A

MAJOR PROJECT REPORT

ON

ADVANCED SECURITY SYSTEM WITH MULTI LEVEL AUTHENTICATION USING RASPBERRY PI PICO

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

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

SUBMITTED BY

K. SRIKANTH	**218R1A04M7**
K. PAVAN KUMAR	**218R1A04M8**
L.NICHITHA	**218R1A04M9**
L.DIVYA	**218R1A04N0**

Under the Esteemed Guidance of

Mr. K. SUBRAMANYA CHARI
Assistant Professor.

CMR
ENGINEERING COLLEGE
EXPLORE TO INVENT

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CMR ENGINEERING COLLEGE

UGC AUTONOMOUS

(Approved by AICTE, Affiliated to JNTU Hyderabad, Accredited by NBA)
Kandlakoya(V), Medchal(M), Telangana – 501401

(2024-2025)

CMR ENGINEERING COLLEGE

UGC AUTONOMOUS

(Approved by AICTE, Affiliated to JNTU Hyderabad, Accredited by NBA)

Kandlakoya (V), Medchal Road, Hyderabad - 501 401

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that Major project work entitled **"ADVANCED SECURITY SYSTEM WITH MULTI LEVEL AUTHENTICATION USING RASPBERRY PI"** is being Submitted by **K. SRIKANTH** bearing **Roll** No: **218R1A04M7, K. PAVAN KUMAR** bearing Roll No**: 218R1A04M8, L. NICHITHA** bearing Roll No: **218R1A04M9, L. DIVYA** bearing Roll No: **218R1A04N0** in B.Tech IV-II semester, Electronics and Communication Engineering is a record Bonafide work carried out by then during the academic year 2024-25. The results embodied in this report have not been submitted to any other University for the award of any degree.

INTERNAL GUIDE:
MR. K. SUBRAMANYA CHARI
Assistant Professor.

HEAD OF THE DEPARTMENT
Dr. SUMAN MISHRA
Professor & HOD.

EXTERNAL EXAMINER

ACKNOWLEDGEMENTS

We sincerely thank the management of our college CMR Engineering College for providing required facilities during our project work. We derive great pleasure in expressing our sincere gratitude to our Principal **Dr. A. S. Reddy** for his timely suggestions, which helped us to complete the project work successfully. It is the very auspicious moment we would like to express our gratitude to **Dr. SUMAN MISHRA**, Head of the Department, ECE for his consistent encouragement during the progress of this project.

We take it as a privilege to thank our project coordinator **Dr. T. SATYANARAYANA**, Associate Professor, Department of ECE for the ideas that led to complete the project work and we also thank him for his continuous guidance, support and unfailing patience, throughout the course of this work. We sincerely thank our project internal guide **Mr. K. SUBRAMANYA CHARI,** Assistant Professor of ECE for guidance and encouragement in carrying out this project work.

DECLARATION

We hereby declare that the Major project entitled **"ADVANCED SECURITY SYSTEM WITH MULTI LEVEL AUTHENTICATION USING RASPBERRY PI"** is the work done by us in campus at **CMR ENGINEERING COLLEGE,** Kandlakoya during the academic year 2024-2025 and is submitted as major project in partial fulfillment of the requirements for the award of degree of **BACHELOR OF TECHNOLOGY in ELECTRONICS AND COMMUNICATION ENGINEERING FRO JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD.**

K. SRIKANTH	**(218R1A04M7)**
K. PAVAN KUMAR	**(218R1A04M8)**
L. NICHITHA	**(218R1A04M9)**
L.DIVYA	**(218R1A04N0)**

ABSTRACT

The abstract of an Advanced security system with multilevel authentication using project focuses on developing a comprehensive security system for banks to protect their assets and ensure the safety of customers' funds. This project involves implementing various security measures, including surveillance cameras, access control systems, biometric authentication, and alarm systems. Security system can be minimized the risk of unauthorized access, theft, and fraud.

By incorporating advanced technologies and protocols, this project aims to enhance the overall security of the bank and provide peace of mind to both the bank and its customers. Since last few years, security systems are getting more awareness and importance. Advanced security system with multilevel authentication using is a system for validating, monitoring and controlling the security at bank locker rooms. Today, there are many banks using authorize access control approach to prevent the locker room from unauthorized access.

In this paper highly reliable, multilevel and most efficient locker room security system has been designed. The system includes a biometric system, i.e. a fingerprint scanner and an iris scanner, camera, which are responsible for the security of the main door of the locker room and the system also includes a system to provide access of the locker room area to only authorize people. To monitor the unauthorized people in the locker room area a passive infrared sensor is fixed. In case of any unauthorized motion the picture from the camera will be mailed to security officials and the alarms will be onto inform the local security. The system proposed in this paper is a better security system in terms of number of level of security.

The system proposed in this paper is a better security system in terms of number of level of security. The main goal of this project is to design and implement a bank locker security system based on Finger print and OTP technology. This can be organized in bank, offices and homes. In this system only the authenticate person recover the documents or money from the lockers. In this security system fingerprint and OTP is used. In this system first person enroll user name and password and mobile number. If user name and password matches then Finger of person will detect and store with ID.

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The "Advanced security system with multilevel authentication using raspberry pi" project is designed to create a robust and sophisticated security solution for areas requiring stringent access control, such as homes, offices, or high-security facilities. The system is built around an Arduino microcontroller, which serves as the central hub, integrating multiple security layers to ensure that only authorized individuals can gain entry.

The biometric layer uses a fingerprint sensor to scan and verify fingerprints, allowing access only to registered users. The protection layer adds an additional level of security by requiring a correct face id. The system incorporates three key components: biometric authentication, password protection, camera Even if the fingerprint is recognized, access will not be granted without the correct face id. The final layer employs RFID technology, where each user is provided with an RFID tag that must be scanned to complete the authentication process. Access is granted only when all three security checks are successfully passed. The project highlights the practicality of using Arduino for developing advanced security systems, offering a cost-effective, reliable, and flexible solution that can be tailored to meet diverse security need.

1.1 OVERVIEW OF THE PROJECT

OBJECTIVE OF THE PROJECT

The primary aim of this project is to design and implement advanced security system with multi-level authentication using raspberry pi Pico that ensures the highest level of protection for bank assets and customer funds. By integrating advanced technologies such as biometric authentication, RFID access control, and real-time surveillance camera, the system seeks to prevent unauthorized access, reduce the risk of theft and fraud, and enhance overall security within bank locker rooms.

The project aspires to create a comprehensive security solution that not only safeguards physical assets but also provides peace of mind to both the banking institution and its customers. The biometric layer uses a fingerprint sensor to scan and verify fingerprints, allowing access only to registered users. The protection layer adds an additional level of security by requiring a correct face id The image shows a Raspberry Pi.

1.2 OBJECTIVE OF THE PROJECT

1. **Develop a Multi-Layered Security Framework**: To design a security system that integrates various layers of protection, including biometric verification (fingerprint and iris scanners), RFID technology for controlled access, and surveillance measures to monitor and record activities within the locker room.
2. **Implement Biometric Authentication:** To incorporate advanced camera biometric methods such as fingerprint and iris scanners, to authenticate individuals seeking access to secure areas. This will ensure that only authorized personnel can gain entry.
3. **Integrate RFID Technology**: To deploy RFID access control systems that further restrict access to authorized users, ensuring that only those with proper credentials can enter sensitive areas.
4. **Establish Real-Time Surveillance and Monitoring**: To set up passive infrared sensors and surveillance cameras that detect unauthorized movement and capture real- time images. These images will be used to trigger alarms and send notifications to security personnel in case of any suspicious activity.
5. **Incorporate OTP Verification**: To integrate One-Time Password (OTP) technology for an additional layer of security. Authorized users will receive a unique code on their mobile devices, which must be entered to access the locker.
6. **Create a Comprehensive Logging System**: To develop a logging mechanism that tracks and records user activities, including check-ins and check-outs, along with essential user information, to enhance accountability and facilitate auditing.

1.3 ORGANIZATION OF THE PROJECT

The "advanced security system with multilevel authentication using raspberry pi" project is organized to systematically guide the development of a comprehensive security solution. The project begins with an Introduction that provides background information on the importance of security systems and the need for a multilevel approach. The objective of the project is clearly defined as the design and implementation of a security system using Arduino, with a focus on environments such as homes, offices, and sensitive installations.

The next section is the Literature Review, which is the examines existing security technologies, including biometric systems, face id-based systems, and RFID technology. This review identifies the limitations in current solutions and highlights the gaps that the proposed project aims to address. Suggestions for future work are provided, including potential improvements like wireless connectivity, remote monitoring, or additional.

In the System Design section, the project presents the architecture of the security system through block diagrams or flowcharts, illustrating how the Arduino microcontroller coordinates with the biometric sensor, keypad, RFID module, and other components. The selection of components is detailed, along with their technical specifications. The software design is also explained, covering the logic for fingerprint matching, face id, and RFID tag reading, along with an overview of the code structure.

The Implementation phase focuses on the practical aspects of the project, including the hardware setup and software development. This section describes the physical assembly of components, wiring connections, and integration with the raspberry pi board. The process of developing the Arduino code is discussed, along with the testing and calibration of each security layer to ensure accurate and reliable performance. In the Results and Analysis section, the project presents the outcomes of the system testing, including the accuracy of fingerprint recognition, face id verification, and RFID tag reading. The performance of the system is evaluated based on response time, reliability, and overall limitations of the current system are also discussed.

The project concludes with a Conclusion that summarizes the key achievements, such as the successful integration of multiple security layers. Suggestions for future work are provided, including potential improvements like wireless connectivity, remote monitoring, or additional security features. Finally, the project includes References and Appendices. The references list all the sources used in the literature review and component selection, while the appendices contain the full Arduino code, circuit diagrams, and datasheets for the key components. This organized approach ensures that all aspects of the project are thoroughly covered, leading to a well- documented and successful project outcome.

CHAPTER 2

LITERATURE SURVEY

Early research, such as work by Nand Kishore and S. K. Sharma (2008), emphasized the use of Raspberry pi pico for multi-level authentication, demonstrating basic GSM integration for authority access.

