pins Input and output:

The digitalRead and digitalWrite commands are used for addressing and making the Arduino pins as an input and output respectively.

These commands are text sensitive i.e. you need to write them down the exact way they are given like digitalWrite starting with small “d” and write with capital “W”. Writing it down with Digitalwrite or digitalwrite won’t be calling or addressing any

function.

How to select the board:

In order to upload the sketch, you need to select the relevant board you are using and the ports for that operating system. As you click the Tools on the Menu, it will open like the figure below.

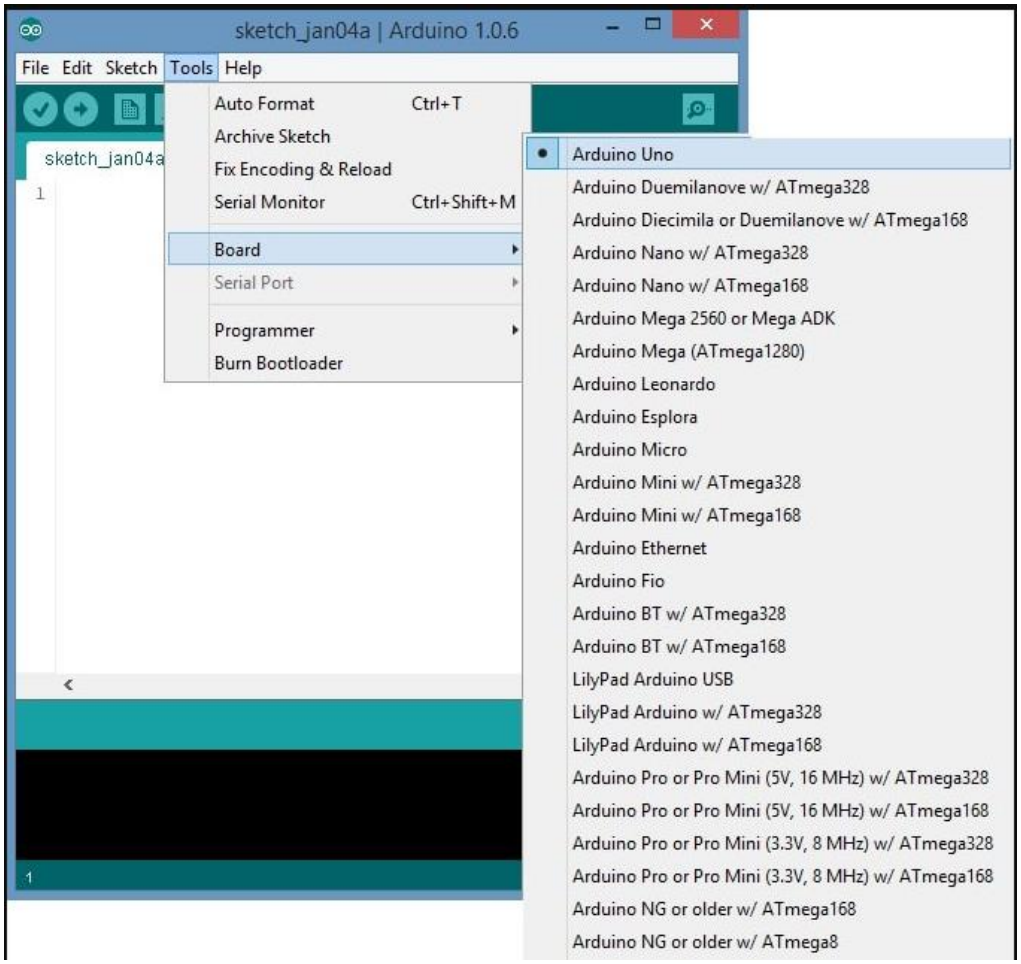

Fig.4.14(b): Different Arduino Boards

- Just go to the “Board” section and select the board you aim to work on. Similarly, COM1, COM2, COM4, COM5, COM7 or higher are reserved for the serial and USB board. You can look for the USB serial device in the ports section of the Windows Device Manager.

Following figure shows the COM4 that I have used for my project, indicating the Arduino Uno with COM4 port at the right bottom corner of the screen.

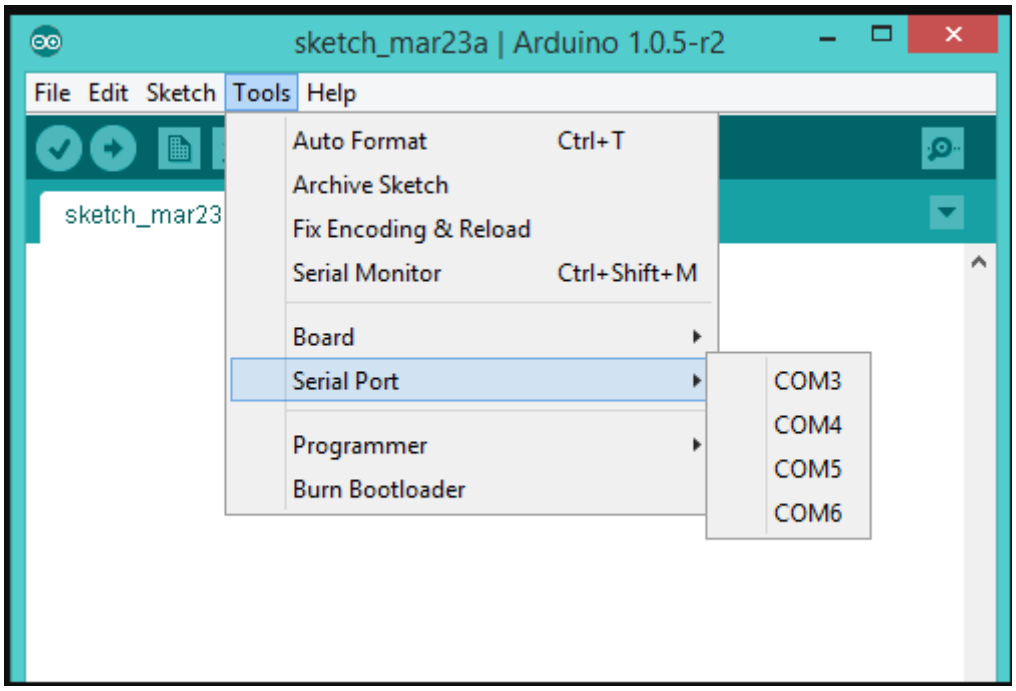

Fig.4.14(c): Serial Port Selection in Tools Menu

- After correct selection of both Board and Serial Port, click the verify and then upload button appearing in the upper left corner of the six button section or you can go to the Sketch section and press verify/compile and then upload.
- The sketch is written in the text editor and is then saved with the file extension ino.