In 2008, Nand Kishore and S. K. Sharma made significant contributions to the field of multi-level authentication by focusing on the integration of raspberry pi pico, GSM (Global System for Mobile Communications) technology. Their research demonstrated the potential of using embedded systems for real-time multi-level authentication. They employed PIC and raspberry for their simplicity and effectiveness in interfacing with external modules, including GSM, to authority access. The GSM module provided the real-time coordinates, while the GSM module enabled the transmission of this data to a remote server or directly to the mobile phone. This integration allowed for basic authority access and improved security, as the system could alert owners about the making it easier to respond in case of theft or unauthorized movement.

2.1 EXISTING SYSTEM

Existing security systems typically rely on one or two layers of protection to secure areas, such as homes, offices, and sensitive installations. These systems often include either biometric authentication, camera, password-based entry, or RFID technology, but rarely do they integrate all three into a single cohesive system**.** Biometric Authentication Systems are widely used in high-security environments due to their ability to uniquely identify individuals based on physical characteristics, such as fingerprints or facial recognition. These systems offer a high level of security since biometric data is difficult to replicate or steal. However, they are not foolproof; issues such as sensor malfunctions or the inability to recognize authorized users under certain conditions can compromise their effectiveness. Camera face id-based Systems are another common approach, particularly in environments where cost and simplicity are priorities.

These systems require users to input a PIN or password to gain access. While easy to implement, password-based& camera systems have notable vulnerabilities. Passwords can be easily guessed, stolen, or shared, making them less secure, especially when used as the sole method of authentication. Camera face id systems are vulnerable to security breaches due to weak or compromised passwords. RFID systems, while convenient, are susceptible

to cloning and do not offer a not foolproof method of ensuring that only authorized users gain access. These limitations highlight the need for a more robust and integrated approach to security. By combining biometric authentication, password protection, and RFID technology, a multilevel security system can overcome the individual weaknesses of each component Janani.

The integration of these technologies into a single system offers a more secure, reliable, and comprehensive solution, addressing the gaps present in existing systems. This project seeks to fill this need by developing advanced security system with multilevel authentication using raspberry pi, providing an enhanced level of protection that is greater than the sum of its parts. Platforms like Arduino and Raspberry Pi have enabled the development of customizable security systems tailored to specific needs. This multilevel approach enhances security, reduces vulnerabilities, and provides a more comprehensive solution, making it an attractive option for a wide range of applications.

FIG:2.1 Examples for Security Systems

These limitations highlight the need for a more robust and integrated approach to security. By combining biometric authentication, password protection, and RFID technology, a multilevel security system can overcome the individual weaknesses of each component Janani.

The project begins with an Introduction that provides background information on the importance of security systems and the need for a multilevel approach. The objective of the project is clearly defined as the design and implementation of a security system using Arduino. With a focus on environments such as homes, offices, and sensitive installations. illustrating how the Arduino microcontroller coordinates with the biometric sensor, keypad, RFID module, and other components illustrating how the Arduino microcontroller coordinates with the biometric sensor, keypad, RFID module, and other components illustrating how the Arduino microcontroller coordinates with the biometric sensor, keypad, RFID module, and other components illustrating how the Arduino microcontroller coordinates with the biometric sensor, keypad, RFID module, and other components.

2.2 PROPOSED SYSTEMS

1. Traditional Physical Security Measures

Features:

Manual Locks and Keys: Standard practice involves physical locks and keys for securing bank locker rooms and vaults.

Security Guards: Presence of security personnel for surveillance and immediate response.

Basic Surveillance Cameras: Analog CCTV cameras to monitor and record footage.

Advantages:

Simplicity: Easy to understand and implement.

Cost-Effective: Initial setup cost is relatively low compared to advanced system.

Limitations:

Vulnerability to Theft: Keys can be lost, stolen, or duplicated, leading to potential security branches.

Limited Access Control: Difficulty in managing and tracking multiple users with physical.

Reactive Security: Security is largely reactive rather than proactive, with limited capability to detect or prevent unauthorized access before it occurs.

Electronic Access Control Systems

Features:

Card-Based Access: Use of magnetic stripe cards or proximity cards for entry.

Electronic Locks: Locks that are controlled electronically and the can be managed remotely.

Biometric Authentication Systems

Features:

Fingerprint Scanners: Use of fingerprint recognition to authenticate users. Iris Scanners: Iris recognition systems for secure access.

Advanced Authentication: Often combined with PINs or passwords for enhanced security.

Advantages:

High Security: Biometric data is unique to individuals, significantly reducing the risk of unauthorized access.

No Physical Keys or Cards: Eliminates the risk associated with lost or stolen physical access credentials.

Limitations:

High cost: implementation of advanced biometric system can be expensive.

False Acceptance/Rejection Rates: Biometric systems can sometimes have issues with false. positives or negatives, impacting reliability.

Privacy Concerns: Collection and storage of biometric data raise privacy issues and require robust data protection measures.

RFID-Based Access Control Systems

Features:

RFID Tags/Cards: Use of radio frequency identification tags or cards for access. Reader Devices: RFID readers to authenticate tags and grant access.

Integration with Other Systems: Often integrated with electronic locks and alarm systems.

Advantages:

Convenient Access: Quick and easy access without the need for physical contact.

Scalable: Easy to manage and scale for large numbers of users.

Limitations:

Security Risks: RFID tags can be intercepted or cloned, posing potential security risks.

Battery Dependence: Some RFID systems require batteries for the tags or readers, which can affect reliability if not properly maintained.

Integrated Security Systems

Features:

Combination of Technologies: Integration of surveillance cameras, biometric authentication, RFID access control, and alarm systems.

Centralized Management: Unified control and monitoring through a central system.

These DIY system wireless indoor security cameras are installed to provide.

Advantages:

High Security: Biometric data is unique to individuals, significantly reducing the risk of unauthorized access.

No Physical Keys or Cards: Eliminates the risk associated with lost or stolen physical access credentials.

Limitations:

High cost: implementation of advanced biometric system can be expensive.

False Acceptance/Rejection Rates: Biometric systems can sometimes have issues with false. positives or negatives, impacting reliability.

Privacy Concerns: Collection and storage of biometric data raise privacy issues and require robust data protection measures.

RFID-Based Access Control Systems

Features:

RFID Tags/Cards: Use of radio frequency identification tags or cards for access. Reader Devices: RFID readers to authenticate tags and grant access.

Integration with Other Systems: Often integrated with electronic locks and alarm systems.

Advantages:

Convenient Access: Quick and easy access without the need for physical contact. Scalable: Easy to manage and scale for large numbers of users.

Limitations:

Security Risks: RFID tags can be intercepted or cloned, posing potential security risks.

Battery Dependence: Some RFID systems require batteries for the tags or readers, which can affect reliability if not properly maintained.

Integrated Security Systems

Features:

Combination of Technologies: Integration of surveillance cameras, biometric authentication, RFID access control, and alarm systems.

Centralized Management: Unified control and monitoring through a central system

Real-Time Alerts: Automated alerts and notifications for security breaches or unauthorized access. The Raspberry Pi Pico reads the entered OTP and compares it with the one sent to the mobile device. If the OTP matches, authentication is successful, and access is granted The Raspberry Pi Pico reads the entered OTP and compares it with the one sent to the mobile.

Advantages:

Comprehensive Security: Multi-layered approach provides enhanced protection against various security threats.

Enhanced Monitoring: Centralized management allows for real-time monitoring and quick response to security incidents.

Customizable Solutions: Can be tailored to specific security need environments.

Limitations:

Complexity: Integration of multiple technologies can be complex and require specialized knowledge for implementation and maintenance.

Higher Costs: Comprehensive systems typically involve higher initial investment and ongoing maintenance costs.

Existing security systems for banks vary widely in terms of technology and effectiveness. Traditional methods offer simplicity but lack advanced features, while modern electronic, biometric, and RFID systems provide higher levels of security but come with their own set of challenges. Integrated security systems represent a comprehensive approach, combining various technologies to enhance overall protection, though they also involve higher costs.

FIG: 2.2 Example For Multilevel Security

The Raspberry Pi Pico reads the entered OTP and compares it with the one sent to the mobile device. If the OTP matches, authentication is successful, and access is granted The systems can be programmable or with fixed functionality. Industrial machines, consumer.

The image represents a smart home security system, showcasing the various interconnected devices that enhance home protection. At the centre of the system is a wireless security door lock, which provides access control, likely using a keypad, fingerprint scanner, or mobile app for authentication. Surrounding the house, there are wireless outdoor security cameras positioned strategically to monitor the premises and detect any unauthorized activity.

Inside the home, wireless indoor security cameras are installed to provide surveillance in different rooms, ensuring that any movement or unusual activity is recorded. A wireless motion sensor detects movement in specific areas and can trigger an alert or alarm if unexpected activity occurs. Additionally, a wireless glass break sensor is included, which can detect the sound of breaking glass, adding an extra layer of security against forced entry.

The home keypad serves as a central control panel for the security system, allowing users to arm or disarm the system easily. A mobile app provides remote access, enabling homeowners to monitor their property in real time, receive alerts, and control security devices from anywhere. This integration of multiple security components ensures comprehensive home protection with the convenience of wireless connectivity and smart technology.

2.3 EMBEDDED INTRODUCTION

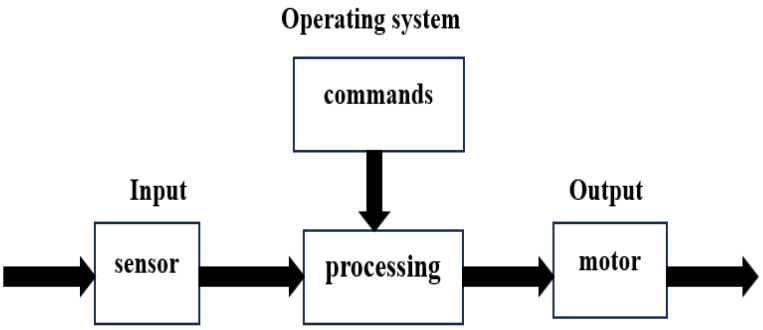


FIG: 2.3 Embedded OS

An embedded system is a combination of computer hardware and software designed for a specific function or functions within a larger system. The systems can be programmable or with fixed functionality.

Industrial machines, consumer electronics, agricultural and process industry devices, automobiles, medical equipment, cameras, household appliances. While embedded systems are computing systems, they can range from having no user interface (UI) for example, on devices in which the system is designed to perform a single task to complex graphical user interfaces (GUIs), such as in mobile devices. User interface scan include buttons, LEDs and touchscreen sensing. Some systems use remote user interfaces as well.

History of embedded systems

Embedded systems date back to the 1960s. Charles Stark Draper developed an integrated circuit (IC) in 1961 to reduce the size and weight of the Apollo Guidance Computer, the digital system installed on the Apollo Command Module and Lunar Module. The first computer to use ICs, it helped astronauts collect real-time flight data.

In 1965, Autonoetic, now a part of Boeing, developed the D-17B, the computer used in the Minuteman I missile guidance system. It is widely recognized as the first mass-produced embedded system. When the Minuteman II went into production in 1966, the D-17B was replaced with the NS-17 missile guidance system, known for its high-volume use of integrated circuits. In 1968, the first embedded system for a vehicle was released; the Volkswagen 1600 used a microprocessor to control its electronic fuel injection system.

On devices in which the system is designed to perform a single task to complex graphical user Also, in 1971, Intel released what is widely recognized as the first commercially available processor, the 4004. The 4-bit microprocessor was designed for use in calculators and small electronics, though it required eternal memory and support chips. The 8-bit Intel 8008, released in 1972, had 16 KB of memory; the Intel 8080 followed in 1974 with 64 KB of memory. The 8080's successor, x86 series, was released in 1978 and is still largely in use today. In 1987, the first embedded operating system, the real-time VxWorks, was released by Wind River, followed by Microsoft's Windows Embedded.

CHAPTER 3

HARDWARE REQUIREMENTS

3.1 HARDWARE

Embedded system hardware

Embedded system hardware can be microprocessor- or microcontroller-based. In either case, an integrated circuit is at the heart of the product that is generally designed to carry out the real-time computing. Microprocessors are visually indistinguishable from microcontrollers. However, the microprocessor only implements a central processing unit (CPU) and, thus, requires the addition of other components such as memory chips. Conversely, microcontrollers are designed as self-contained systems.

Microcontrollers include not only a CPU, but also memory and peripherals such as flash memory, RAM or serial communication ports. Because microcontrollers tend to implement full (if relatively low computer power) systems, they are frequently used on more complex tasks. For example, microcontrollers are used in the operations of vehicles, robots, medical devices and home appliances. At the higher end of microcontroller capability, the term System on a chip (SOC) is often used, although there's no exact delineation in terms of RAM, clock speed, power consumption and so on. It is one of the characteristics of embedded and cyber-physical systems that both.

Hardware for embedded systems is much less standardized than hardware for personal computers. Due to the huge variety of embedded system hardware, it is impossible to provide a comprehensive overview of all types of hardware components. Nevertheless, we will try to provide a survey of some of the essential components which can be found in most systems. The choice of components for the WHO-recommended hand rub formulations takes into account cost constraints and microbicidal activity. The following two formulations are recommended for local production with a maximum of 50 liters per lot to ensure safety in production and storage.

The systems can be programmable or with fixed functionality. Industrial machines, consumer electronics, agricultural and process industry devices, automobiles, medical equipment, cameras, household appliances. The systems can be programmable or with fixed functionality. Industrial machines, consumer electronics, agricultural and process industry devices, automobiles.

Markets and Markets, a business to business (B2B) research firm, predicts that the embedded market will be worth $116.2 billion by 2025. Chip manufacturers for embedded systems include many well-known technology companies, such as Apple, IBM, Intel and Texas Instruments, as well as numerous other companies less familiar to those outside the field.

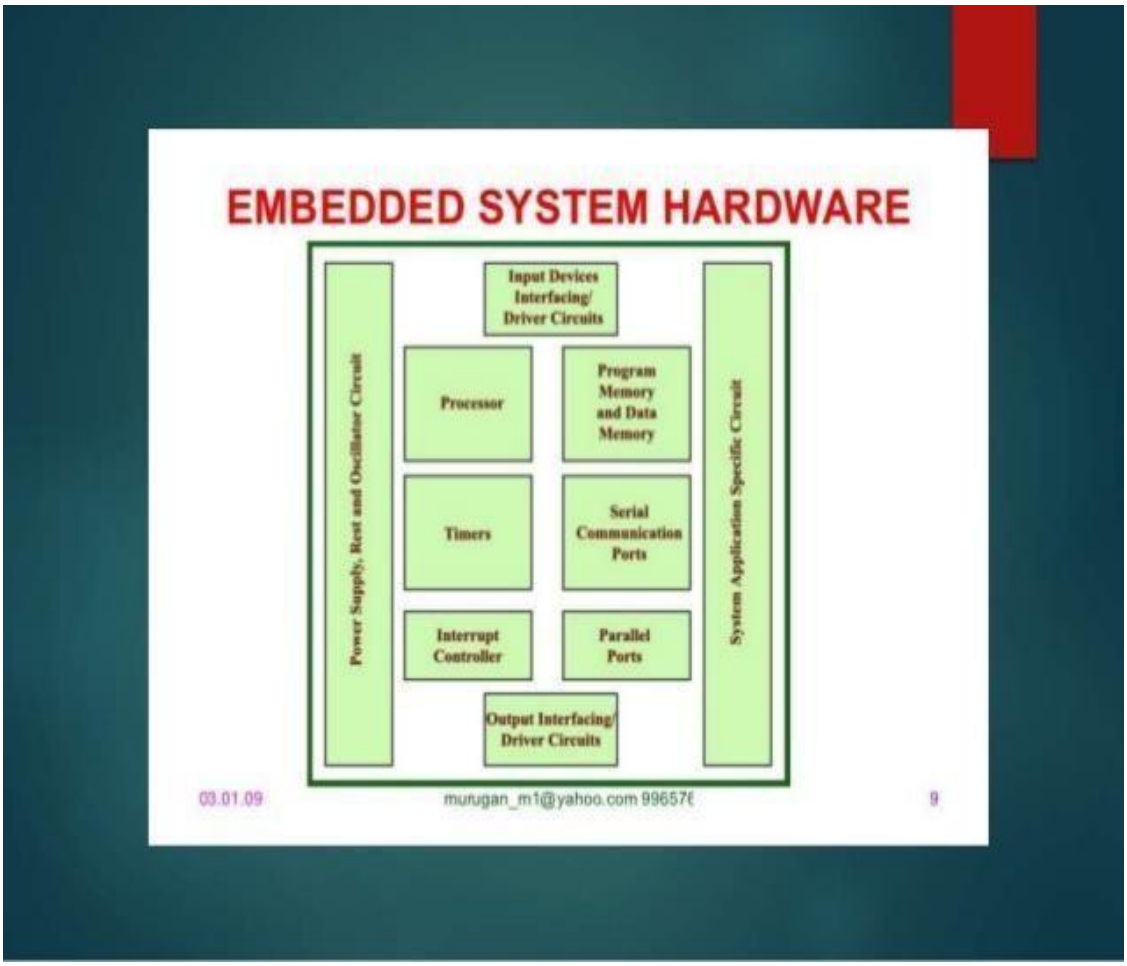


FIG: 3.1 Embedded Systems Hardware Block Diagram

The expected growth is partially due to the continued investment in artificial intelligence (AI), mobile computing and the need for chips designed for that high-level processing. To be used efficiently, all computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. A second meaning of the term of system requirements, is a the generalization of this first definition.

The image represents the hardware architecture of an embedded system, outlining its key components and their interactions. At the core of the system is the processor, which executes instructions and performs computations. The processor is supported by program memory and data memory, which store the embedded program and runtime data, respectively.

Timers are included in the architecture to manage time-sensitive operations and scheduling tasks. An interrupt controller handles external and internal events that require immediate attention, ensuring efficient real-time processing. Serial communication ports facilitate data exchange with external devices, while parallel ports allow for multiple bits of data to be transmitted simultaneously. Input device interfacing and driver circuits enable the system to interact with various input devices, such as sensors or keypads, while output interfacing and driver circuits manage the connection to output components, such as displays or actuators.

The system also includes application-specific circuits tailored to meet the unique requirements of the embedded system's intended function. These circuits optimize performance for specialized tasks, making the embedded system more efficient and reliable. Often manufacturers of games will provide the consumer with a set of requirements that are different from those that are needed to run a software.

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, A hardware requirements list is often accompanied by a hardware compatibility list(HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application.

Serial communication ports facilitate data exchange with external devices, while parallel ports allow for multiple bits of data to be transmitted simultaneously. Input device interfacing and driver circuits enable the system to interact with various input devices, such as sensors or keypads, while output interfacing and driver circuits manage the connection to output components, such as displays or actuators. A power supply, reset, and oscillator circuit ensures stable operation by providing the necessary voltage, system resets, and clock signals. Input device interfacing and driver circuits enable the system to interact with various input devices, such as sensors or keypads, Input device interfacing and driver circuits enable the system to interact with various input.

Architecture

All computer operating systems are designed for a particular computer architecture. Most software applications are limited to particular operating systems running on particular architectures. Although architecture-independent operating systems and applications exist, most need to be recompiled to run on a new architecture. See also a list of common operating systems and their supporting architectures.

Processing power

The power of the central processing unit (CPU) is a fundamental system requirement for any software. Most software running onx86 architecture define processing power as the model and the clock speed of the CPU. Many other features of a CPU that influence its speed and power, like bus speed, cache, and MIPS are often ignored. This definition of power is often erroneous, as AMD Athlon and Intel Pentium CPUs at similar clock speed often have different throughput speeds.