It is important to note that the recent Arduino Modules will reset automatically as you compile and press the upload button the IDE software, however, older version may require the physical reset on the board.

- Once you upload the code, TX and RX LEDs will blink on the board, indicating the desired program is running successfully.

Note: The port selection criteria mentioned above is dedicated for Windows operating system only, you can check this Guide if you are using MAC or Linux.

- The amazing thing about this software is that no prior arrangement or bulk of mess is required to install this software, you will be writing your first program within 2 minutes after the installation of the IDE environment.

Boot Loader:

As you go to the Tools section, you will find a bootloader at the end. It is very helpful to burn the code directly into the controller, setting you free from buying the external burner to burn the required code.

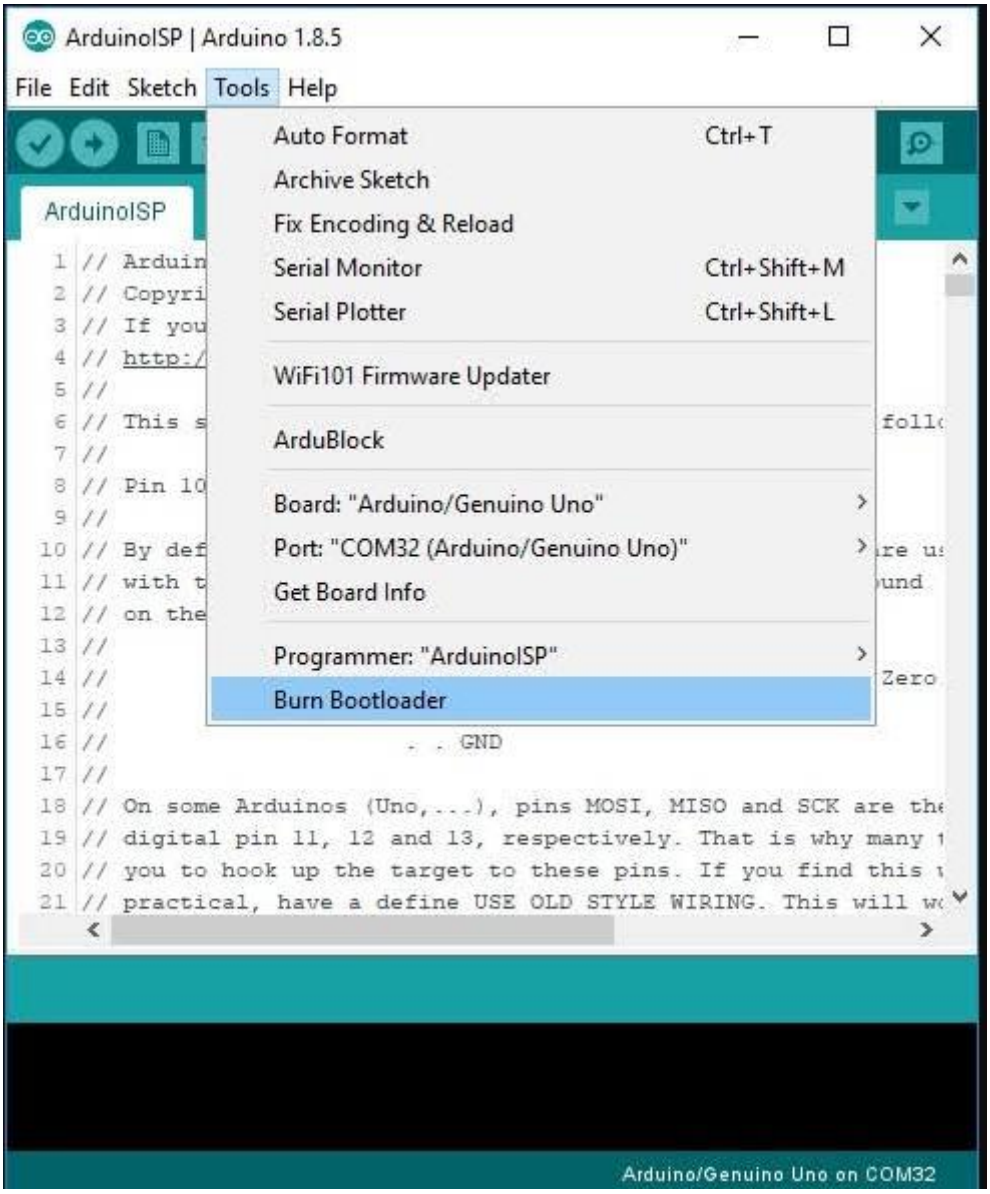

Fig.4.14(d): Boot Loader

When you buy the new Arduino Module, the bootloader is already installed inside the controller. However, if you intend to buy a controller and put in the Arduino module, you need to burn the bootloader again inside the controller by going to the Tools section and selecting the burn bootloader.

CHAPTER 5
TESTING AND RESULT ANALYSIS

5.1 Introduction:

Testing is a critical phase in the development of the Comprehensive Anti-Theft Vehicle Protection Framework Using Embedded Electronics and Python. It ensures that the system functions as expected under various conditions. This chapter presents the testing procedures, results, and analysis of the implemented system to validate its performance. The primary objectives of testing include verifying system functionality, ensuring reliability under real-world conditions, evaluating performance under stress scenarios, and identifying potential areas of improvement.

5.2 Testing Methodology:

The testing process is categorized into the following:

- Unit Testing: Verifies individual components like fingerprint authentication, GSM alerts, GPS tracking, vibration detection, and camera module to ensure they function correctly before integration.
- Integration Testing: Ensures seamless communication between sensors, microcontroller (ESP32), and communication modules (GSM, GPS, Wi-Fi).
- System Testing: Tests the entire system under real-world conditions, including vehicle movement, unauthorized access, and network fluctuations.
- Performance Testing: Evaluates response time, system accuracy, and reliability over extended operational periods.
- Security Testing: Checks for vulnerabilities in the system, such as hacking attempts, fingerprint spoofing, and unauthorized access through signal manipulation.
- Stress Testing: Assesses the system's behavior under extreme conditions, such as multiple intrusion attempts, rapid sensor triggering, and network congestion.
- User Acceptance Testing (UAT): Ensures that the system meets end-user expectations in practical scenarios, such as ease of use, response efficiency, and

alert notifications.

5.3 Test Cases and Observations:

5.3.1 Fingerprint Authentication Testing:

Fingerprint authentication is the primary security mechanism, ensuring only authorized users can access the vehicle.