Intel Pentium CPUs have enjoyed a considerable degree of popularity, and are often mentioned in this category. They are frequently used on more complex tasks. For example, microcontrollers are used in the operations of vehicles, robots, medical devices and home appliances. At the higher end of microcontroller capability, the term System on a chip (SOC) is often used, although there's no exact delineation in terms of RAM, clock speed.

The architecture of an embedded system consists of various hardware components that work together to perform specific functions efficiently. At the core of the system is the processor, which executes program instructions and controls all operations. It can be a microcontroller or a microprocessor, depending on the complexity of the application. The processor interacts with memory components, which include program memory such as ROM, Flash, or EEPROM, where firmware is stored permanently, and data memory like RAM, which holds temporary variables and execution data.

Timers play an essential role in managing scheduled operations, delays, and controlling time-dependent processes such as signal generation. Alongside timers, the interrupt controller allows the system to handle multiple tasks efficiently by responding to hardware or software events without constantly checking for updates, thereby improving responsiveness and power efficiency. For communication with external components, the system uses serial communication ports for data transmission and reception through protocols like UART, SPI, or I2C. Additionally, parallel ports enable fast data transfer.

Memory

All software, when run, resides in the random access memory(RAM) of a computer. Memory requirements are defined after considering demands of the application, operating system, supporting software and files, and other running processes. Optimal performance of other unrelated software running on a multi-tasking computer system is also considered when defining this requirement.

Secondary storage

Data storage device requirements vary, depending on the size of software installation, temporary files created and maintained while installing or running the software, and possible use of swap space(if RAM is insufficient).

Display adapter

Software requiring a better than average computer graphics display, like graphics edit or sand high-endgames, often define high-end display adapter sin the system requirements.

Peripherals

Some software applications need to make extensive and/or special use of some peripherals, demanding the higher performance or functionality of such peripherals. Such peripherals include CD-ROM drives, keyboards, pointing devices, network devices, etc.

Basic Structure of an Embedded System

The following illustration shows the basic structure of an embedded system.

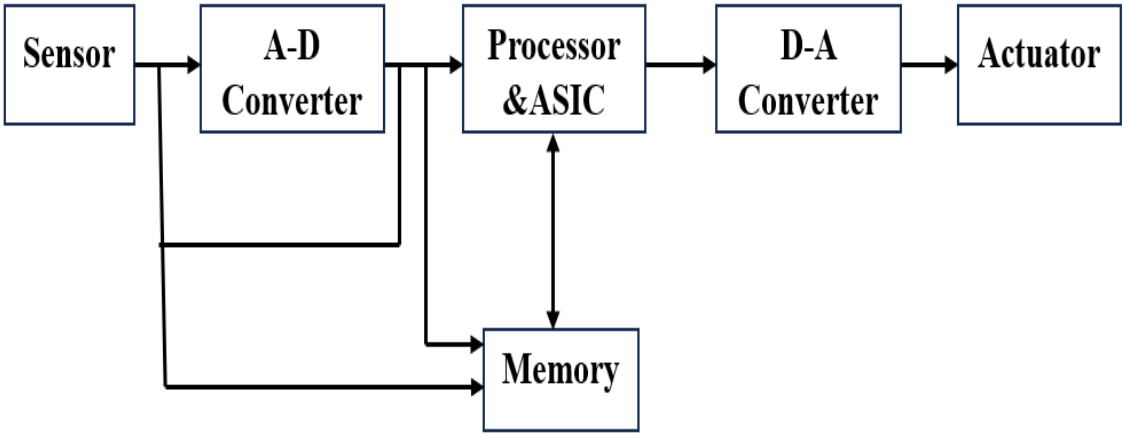


Fig: 3.2 Peripherals Of Embedded Systems

Input device interfacing and driver circuits enable the system to interact with various input devices, such as sensors or keypads, Input device interfacing and driver circuits enable the system.

3.2 INTRODUCTION TO RASPBERRY PI PICO

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has14 digital input/output pins (of which 6 can be used as PWM outputs),6 analogue inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. Arduino boards can read inputs like a finger on a button, light on a sensor, or a Twitter message, and turn them into outputs like turning on an LED, activating a motor, or publishing something online.

It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online**.**

Fig: 3.3 Raspberry pi pico

When you run a “standard” C/C++ program, you have to write a “main” function. This main function will be called first, and from there, you will call other functions and execute the functionalities of your program. In Arduino, there is no main function.

The Raspberry Pi Pico W is a microcontroller board based on the RP2040 chip, and this diagram helps users understand how to interface with its various input/output options. you will call other functions and execute the functionalities of your program. In Arduino, there is no main function. The pinout diagram is color-coded to differentiate between the different functions of each pin

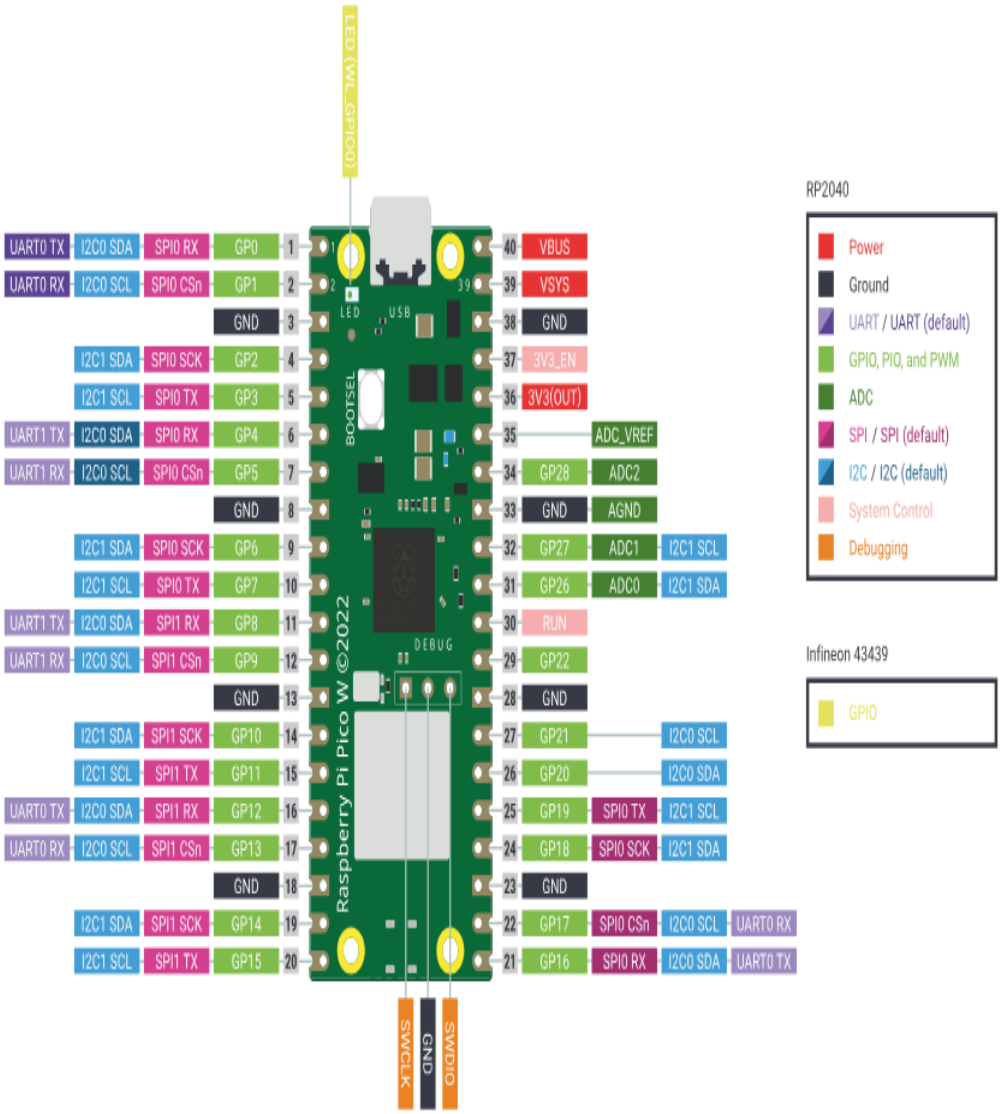


Fig: 3.4 Raspberry pi pico Pin Description

This image is a Raspberry Pi Pico W pinout diagram, which provides a detailed overview of the different pins and their functionalities. The colours help users easily identify categories such that as power, ground, communication protocols (UART, I2C, SPI), GPIO (General Purpose Input/Output), PWM (Pulse Width Modulation), ADC (Analog-to-Digital Converter), and system control pins.

Here is a detailed explanation of the Raspberry Pi Pico W pinout with points:

1. Power Pins (Red)

- VBUS (Pin 40): Provides 5V power when connected via USB.
- VSYS (Pin 39): Main system power input, can take 1.8V to 5.5V.
- 3V3(OUT) (Pin 36): Supplies a regulated 3.3V output for external components.
- 3V3_EN (Pin 37): Enables or disables the 3.3V power supply.
- GND (Multiple Pins): Ground connection for the circuit.

2. GPIO (General Purpose Input/Output) Pins (Green)

- GPIO Pins (GP0 - GP28): Can be used as digital input/output.
- Supports PWM (Pulse Width Modulation) for applications like motor control and LED dimming.

3. Communication Protocols

UART (Universal Asynchronous Receiver-Transmitter) – Blue

- UART0 TX (GP0), UART0 RX (GP1): Used for serial communication.
- UART1 TX (GP8), UART1 RX (GP9): Second UART channel.

I2C (Inter-Integrated Circuit) – Light Blue

- I2C0 SDA (GP0), I2C0 SCL (GP1): First I2C channel for sensor communication.
- I2C1 SDA (GP2), I2C1 SCL (GP3): Second I2C channel for additional peripherals.

SPI (Serial Peripheral Interface) – Pink

- SPI0: TX (GP3), RX (GP4), SCK (GP2), CSn (GP5): SPI communication with peripherals.
- SPI1: TX (GP15), RX (GP12), SCK (GP14), CSn (GP13): Second SPI channel.

4. ADC (Analog-to-Digital Converter) Pins (Green)

- ADC0 (GP26), ADC1 (GP27), ADC2 (GP28): Used for reading analog signals.
- ADC_VREF (Pin 35): Provides reference voltage for ADC.

5. System Control Pins (Orange)

- RUN (Pin 30): Used to reset the microcontroller.

6. Debugging Pins (Orange)

- SWCLK & SWDIO (Bottom Right): Used for debugging the microcontroller.

7. Infineon 43439 (Wi-Fi Module) – Yellow

- Special GPIO pin used for controlling the onboard Wi-Fi module.

The UART (Universal Asynchronous Receiver-Transmitter) pins are marked in blue and are used for serial communication. The Pico W has two UART channels (UART0 and UART1), each with dedicated TX (transmit) and RX (receive) pins.

The I2C (Inter-Integrated Circuit) pins are the marked in light blue and provide communication capabilities for interfacing with sensors and other I2C-compatible devices. The Pico W has two I2C buses (I2C0 and I2C1), each with dedicated SDA (data) and SCL (clock) lines.

The SPI (Serial Peripheral Interface) pins are marked in pink and allow high-speed communication with SPI-compatible peripherals like displays and sensors. The Pico W supports SPI0 and SPI1, with dedicated SCK (clock), TX (MOSI - Master Out Slave In), RX (MISO - Master In Slave Out), and CSn (Chip Select) pins. The ADC (Analog-to-Digital Converter) pins are marked in green and allow the Pico W to read analog signals from sensors. The board has three ADC channels (ADC0, ADC1, and ADC2), as well.

3.3 INTRODUCTION TO SERVO MOTOR

A servo motor is a type of motor that allows for precise control of angular position, velocity, and acceleration. It consists of a motor coupled to a sensor for position feedback. Servos are controlled by sending a pulse-width modulation (PWM) signal to the motor, which determines the desired position.

FIG: 3.5 Servo Motor Interfacing

A servo motor is a type of motor that allows for precise control of angular position, velocity, and acceleration. It consists of a motor coupled to a sensor for position feedback.

Key Components:

Motor: Can be either a DC or AC motor.

Control Circuit: Usually, an electronic circuit that interprets the PWM signal.

Position Sensor: Often a potentiometer or encoder that provides feedback to the control circuit about the motor's position.

The image shows a Raspberry Pi Pico connected to a servo motor. The red wire is connected to the power pin, which can be either 3.3V or 5V, depending on the servo's voltage requirement. The black or brown wire is connected to the GND pin, completing the circuit. The orange or yellow wire is connected to a GPIO pin on the Raspberry Pi Pico, which sends control signals to the servo.

A servo motor is a type of motor that allows for precise control of angular position, velocity, and acceleration. It consists of a motor coupled to a sensor for position feedback. Servos are controlled by sending a pulse-width modulation (PWM) signal to the motor, which determines the desired position

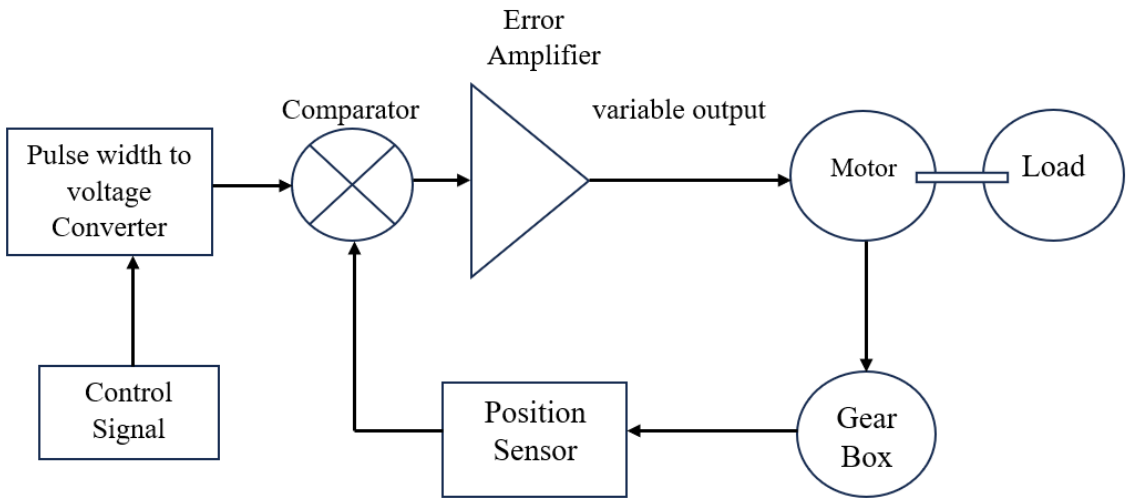


FIG 3.6 Block Diagram of Servo Motor

Applications:

The Raspberry Pi Pico generates PWM signals through the GPIO pin, which determines the position of the servo. A higher or lower pulse width moves the servo to different angles. The power and ground connections ensure proper operation. A simple Arduino IDE code can be used to control the servo by attaching it to a GPIO pin.

3.4 INTRODUCTION TO FINGER PRINT SENSOR

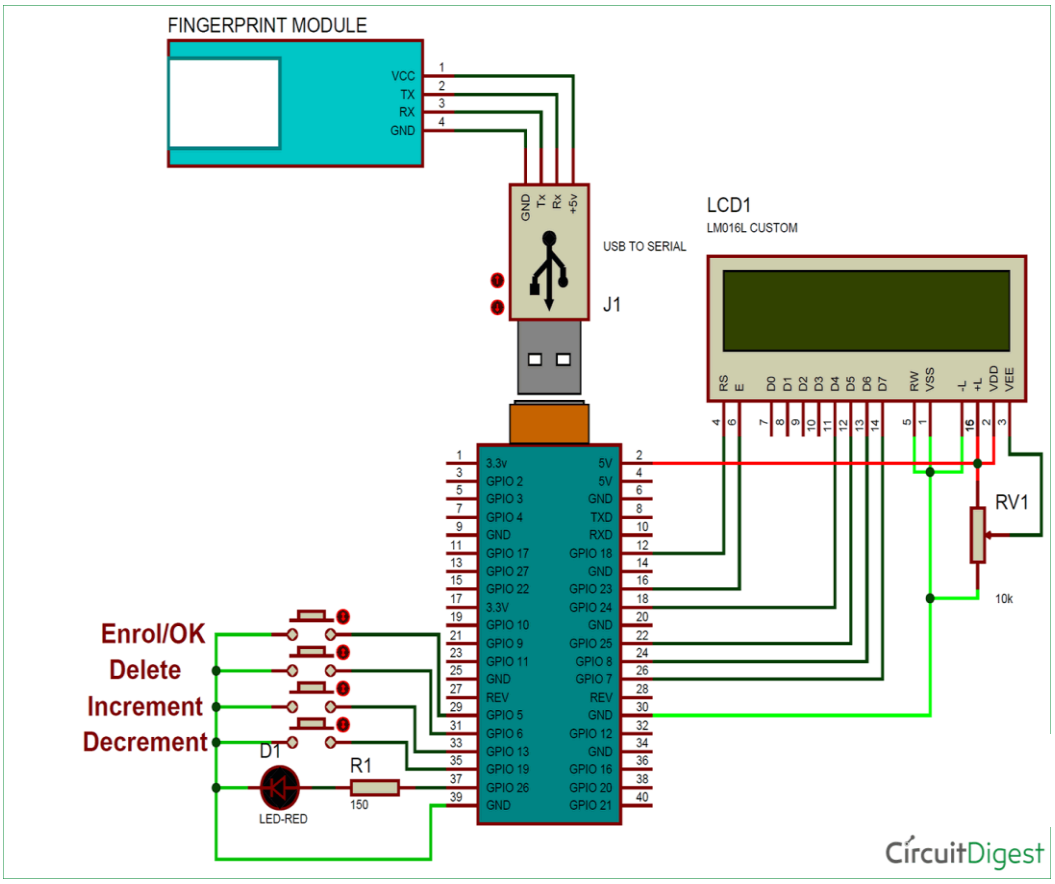
R307 is a finger print sensor module with TTL UART interface. The user can store the finger print data in the module and can configure it in 1:1 or 1: N mode for identifying the person. The FP module can directly interface with 3v3 Microcontroller. A level converter (like MAX232) is required for interfacing with PC.

The R307 fingerprint module has two interface TTL UART and USB2.0, USB2.0 interface can be connected to the computer; RS232 interface is a TTL level, the default baud rate is 57600, can be changed, refer to a communication protocol; can and microcontroller, such as ARM, DSP and other serial devices with a connection, 3.3V 5V microcontroller can be connected directly. Needs to connect the computer level conversion, level conversion note, embodiments such as a MAX232 circuit.

FIG: 3.7 R307 Fingerprint Sensor

No two people have the same fingerprints, not even identical twins. Fingerprints don't change with age unless the deep layer is destroyed or intentionally changed. fingerprint, impression made by the papillary ridges on the ends of the fingers and thumbs. Fingerprints afford an infallible means of personal identification, because the ridge arrangement on every finger of every human being is unique and does not alter with growth or age. Fingerprint sensors are used in many devices, including laptops, smartphones, and tablets. They are also used in security systems, such as those used to access buildings and computer systems RS232 interface is a TTL level, the default baud. Fingerprint sensors are used in many devices, then including laptops, smartphones, and tablets. They are also used in security systems, such as those used to access buildings and computer.

The circuit diagram illustrates a fingerprint-based system using a microcontroller, an LCD display, a fingerprint sensor module, and a set of control buttons. The fingerprint module is connected to the microcontroller through the TX and RX pins, enabling serial communication. A USB-to-serial converter facilitates data transfer between the microcontroller and a computer for programming and debugging. An LCD display is connected to the microcontroller, showing system information and user prompts.