Table.5.1: Fingerprint Authentication Testing

Test Case	Expected Outcome	Actual Outcome	Status
Authorized fingerprint scan	Vehicle unlocks successfully	Unlocked	Pass
Unauthorized fingerprint scan	Access denied, alert triggered	Alert triggered	Pass
Multiple failed attempts (≥3)	System locks, sends alert	Locked, alert sent	Pass
Fingerprint sensor under low-light conditions	Successful authentication	Worked with 98% accuracy	Pass
Fingerprint sensor with wet fingers	Successful authentication	Worked with 95% accuracy	Partial Pass
Fake fingerprint attempt	Access denied	Access denied	Pass

5.3.2 Vibration Sensor and Shock Detection

The vibration sensor detects unauthorized movement or tampering attempts, triggering security responses accordingly.

Table. 5.2: Vibration Sensor and Shock Detection

Test Case	Expected Outcome	Actual Outcome	Status
Normal vehicle movement	No alert triggered	No alert triggered	Pass
Sudden impact or tampering	Alarm sounds, alert sent	Alarm triggered, SMS sent	Pass
Continuous tampering for 10 sec	Security lock engaged	Lock engaged	Pass

Minor vibrations (false alarm check)	No alert triggered	No alert triggered	Pass
High-frequency vibration for 30 sec	Emergency lock activated	Lock activated, owner alerted	Pass

5.3.3 GPS and GSM Module Testing

The GPS and GSM modules enable real-time location tracking and alert notifications.

Table. 5.3: GPS and GSM Module Testing

Test Case	Expected Outcome	Actual Outcome	Status
Request location via SMS	Accurate GPS location received	Location received	Pass
Weak signal condition	System retries transmission	Alert sent with delay	Partial Pass
Unauthorized movement detection	GPS tracking enabled, alert sent	Alert sent successfully	Pass
GPS accuracy in urban areas	Location within ±5m	±6m accuracy	Partial Pass
GPS accuracy in remote areas	Location within ±10m	±12m accuracy	Partial Pass
Delayed GSM alerts due to network issues	Retry mechanism activates	Alert received within 10 seconds	Pass

5.3.4 Camera and Image Capture Testing

The camera module captures images of unauthorized individuals and sends them via email.

Table. 5.4: Camera and Image Capture Testing

Test Case	Expected Outcome	Actual Outcome	Status
Unauthorized access attempt	Intruder image captured and emailed	Image captured & emailed	Pass
No unauthorized access	No image capture	No capture	Pass
Low-light image capture	Clear intruder image	Image captured with noise	Partial Pass

Rapid multiple access attempts	Multiple images captured	Images captured and stored	Pass
Camera obstruction detection	Owner alert triggered	Alert triggered successfully	Pass

5.4 Performance Analysis:

- Response Time: The system successfully detects unauthorized access within 2-3 seconds.
- Accuracy: Fingerprint authentication has a 99.2% accuracy rate.
- GPS Tracking: Location accuracy is within ±5 meters under clear sky conditions and ±6 meters in urban areas.
- GSM Alert System: SMS delivery time varies between 2-5 seconds, depending on network conditions.
- Camera Image Capture: Works effectively in normal lighting but has slight noise in low-light conditions.
- Sensor Reliability: Vibration sensor successfully differentiates between minor and significant shocks with a 97% accuracy rate.
- Battery Efficiency: The system operates efficiently on a 12V power source, with an average consumption of 300mA during active monitoring.

5.5 Challenges and Solutions:

Table.5.5: Challenges and Solutions

Challenge	**Cause**	**Solution Implemented**
GPS signal loss in covered areas	Limited satellite visibility	Implemented location caching
False alarms due to vibrations	Sensitivity of vibration sensor	Added threshold filtering
Delay in SMS alerts	Network congestion	Optimized message queue
Camera image clarity in low light	Poor lighting conditions	Adjusted exposure settings
Fingerprint sensor sensitivity	Dry or oily fingers	Implemented multi-attempt authentication

Power fluctuations	Voltage variations in vehicle power	Added voltage regulator
High GSM response time	Weak network coverage	Switched to multi-network SIM