Fig

3.8 Block Diagram Of Fingerprint

A 10k potentiometer is included to adjust the contrast of the LCD. Several push buttons allow user interaction, such as enrolling fingerprints, deleting records, and navigating options. An LED with a resistor is included as an indicator. The microcontroller manages all components, processing fingerprint data and displaying messages on the LCD. It interacts with the fingerprint module to enroll, store, and match fingerprints.

The system is powered through the microcontroller's power pins, with connections to both 3.3V and 5V sources, depending on the components. The ground connections complete the circuit. The circuit diagram illustrates a fingerprint-based system using a microcontroller, an LCD display, a fingerprint sensor module, and a set of control buttons. The fingerprint module is connected to the microcontroller through the TX and RX pins.

3.5 INTRODUCTION TO CAMERA

DESCRIPTION:

Optical device for recording or transmitting photographic images or videos. A camera is a device that takes pictures (photographs). It uses film or electronics to make a picture of something. It is a tool of photography. A lens makes the image that the film or electronics "sees".

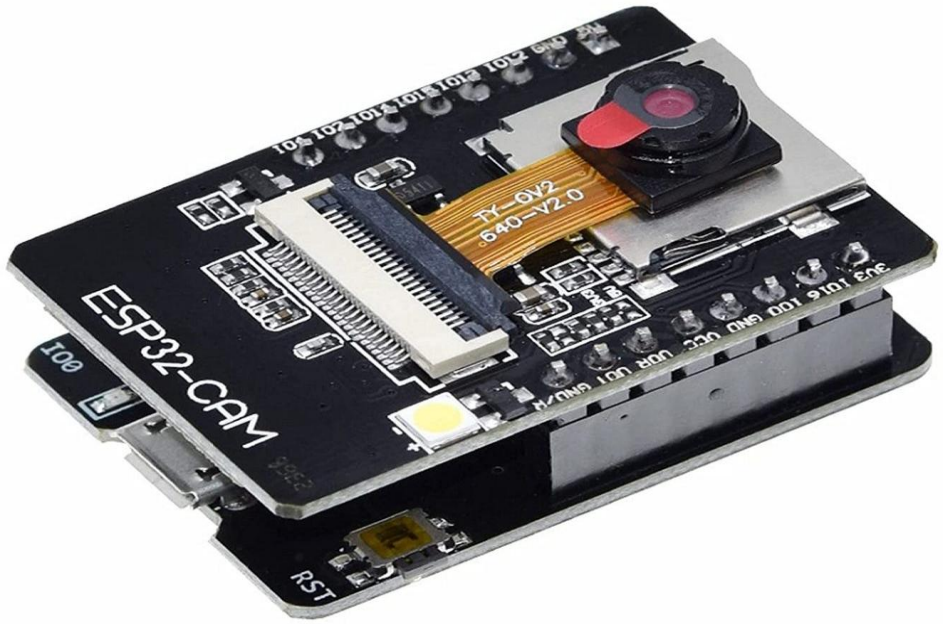


FIG: 3.9 Raspberry Pi Camera Module

Key Features of the Camera:

Image Sensor

Resolution (Megapixels): The sensor captures the image in pixels. Higher megapixel counts generally provide more detail, though other factors also important.

Sensor Size: Larger sensors (e.g., full-frame or APS-C) usually offer better image quality, especially in low light.

Sensor Type: CCD (Charged Coupled Device) and CMOS (Complementary Metal-Oxide-Semiconductor) are two main sensor types. CMOS is more common in modern cameras.

Autofocus (AF) System

Autofocus Points: The number and distribution of AF points determine how precisely the camera can focus on subjects.

Focus Modes: Common modes include Single AF (for stationary subjects) and Continuous AF (for moving subjects). Some cameras also feature Eye Detection AF for portrait photography.

ISO Sensitivity

ISO Range: ISO determines the sensor's sensitivity to light. Higher ISO settings allow for better low-light performance but can introduce noise (graininess).

Noise Reduction: Modern cameras often include noise reduction features that help mitigate the effects of high ISO settings.

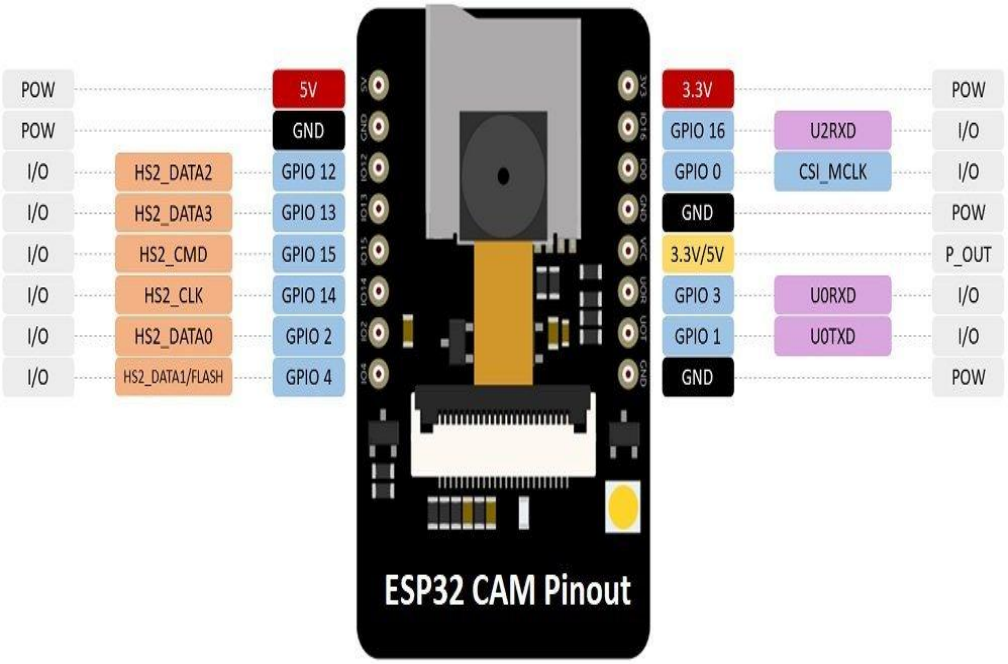


Fig 3.10 Esp32-Cam Pinout Diagram

Viewfinder & Display

Optical Viewfinder (OVF): Found in DSLR cameras, it gives a direct optical view of the scene.

Electronic Viewfinder (EVF): Found in mirrorless cameras, it provides a digital view of the scene, often with helpful overlays (e.g., histograms, focus peaking).

LCD Screen: A screen on the back of the camera used to frame shots, change settings, and review photos. Some cameras feature a touch-enabled or articulating screen for flexibility.

Image Stabilization (IS)

Optical Image Stabilization (OIS): A feature in some lenses that reduces camera shake by adjusting the lens elements.

Video Recording Capabilities

Resolution: Many modern cameras offer 4K video recording, while some high-end models support 6K or even 8K.

Frame Rate: Common frame rates include 24fps (cinematic), 30fps (standard), and 60fps (smooth motion). Some cameras support higher frame rates for slow-motion effects.

Audio Input: Many cameras have an external microphone input for better sound quality during video recording.

The ESP32-CAM pinout diagram provides a detailed representation of the various pins available on the module and their functions. The board is powered using either a 3.3V or 5V power supply, with dedicated ground (GND) pins for proper electrical grounding. It features multiple general-purpose input/output (GPIO) pins, which can be used for interfacing with external components such as sensors, actuators, or communication modules.

The left side of the diagram highlights several high-speed (HS2) data lines, which are primarily used for SD card interfacing and flash memory operations. These include HS2_DATA0 to HS2_DATA3, HS2_CMD, and HS2_CLK, which handle data transfer, command execution, and clock synchronization. Additionally, GPIO pins such as GPIO 2 and GPIO 4 are available for custom applications. On the right side, there are UART communication pins labelled U0RXD and U0TXD, which are used for serial communication with external devices like microcontrollers or computers. Another set, U2RXD, is an additional UART receive pin. The CSI_MCLK pin is dedicated to providing a master clock signal for the camera interface. The module is also equipped with a power.

Optical device for recording or transmitting photographic images or videos. A camera is a device that takes pictures (photographs). It uses film or electronics to make a picture of something. It is a tool of photography. A lens makes the image that the film or electronics "sees". The ESP32-CAM pinout diagram provides a detailed representation of the various pins available on the module and their functions. The board is powered using either a 3.3V or 5V power supply, with dedicated ground (GND) pins for proper electrical grounding. It features multiple general-purpose input/output (GPIO) pins, which can be used for interfacing with external components such as sensors, actuators, or communication modules. On the right side, there are UART communication pins labelled U0RXD and U0TXD, which are used for serial communication with external devices.

Connectivity Features

Wi-Fi/Bluetooth: Enables wireless transfer of images to smartphones, computers, or cloud storage, and can allow remote control of the camera.

GPS: Some cameras have built-in GPS to tag photos with location data.

USB-C / HDMI Ports: Used for transferring data, charging, or outputting video to external displays.

Creative Modes & Filters

Manual Mode (M): Full control over exposure settings.

Program Mode (P): Camera selects aperture and shutter speed, but the user can adjust other settings.

Scene Modes: Pre-set modes for specific environments, such as portrait, landscape, night, or macro.

Filters & Effects: Digital effects or in-camera processing options to alter the look of photos, such as black and white, sepia, or vignette.

RAW vs JPEG

RAW Files: Capture all the sensor data, allowing for maximum flexibility in post-processing, but require more storage and processing time.

3.6 INTRODUCTION TO SIM 800L MODULE DESCRIPTION:

The image shows a SIM800L GSM module, which is used for cellular communication in embedded systems. This module allows devices to send and receive SMS, make calls, and access mobile data over a GSM network. The module has a small form factor with a SIM Com chipset and an IMEI number printed on it for network identification. A helical antenna is included, which improves signal reception for better communication. The module has multiple pin headers for easy interfacing with microcontrollers like Arduino, ESP32, or Raspberry Pi. It operates on a voltage range of 3.7V to 4.2V, making it essential to use a proper power source to avoid instability.

The SIM800L communicates via UART, allowing serial communication with the microcontroller. It can be controlled using AT commands, which enable functions like sending text messages, making calls, or connecting to the internet. The small capacitor and other onboard components ensure stable operation by regulating power and filtering signals. This module is widely used in IoT applications, GPS tracking, and remote monitoring systems due to its reliability and low power consumption.

Key Features of The SIM800L GSM module:

GSM Communication**:** The SIM800L module provides GSM communication capabilities, allowing you to connect to mobile networks for data transmission and voice communication.

The SMA antenna connector is used to attach an external antenna to the modem. This is particularly useful in situations where you need to improve signal reception.

SIM Card Slot: The module typically includes a SIM card slot where you can insert a GSM SIM card. The SIM card is necessary for authenticating your device on the mobile network**.**

Serial Communication: The SIM800L communicates with a microcontroller or computer using serial communication (UART). You send AT commands to the module to control its data

FIG:3.11 Sim 800L GSM Module

SMA Antenna:

Voltage Input: The module usually requires a DC power supply voltage within a specified range often around 3.4v to 4,5v, to operate properly in the sim 800L GSM module.

Digital and Analog I/O Pins: Some versions of the module may provide digital and analog devices.

SMS and Call Functions: You can use the SIM800L to send and receive SMS messages and make or receive phone calls. It can be programmed to perform specific actions based on commands.

Data Transmission: The module supports data transmission over GPRS (General Packet Radio Service) and can be used for internet connectivity in addition to SMS and voice communication.

Applications:

Remote Monitoring: Monitoring and controlling remote equipment and systems via SMS OR data communications.

IoT Projects: Integrating GSM comm into IoT devices for remote data collection and control.

Security Systems: Sending SMS alerts in security and alarm systems.

Telemetry and Data Logging: Collecting and transmitting data from remote sensors and loggers.

Vehicle Tracking: Tracking the location and status of vehicles and assets.

Home Automation: Implementing SMS-based control and monitoring of home automation systems.

Industrial Automation: Integrating GSM communication into industrial control and automation processes.

Prototyping and Experiments: Using the module for prototyping and experimentation in electronics and communications projects.

The SIM800L GSM modem is a communication device that allows you to establish a GSM (Global System for Mobile Communications) connection to send and receive data, including text messages and calls, over the cellular network. It is commonly used in various applications, including remote monitoring, IoT (Internet of Things) projects, and SMS-based systems.

Technical Details:

- Single supply voltage: 3.4V – 4.5V
- Power saving mode: Typical power consumption in SLEEP mode is 1.5mA
- Frequency bands: SIM900A Dual-band: EGSM900, DCS1800. The SIM900A can search the two frequency bands automatically. The frequency bands also can be set by AT command.
- GPRS connectivity: GPRS multi-slot class 10 (default), GPRS multi-slot class 8 (option)

connection to send and receive data, including text messages and calls, over the cellular.

- Transmitting power: Class 4 (2W) at EGSM 900, Class 1 (1W) at DCS 1800
- Operating Temperature: -30ºC to +80ºC**.**

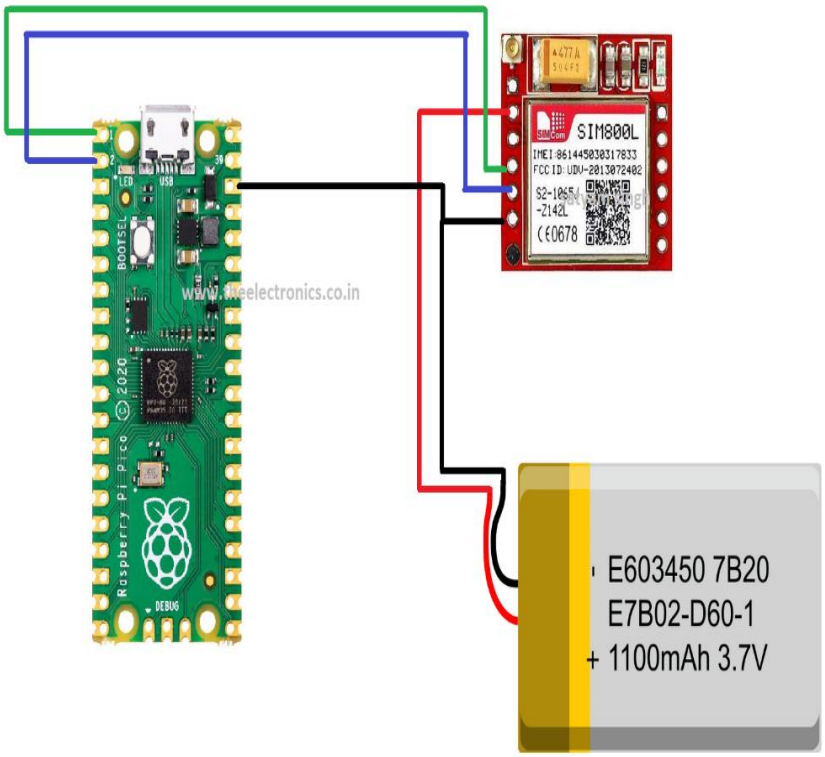


FIG :3.12 GSM With Raspberry pi PICO

Adjusting Sensitivity:

Use the potentiometer on the module to adjust the sensitivity. Turning it clockwise or counterclockwise will increase or decrease sensitivity. In an advanced security system with multi-level authentication, speech recognition plays a vital role in verifying a person's identity based on their voice. This adds an extra layer of protection alongside PIN entry and fingerprint scanning. The module captures audio input through a microphone, processes the speech, and converts it into a digital format.

It then analyzes the unique features of the voice, such as tone, pitch, and frequency, to match it with pre-stored voice patterns. If the authentication is successful, the system grants access or executes a specific action, such as unlocking a door or triggering alarm. By recognizing voice patterns, the module allows users to interact with devices effortlessly, enhancing the authentication of the user input both convenience and security. The given diagram represents the architecture of a communication module, which integrates multiple interfaces to facilitate wireless communication.

The memory is responsible for storing essential data, firmware, and operational instructions required for processing communication tasks. The baseband engine acts as the control unit, handling signal processing and managing communication protocols, ensuring smooth interaction between different components. The radio frequency section is responsible for transmitting and receiving signals over the air, enabling wireless connectivity.

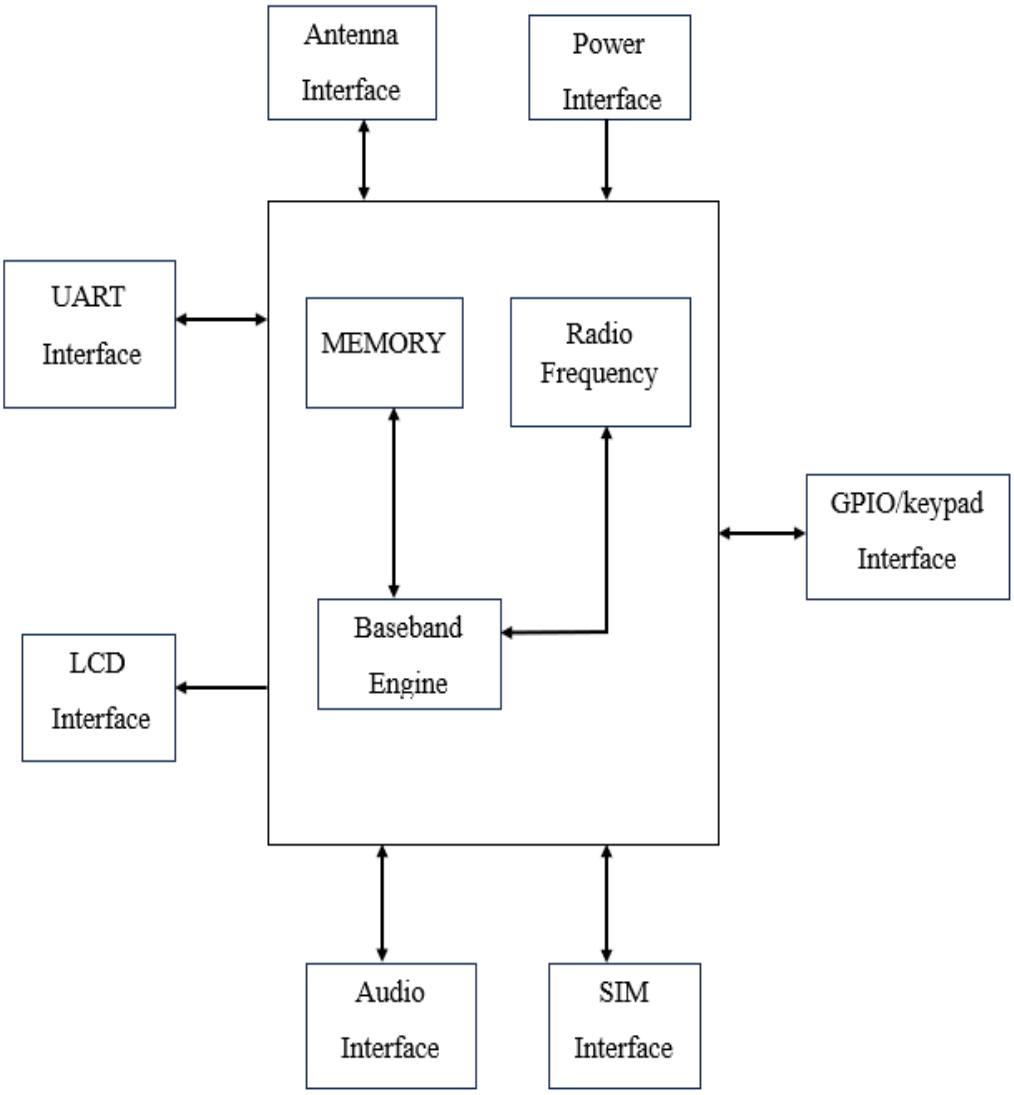


FIG:3.13 Dc Motor Block Diagram

A speech recognition module is a device that enables machines to understand and process human speech, converting spoken language into digital signals for authentication or command execution. It is widely used in security systems, smart home automation, and hands-free control applications. By recognizing voice patterns, the module allows users to interact with devices effortlessly, enhancing both convenience and security.