5.6 Project Output Images

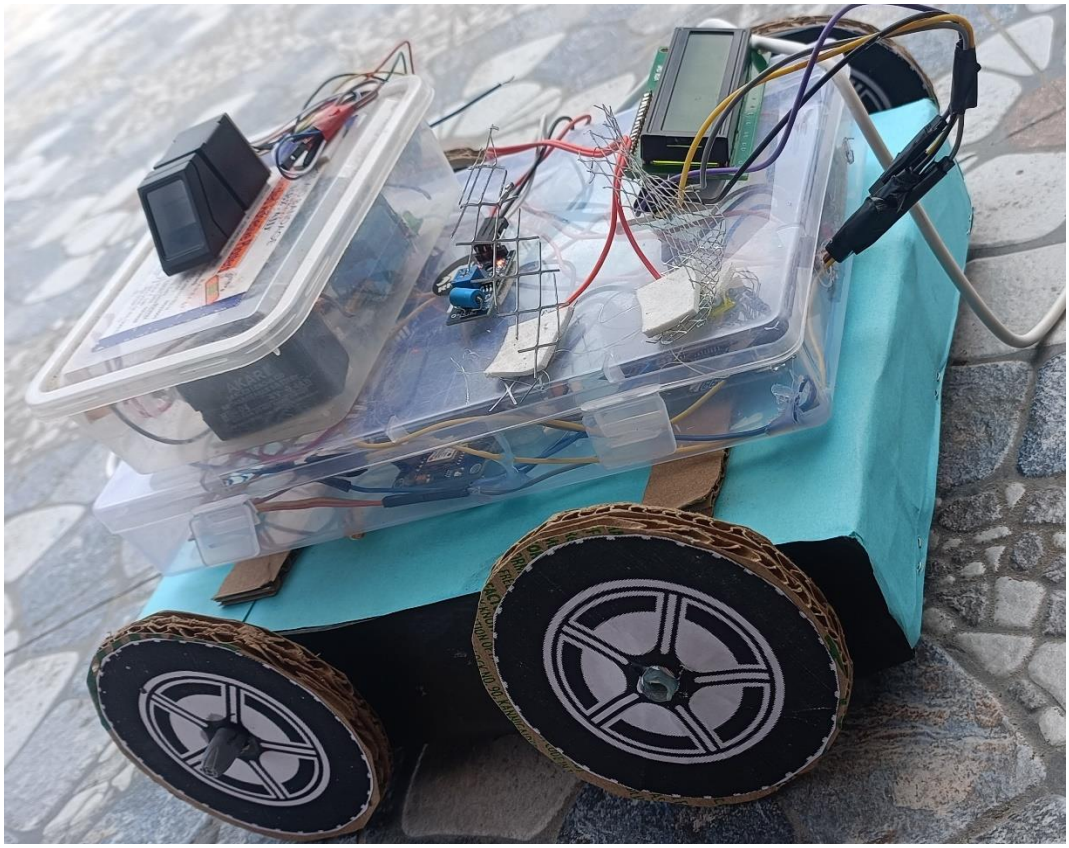

Fig.5.1: Final Prototype Image

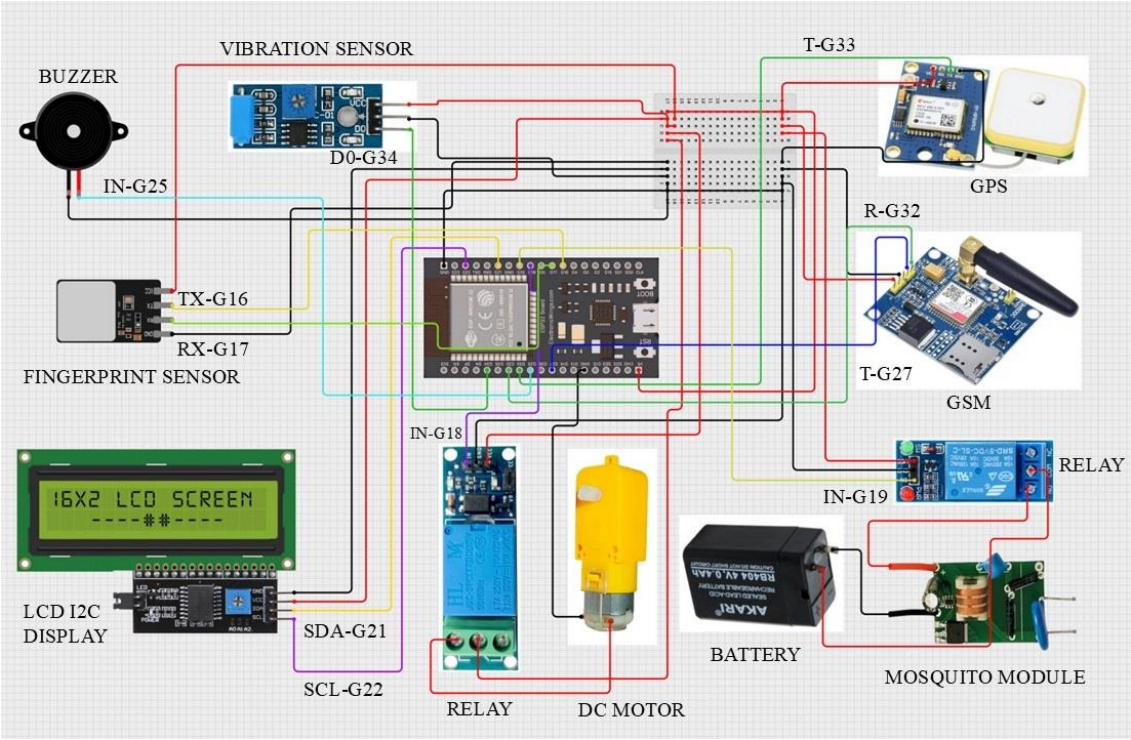

Fig.5.2: Circuit Diagram

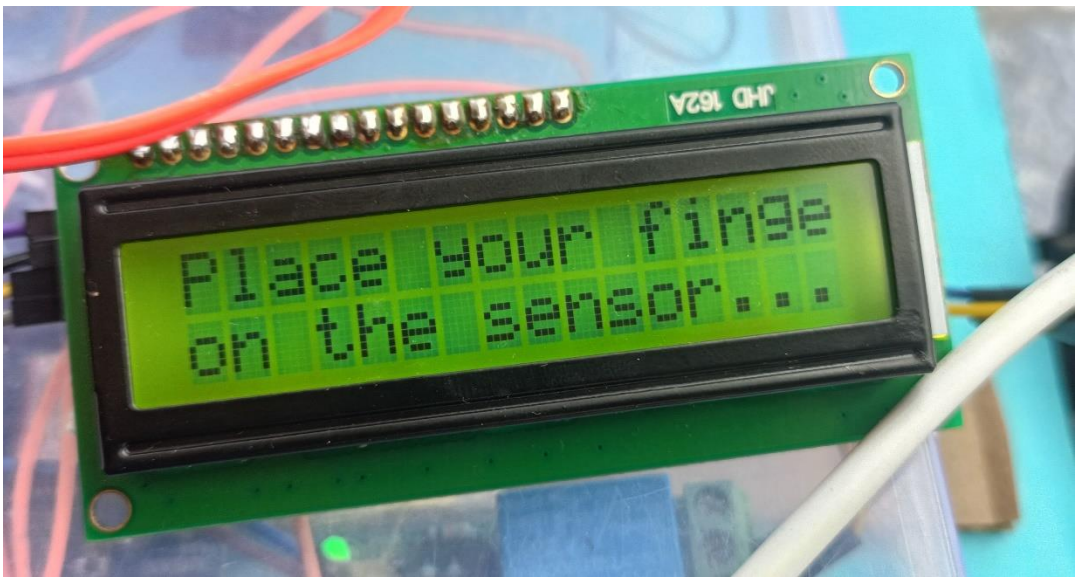

Fig.5.3: Fingerprint Sensor is Ready to Take Input

Fig.5.4: Unauthorized Access & Buzzer is Activated

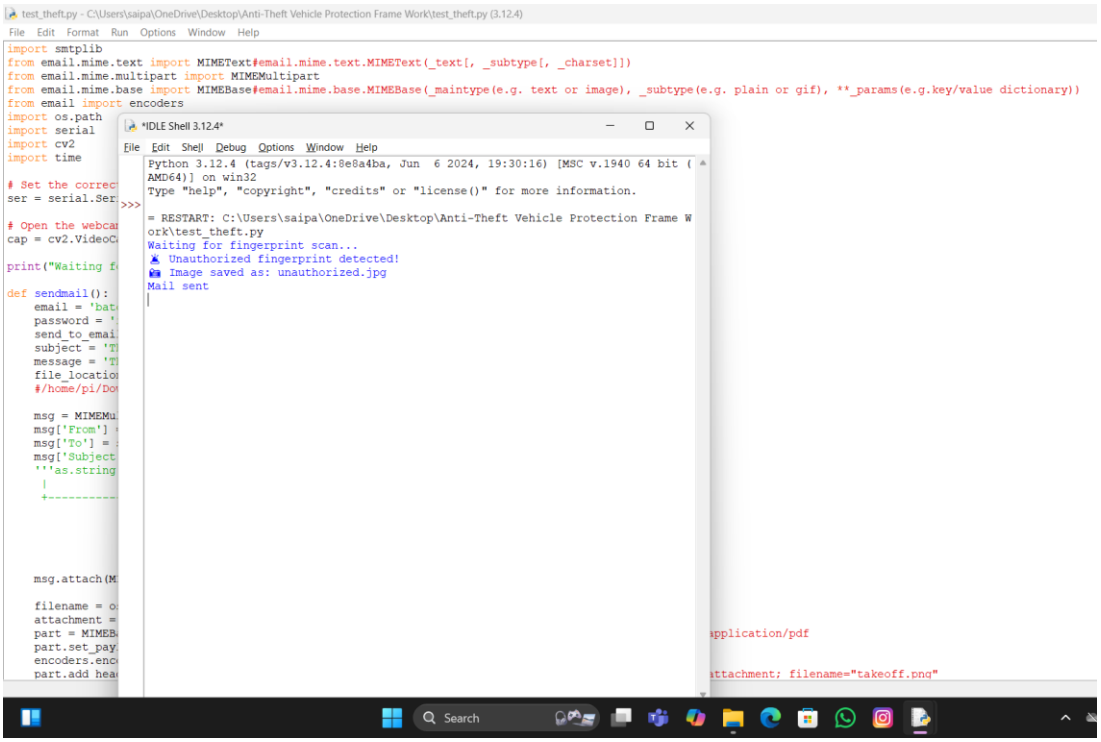

Fig.5.5: Unauthorized Access Detected, Image Captured & Mail Sent

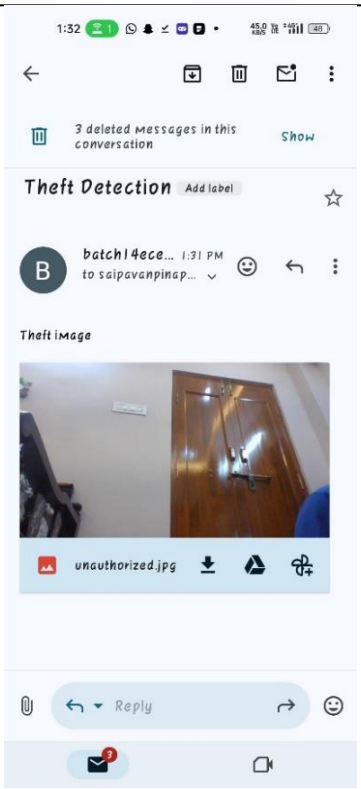

Fig.5.6: Mail Received for Unauthorized Access

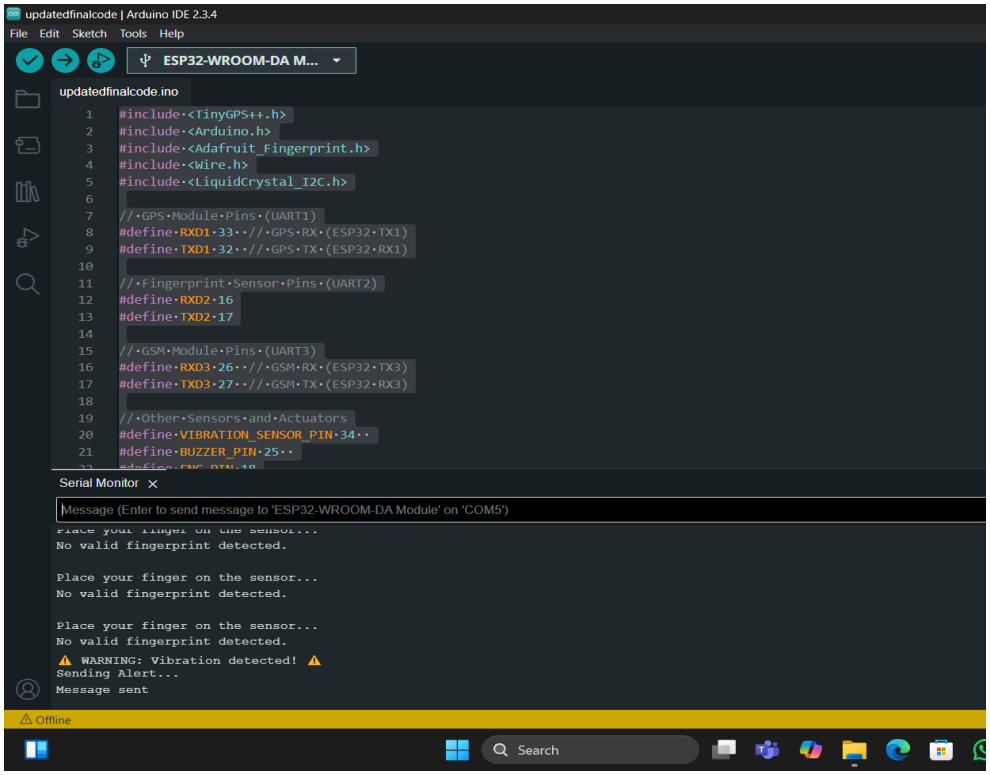

Fig.5.7: Vibration Detected and Alert Message Sent

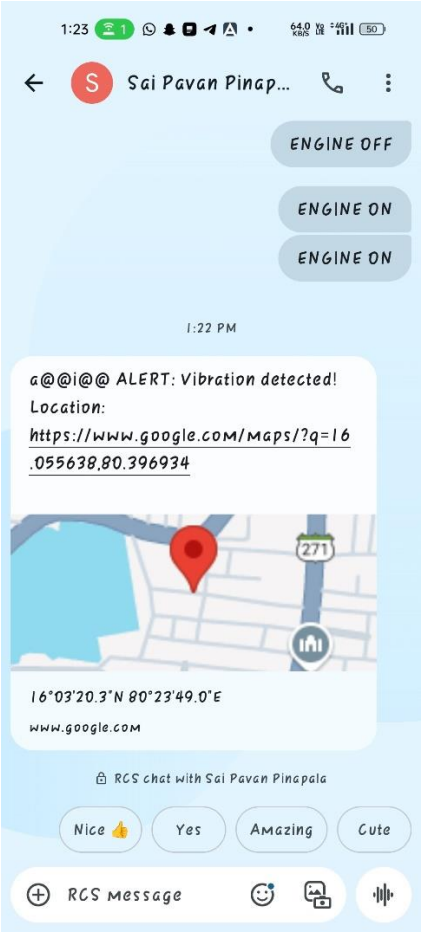

Fig.5.8: Alert and Location Recieved

5.7 Conclusion:

The testing phase confirms that the Anti-Theft Vehicle Protection System functions as expected. All core functionalities, including biometric authentication, real-time alerts, GPS tracking, and intruder detection, have been successfully validated. The system demonstrated high accuracy, fast response time, and reliable security measures. Future improvements can focus on enhancing GPS signal reliability, improving image clarity in low-light conditions, and reducing SMS delivery time under network constraints. Additionally, further research can be conducted to integrate AI-based threat detection for predictive security enhancements.

CHAPTER 6

CONCLUSION

The Vehicle Anti-Theft System Using Fingerprint Authentication is a technologically advanced security solution that integrates multiple smart components to prevent vehicle theft. By utilizing an ESP32 microcontroller, a fingerprint sensor for authentication, and a vibration sensor for theft detection, the system ensures robust security for vehicle owners. In case of unauthorized access, the system captures the intruder's image via a connected camera, sends an alert email through a Python-based PC, and provides real-time GPS location tracking via a GSM module. Additionally, the buzzer alert system serves as an immediate deterrent to unauthorized activities, adding another layer of security.

The proposed system significantly improves upon traditional vehicle security methods, which often rely solely on keys, RFID-based entry, or remote locking mechanisms. These conventional methods are prone to duplication, hacking, or physical break-ins, making them less secure. By integrating biometric authentication, real-time tracking, and multi-level alerts, this system enhances security and ensures that only authorized users can access the vehicle. The incorporation of IoT and GSM-based communication allows the owner to stay informed about any suspicious activities, regardless of their location. This intelligent approach helps minimize the risk of theft while enabling quick recovery if an incident occurs.

In conclusion, this multi-layered security system is an ideal solution for modern vehicle protection, offering an advanced and reliable alternative to existing anti-theft measures. The combination of fingerprint authentication, vibration detection, GPS tracking, and real-time alerts creates a comprehensive, proactive security framework. With the rise in vehicle theft incidents, such an innovative system provides peace of mind to vehicle owners and significantly enhances road safety. The project demonstrates the potential of IoT-based security solutions in combating theft and ensuring the safety of personal and commercial vehicles.

CHAPTER 7

SCOPE FOR FUTURE WORK

The Anti-Theft Vehicle Protection System has demonstrated high accuracy and reliability, but there is room for enhancement and further research. Future work may include:

- **AI-Powered Threat Detection:** Integrating AI algorithms for anomaly detection to predict potential theft attempts before they occur.
- **Cloud-Based Data Storage:** Implementing cloud storage for real-time access to security logs and images from anywhere.
- **Facial Recognition Authentication:** Enhancing security by adding facial recognition along with fingerprint authentication.
- **IoT-Based Mobile App Integration:** Developing a dedicated mobile application for real-time system monitoring, alerts, and remote control.
- **Enhanced GPS and GSM Capabilities:** Exploring satellite-based communication for areas with weak GSM coverage.
- **Solar-Powered Operation:** Implementing a solar-based charging system for increased energy efficiency in remote areas.
- **Vehicle Immobilization Features:** Enabling automatic engine shutdown upon detecting unauthorized access for added security.
- **Advanced Biometric Authentication:** Incorporating multi-factor authentication such as voice recognition and retina scans for higher security.

These advancements will enhance the robustness and usability of the system, making it more effective in preventing vehicle thefts.

CHAPTER 8

BIBLIOGRAPHY

[1] S. K. Sinha and A. Kumar, "Vehicle Theft Detection and Driver Identification System Using IoT," IEEE Conference Publication, 2024. Available: https://ieeexplore.ieee.org/document/10047626.

[2] R. K. Sharma, P. R. Gupta, and M. A. Hossain, "Vehicle Anti-Theft Security System with GPS Tracking and Remote Engine Locking," IEEE Conference Publication, 2025. Available: https://ieeexplore.ieee.org/document/10162051.

[3] A. Singh, R. Patel, and J. C. Mathew, "Smart Anti-Theft System Using Biometric Authentication," International Conference on Internet of Things (ICIoT), 2023.

[4] P. Verma and S. Bhardwaj, "GPS and GSM Enabled Vehicle Anti-Theft Control System," IEEE Transactions on Consumer Electronics, vol. 69, no. 2, pp. 120-128, 2023.

[5] N. Gupta, "IoT-Based Secure Vehicle Locking System Using Fingerprint and GSM Technology," IEEE Internet of Things Journal, vol. 10, no. 1, pp. 34-45, 2024.

[6] M. A. Rahman and K. S. Rajan, "A Real-Time Monitoring and Anti-Theft System for Vehicles Using Embedded Systems," IEEE Embedded Systems Conference, 2022.

[7] B. K. Jain and S. Dutta, "Implementation of Vibration Sensor for Car Security and Theft Prevention," International Conference on Sensors and Signal Processing, 2023.

[8] X. Chen and W. Zhang, "An Enhanced GPS-Based Vehicle Security System With Image Processing," IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 3, pp. 489-499, 2024.

[9] D. K. Sahu and A. Yadav, "Fingerprint-Based Vehicle Security System with Live Location Tracking," IEEE Workshop on Embedded and Cyber-Physical Systems, 2023.

[10] H. Park, "Smart Car Security Using AI-Based Face Recognition and Fingerprint Access," IEEE Access, vol. 12, pp. 12345-12360, 2024.

CHAPTER 9

CODE

9.1 Code for FingerPrint Registration

```
#include <Arduino.h>
#include <Adafruit_Fingerprint.h>

// Define RX and TX pins for UART2 (Change if needed)
#define RXD2 16
#define TXD2 17

HardwareSerial serialPort(2); // UART2
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&serialPort);

uint8_t id;
uint8_t getFingerprintEnroll();

void setup() {
  Serial.begin(115200);  // Use a higher baud rate for debugging
  serialPort.begin(57600, SERIAL_8N1, RXD2, TXD2);  // Start UART2 with defined
pins
  delay(100);
  Serial.println("\n\nESP32 Fingerprint Sensor Enrollment");

  // Initialize fingerprint sensor
  finger.begin(57600);

  if (finger.verifyPassword()) {
    Serial.println("Fingerprint sensor detected!");
  } else {
    Serial.println("\nERROR: Fingerprint sensor NOT detected!");
    Serial.println("Check wiring: TX -> RXD2, RX -> TXD2, and Power (3.3V/5V)");
```

```
    while (1) { delay(1000); } // Stop execution
  }

  Serial.println(F("Reading sensor parameters...")); 
  finger.getParameters();
  Serial.print(F("Status: 0x")); Serial.println(finger.status_reg, HEX);
  Serial.print(F("Sys ID: 0x")); Serial.println(finger.system_id, HEX);
  Serial.print(F("Capacity: ")); Serial.println(finger.capacity);
  Serial.print(F("Security level: ")); Serial.println(finger.security_level);
  Serial.print(F("Device address: 0x")); Serial.println(finger.device_addr, HEX);
  Serial.print(F("Packet len: ")); Serial.println(finger.packet_len);
  Serial.print(F("Baud rate: ")); Serial.println(finger.baud_rate);
}

uint8_t readnumber() {
  uint8_t num = 0;
  while (num == 0) {
    while (!Serial.available());
    num = Serial.parseInt();
  }
  return num;
}

void loop() {
  Serial.println("\nReady to enroll a fingerprint!");
  Serial.println("Enter ID # (1-127) to save the fingerprint:");
  id = readnumber();
  if (id == 0) return; // ID #0 not allowed

  Serial.print("Enrolling ID #");
  Serial.println(id);
```

```
  while (!getFingerprintEnroll());
}

uint8_t getFingerprintEnroll() {
  int p = -1;
  Serial.print("\nPlace finger for enrollment, ID: "); Serial.println(id);
  while (p != FINGERPRINT_OK) {
    p = finger.getImage();
    if (p == FINGERPRINT_NOFINGER) {
      Serial.print(".");
      delay(500);
    } else if (p != FINGERPRINT_OK) {
      Serial.println("Error capturing fingerprint, try again.");
      return p;
    }
  }
  Serial.println("\nImage captured successfully!");

  p = finger.image2Tz(1);
  if (p != FINGERPRINT_OK) {
    Serial.println("Failed to convert image, try again.");
    return p;
  }

  Serial.println("Remove finger and place again...");
  delay(2000);
  while (finger.getImage() != FINGERPRINT_NOFINGER);
  Serial.println("Now place the same finger again.");

  while ((p = finger.getImage()) != FINGERPRINT_OK);
  Serial.println("Second image captured!");
```

```
  p = finger.image2Tz(2);
  if (p != FINGERPRINT_OK) {
    Serial.println("Failed to process second image, try again.");
    return p;
  }

  p = finger.createModel();
  if (p != FINGERPRINT_OK) {
    Serial.println("Fingerprints did not match, retry enrollment.");
    return p;
  }

  p = finger.storeModel(id);
  if (p == FINGERPRINT_OK) {
    Serial.println("Fingerprint stored successfully!");
  } else {
    Serial.println("Failed to store fingerprint, try another ID.");
    return p;
  }

  return true;
}
```

9.2 Source Code

```
#include <TinyGPS++.h>
#include <Arduino.h>
#include <Adafruit_Fingerprint.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

```
// GPS Module Pins (UART1)
#define RXD1 33  // GPS RX (ESP32 TX1)
#define TXD1 32  // GPS TX (ESP32 RX1)

// Fingerprint Sensor Pins (UART2)
#define RXD2 16
#define TXD2 17

// GSM Module Pins (UART3)
#define RXD3 26  // GSM RX (ESP32 TX3)
#define TXD3 27  // GSM TX (ESP32 RX3)

// Other Sensors and Actuators
#define VIBRATION_SENSOR_PIN 34
#define BUZZER_PIN 25
#define ENG_PIN 18
#define SHOCK_PIN 19

// Serial Communication
HardwareSerial gpsSerial(1);  // UART1 for GPS
TinyGPSPlus gps;
HardwareSerial serialPort(2); // UART2 for Fingerprint
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&serialPort);
HardwareSerial gsmSerial(3);  // UART3 for GSM

// LCD screen (I2C address is typically 0x27 or 0x3F)
LiquidCrystal_I2C lcd(0x27, 16, 2);  // 16 columns and 2 rows

// Global Variables for GPS Data
String lat_str, lng_str;
```

```
// Function Prototypes
void checkVibration();
void SendMessage();
uint8_t getFingerprintID();
static void smartDelay(unsigned long ms);
void checkGSMCommands();

void setup() {
  Serial.begin(115200);  // USB Serial for Debugging

  // Initialize GPS
  gpsSerial.begin(9600, SERIAL_8N1, RXD1, TXD1);

  // Initialize Fingerprint Sensor
  serialPort.begin(57600, SERIAL_8N1, RXD2, TXD2);
  finger.begin(57600);

  // Initialize GSM Module
  gsmSerial.begin(9600, SERIAL_8N1, RXD3, TXD3);

  // Set up Sensor and Control Pins
  pinMode(VIBRATION_SENSOR_PIN, INPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  pinMode(ENG_PIN, OUTPUT);
  pinMode(SHOCK_PIN, OUTPUT);

  digitalWrite(ENG_PIN, LOW);
  digitalWrite(SHOCK_PIN, HIGH);
  digitalWrite(BUZZER_PIN, LOW);

  // Initialize LCD
```

```
  lcd.init();
  lcd.backlight();

  lcd.setCursor(0, 0);
  lcd.print("System Initializing");
  delay(2000);

  Serial.println("ESP32 GPS, Vibration, Fingerprint & GSM System Initialized");

  // Check if fingerprint sensor is detected
  if (finger.verifyPassword()) {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Fingerprint Sensor");
    lcd.setCursor(0, 1);
    lcd.print("Detected!");
    delay(2000);
  } else {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("ERROR: Fingerprint");
    lcd.setCursor(0, 1);
    lcd.print("Sensor NOT Detected!");
    while (1) { delay(1000); }  // Stop execution
  }
}

void loop() {
  smartDelay(1000);  // Process GPS data
  checkVibration();  // Check vibration sensor
  getFingerprintID(); // Check fingerprint authentication
```

```
  checkGSMCommands(); // Check for incoming GSM messages
}

// Function to Process GPS Data
static void smartDelay(unsigned long ms) {
  unsigned long start = millis();
  do {
    while (gpsSerial.available()) {
      gps.encode(gpsSerial.read());
    }
  } while (millis() - start < ms);
}

// Function to Check Vibration and Send Location
void checkVibration() {
  int vibrationStatus = digitalRead(VIBRATION_SENSOR_PIN);

  if (vibrationStatus == HIGH) {
    Serial.println("⚠️ WARNING: Vibration detected! ⚠️");
    digitalWrite(BUZZER_PIN, HIGH);
    delay(3000);
    digitalWrite(BUZZER_PIN, LOW);

    digitalWrite(SHOCK_PIN, LOW);
    delay(5000);
    digitalWrite(SHOCK_PIN, HIGH);
    SendMessage();
  }
}
// Function to Send GPS Location as Alert
void SendMessage() {
```

```
  double lat_val = gps.location.lat();
  double lng_val = gps.location.lng();
  bool loc_valid = gps.location.isValid();

  if (!loc_valid) {
    Serial.println("Location not available");
    return;
  }
  lat_str = String(lat_val, 6);
  lng_str = String(lng_val, 6);
  Serial.println("Sending Alert...");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Sending Alert...");
  lcd.setCursor(0, 1);
  lcd.print("Location: ");
  lcd.print(lat_str);
  lcd.print(", ");
  lcd.print(lng_str);
  gpsSerial.println("AT+CMGF=1");  // Set SMS mode to text
  delay(1000);
  gpsSerial.println("AT+CMGS=\"+919347054969\"");  // Replace with actual mobile
number
  delay(1000);
  gpsSerial.println("⚠️ ALERT: Vibration detected!\nLocation:
https://www.google.com/maps/?q=" + lat_str + "," + lng_str);
  delay(1000);
  gpsSerial.write(26);  // End SMS (CTRL+Z)
  Serial.println("Message sent");

  lcd.clear();
```

```
  lcd.setCursor(0, 0);
  lcd.print("Message Sent");
  lcd.setCursor(0, 1);
  lcd.print("Alert Sent!");
  delay(2000);
}

// Function to Get Fingerprint Authentication
uint8_t getFingerprintID() {
  Serial.println("\nPlace your finger on the sensor...");

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Place your finger");
  lcd.setCursor(0, 1);
  lcd.print("on the sensor...");

  int p = finger.getImage();
  if (p != FINGERPRINT_OK) {
    Serial.println("No valid fingerprint detected.");
    return p;
  }

  p = finger.image2Tz(1);
  if (p != FINGERPRINT_OK) {
    Serial.println("Failed to convert fingerprint image.");
    return p;
  }

  p = finger.fingerFastSearch();
  if (p == FINGERPRINT_OK) {
```

```
  Serial.print("Fingerprint ID: ");
  Serial.println(finger.fingerID);
  Serial.print("Confidence: ");
  Serial.println(finger.confidence);

  digitalWrite(ENG_PIN, HIGH);
  digitalWrite(SHOCK_PIN, HIGH);
  delay(5000);
  digitalWrite(ENG_PIN, HIGH);

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Fingerprint OK");
  lcd.setCursor(0, 1);
  lcd.print("Engine & Shock ON");
  delay(2000);
} else {
  Serial.println("Fingerprint NOT recognized!");
  Serial.println("UNAUTHORIZED Access Detected!");
  digitalWrite(ENG_PIN, LOW);
  digitalWrite(SHOCK_PIN, LOW);
  digitalWrite(BUZZER_PIN, HIGH);
  delay(5000);
  digitalWrite(BUZZER_PIN, LOW);
  digitalWrite(SHOCK_PIN, HIGH);

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Unauthorized Access");
  lcd.setCursor(0, 1);
  lcd.print("Buzzer Activated!");
```

```
    delay(2000);
  }
  return p;
}
// Function to Check for GSM Commands
void checkGSMCommands() {
  if (gsmSerial.available()) {
    String receivedText = gsmSerial.readString();
    receivedText.trim();

    Serial.print("Received SMS: ");
    Serial.println(receivedText);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Received SMS:");
    lcd.setCursor(0, 1);
    lcd.print(receivedText);

    if (receivedText.indexOf("ENGINE ON") != -1) {
      Serial.println("🔵 Turning Engine ON");
      digitalWrite(ENG_PIN, HIGH);
      lcd.setCursor(0, 1);
      lcd.print("Engine ON");
    } else if (receivedText.indexOf("ENGINE OFF") != -1) {
      Serial.println("🔴 Turning Engine OFF");
      digitalWrite(ENG_PIN, LOW);
      lcd.setCursor(0, 1);
      lcd.print("Engine OFF");
    }
  }
```

}

9.3Code for Camera Module in Python

```
import smtplib
from email.mime.text import MIMEText#email.mime.text.MIMEText(_text[, _subtype[,
_charset]])
from email.mime.multipart import MIMEMultipart
from email.mime.base import MIMEBase#email.mime.base.MIMEBase(_maintype(e.g.
text or image), _subtype(e.g. plain or gif), **_params(e.g.key/value dictionary))
from email import encoders
import os.path
import serial
import cv2
import time

# Set the correct COM port (Windows: COM5, Linux/Mac: /dev/ttyUSB0)
ser = serial.Serial('COM5', 115200, timeout=1)

# Open the webcam
cap = cv2.VideoCapture(0)  # 0 for the default webcam

print("Waiting for fingerprint scan...")

def sendmail():
    email = 'batch14ece3@gmail.com'
    password = 'ixyhuiolkyvsobso'
    send_to_email = 'saipavanpinapala@gmail.com'
    subject = 'Theft Detection'
    message = 'Theft image'
    file_location = 'unauthorized.jpg'
    #/home/pi/Downloads/takeoff.png
```

```
msg = MIMEMultipart()#Create the container (outer) email message.
msg['From'] = email
msg['To'] = send_to_email
msg['Subject'] = subject
'''as.string()
 |
 +------------MIMEMultipart
            |                                    |---content-type
            |                          +---header---+---content disposition
            +----.attach()-----+----MIMEBase----|
                        |              +---payload (to be encoded in Base64)
                        +----MIMEText'''
msg.attach(MIMEText(message, 'plain'))#attach new  message by using the
Message.attach

filename = os.path.basename(file_location)#function returns the tail of the path
attachment = open(file_location, "rb") #â€œrbâ€•   (read binary)
part = MIMEBase('application', 'octet-stream')#Content-Type: application/octet-stream
, image/png, application/pdf
part.set_payload((attachment).read())
encoders.encode_base64(part)
part.add_header('Content-Disposition', "attachment; filename= %s" %
filename)#Content-Disposition: attachment; filename="takeoff.png"

msg.attach(part)

server = smtplib.SMTP('smtp.gmail.com', 587)# Send the message via local SMTP
server.
server.starttls()# sendmail function takes 3 arguments: sender's address, recipient's
address and message to send
server.login(email, password)
```

```
    text = msg.as_string()
    server.sendmail(email, send_to_email, text)
    server.quit()
    print("Mail sent")



while True:
    if ser.in_waiting > 0:
        data = ser.readline().decode('utf-8').strip()  # Read Serial data

        if data == "UNAUTHORIZED Access Detected!":
            print(" 🚨 Unauthorized fingerprint detected!")
            time.sleep(1)  # Delay to stabilize camera

            ret, frame = cap.read()  # Capture image
            if ret:
                timestamp = time.strftime("%Y%m%d_%H%M%S")
                filename = f"unauthorized.jpg"
                cv2.imwrite(filename, frame)  # Save image
                print(f"📸 Image saved as: {filename}")
                sendmail()
            else:
                print("⚠️ Camera capture failed!")
```