It then analyzes the unique features of the voice, such as tone, pitch, and frequency, to match it with pre-stored voice patterns. If the authentication is successful, the system grants access or executes a specific action, such as unlocking a door or triggering alarm. The baseband engine acts as the control unit, handling signal processing and managing communication.

By recognizing voice patterns, the module allows users to interact with devices effortlessly, enhancing the authentication of the user input both convenience and security. The given diagram represents the architecture of a communication module, which integrates multiple interfaces to facilitate wireless communication. At the core of the module, there are three key components: memory, the baseband engine, and the radio frequency unit.

3.7 16x2 LCD (Liquid Crystal Display)

Structure and Functionality

The 16x2 LCD consists of a display panel, a controller (usually an HD44780), and an interface for connecting to a microcontroller like the Raspberry Pi Pico. The display works by manipulating liquid crystals to control the passage of light, which is illuminated by a backlight. Each character is displayed using a 5x8 pixel matrix.

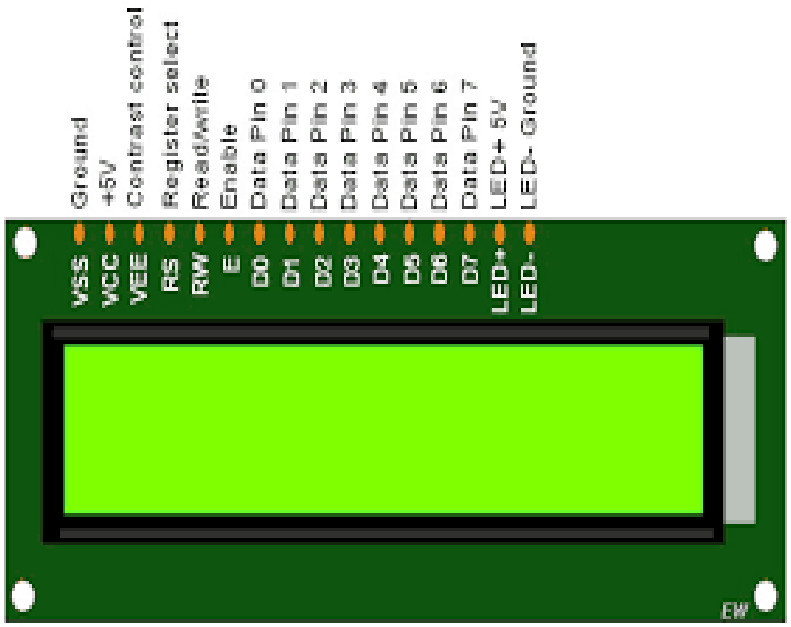


Fig 3.14 16x2 LCD (Liquid Crystal Display)

The Raspberry Pi Pico sends commands via GPIO pins to control text display. Common commands include clearing the screen, moving the cursor, and setting display modes.

pin configuration

A typical 16x2 LCD has 16 pins, which include power, data, and control signals:

1. **VSS (Ground)** – Connected to ground.
2. **VCC (Power Supply)** – Typically 5V or 3.3V.
3. **VEE (Contrast Control)** – Connected to a potentiometer to adjust display contrast.
4. **RS (Register Select)** – Determines if data is interpreted as command (0) or character (1).
5. **RW (Read/Write)** – Reads (1) or writes (0) data.
6. **E (Enable)** – Triggers data reading/writing when pulsed. 7-14. D0 to D7 (Data Pins**)** – Used for 8-bit or 4-bit data communication.
7. **LED+ (Backlight Power)** – Powers the backlight.
8. **LED- (Backlight Ground)** – Connects to ground.

Interfacing with Raspberry Pi Pico

The LCD can operate in 4-bit or 8-bit mode, with 4-bit mode being preferred for saving GPIO pins.

Applications

A 16x2 LCD in a Raspberry Pi Pico-based security system is used to display status messages, such as authentication prompts, access granted/denied notifications, or system alerts. It enhances user interaction by providing visual feedback about the system's operation.

3.8 4*3 KEY BOARD

FIG: 3.15 4X4 Membrane Keypad

Structure and Functionality

The keypad is a membrane-type switch with a flexible flat cable that connects to a microcontroller or microprocessor. Each keypress forms a connection between a specific row and column, allowing the microcontroller to detect which key was pressed by scanning the matrix.

Pin Configuration

The keypad has eight output pins, which represent the four rows and four columns:

1. **Four Row Pins (R1-R4)** – Represent the horizontal connections.
2. **Four Column Pins (C1-C4)** – Represent the vertical connections.

When a key is pressed, it connects a row to a column, allowing current to flow. The microcontroller continuously scans the row and column combinations to detect keypresses.

Interfacing with Microcontrollers

To use this keypad with a microcontroller like Raspberry Pi Pico, Arduino, or ESP32, a scanning algorithm is used. The microcontroller:

1. Sets all columns as input with pull-up resistors.
2. Sets all rows as outputs and pulls them LOW one by one.
3. When a button is pressed, the microcontroller detects which row and column are connected and identifies the pressed key.

Applications

- **Security Systems**: Used for password-based access control.
- **Home Automation**: Controls smart home devices.
- **Industrial Applications**: Menu navigation in embedded systems.
- **Electronic Projects**: Used for user input in DIY electronics.

This 4x4 keypad is lightweight, easy to interface, and ideal for projects requiring numeric or command-based input.

CHAPTER 4

SOFTWARE REQUIREMENTS

4.1 SOFTWARE TOOLS

Embedded system software

A typical industrial microcontroller is unsophisticated compared to the typical enterprise desktop computer and generally depends on a simpler, less-memory-intensive program environment. The simplest devices run on bare meta land are programmed directly using the chip CPU's machine code language.

Often, embedded systems use operating systems or language platforms tailored to embedded use, particularly where real-time operating environments must be served. At higher levels of chip capability, such as those found in SoCs, designers have increasingly decided the systems are generally fast enough and the tasks tolerant of slight variations in reaction time that near-real-time approaches are suitable. In these instances, stripped-down versions of the Linux operating system are commonly deployed, although other operating systems have been pared down to run on embedded systems, including Embedded Java and Windows IoT (formerly Windows Embedded).

Generally, storage of programs and operating systems on embedded devices make use of either flash or rewritable flash memory.

These activities include:

- ✓ Setting up a cloud service provider such as Amazon Web Services, Google Cloud, et
- ✓ Set up private and public keys along with a device certificate.
- ✓ Write a device policy for devices connecting to the cloud service
- ✓ Connect an embedded system to the cloud service
- ✓ Transmit and receive information to the cloud
- ✓ Build a basic dashboard to examine data in the cloud and control the device.

If developers are able to do these things, they will have built a good foundation from which to master cloud connectivity for their embedded systems.

Software requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation. If the authentication is successful, the system grants access or executes a specific action.

Software requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Platform

A computing platform describes some sort of framework, either in hardware or software, which allows software to run. Typical platforms include a computer's architecture, operating system, or programming languages and their run time libraries. Operating system is one of the requirements mentioned when defining system requirements (software). Software may not be compatible with different versions of same line of operating systems, although some measure of backward compatibility is often maintained. For example, most software designed for Microsoft Windows XP does not run on Microsoft Windows 98, although the converse is not always true. Similarly, software designed using newer features of Linux Kernelv2.6 generally does not run or compile properly (or at all) on Linux distributions using Kernel v2.2 or v

APIs and drivers

Software making extensive use of special hardware devices, like high-end display adapters, needs special API or newer device drivers. A good example is DirectX, which is a collection of APIs for handling tasks related to multimedia, especially game programming, on Microsoft platforms.

Web browser

Most web applications and software depend heavily on web technologies to make use of the default browser installed on the system. Microsoft Internet Explorer is a frequent choice of software running on Microsoft Windows, which makes use of Active X controls, despite their.

- ✓ Raspberry is an open-source software that is mainly used for writing and compiling the code into the raspberry.
- ✓ It is an official raspberry, 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, Linux and runs on the Java, python Platform that comes with inbuilt functions and commands that play a vital.
- ✓ It is easily available for operating systems like MAC, Windows, Linux and runs on the Java, python Platform that comes with inbuilt functions and commands that play a vital.

- ✓ The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins, allowing you to control electronic components for physical computing and explore the Internet of Things (IoT).
- ✓ Each of them contains a micro controller on the board that is actually programmed and accepts the information in the form of code.

The Raspberry Pi is a low-cost, credit-card-sized single-board computer that has become immensely popular for education, DIY projects, prototyping, and hobbyist electronics. Developed by the Raspberry Pi Foundation. Programming Languages: The Raspberry Pi supports a variety of programming languages, including Python, C, C++, Java, and many others. Python is particularly popular for educational purposes.

How to Download 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.
- You can download the latest version of Arduino IDE for Windows (Non-Admin standalone version)

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. It is easily available for operating systems like MAC, Windows, Linux and runs on the Java, python Platform that comes with inbuilt functions and commands that play a vital. Operating system is one of the requirements mentioned when defining system requirements (software). Software may not be compatible with different versions of same line of operating systems. The Raspberry Pi is a low-cost, credit-card-sized single-board computer that has become immensely popular for education, DIY projects, prototyping, and hobbyist electronics.

Developed by the Raspberry Pi Foundation. Programming Languages: The Raspberry Pi supports a variety of programming languages, including Python, C, C++, Java, and many others. Python is particularly popular for educational purposes. The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins, allowing you to control electronic components for physical computing and explore the Internet of Things (IoT). Each of them contains a micro controller on the board that is actually programmed and accepts the information in the form of code.

How to Download New Libraries on Arduino IDE

FIG: 4.1 Arduino IDE

- Go to the "tools" section on the top left of the Arduino IDE.

Fig: Arduino IDE step 1

- Select "Manage Libraries" (or) directly press "ctrl+shift+I" on your keyboard.

To get started with the Arduino IDE, first, download it from the official Arduino website.

Choose the version that matches your operating system and install it by following the on-screen instructions.

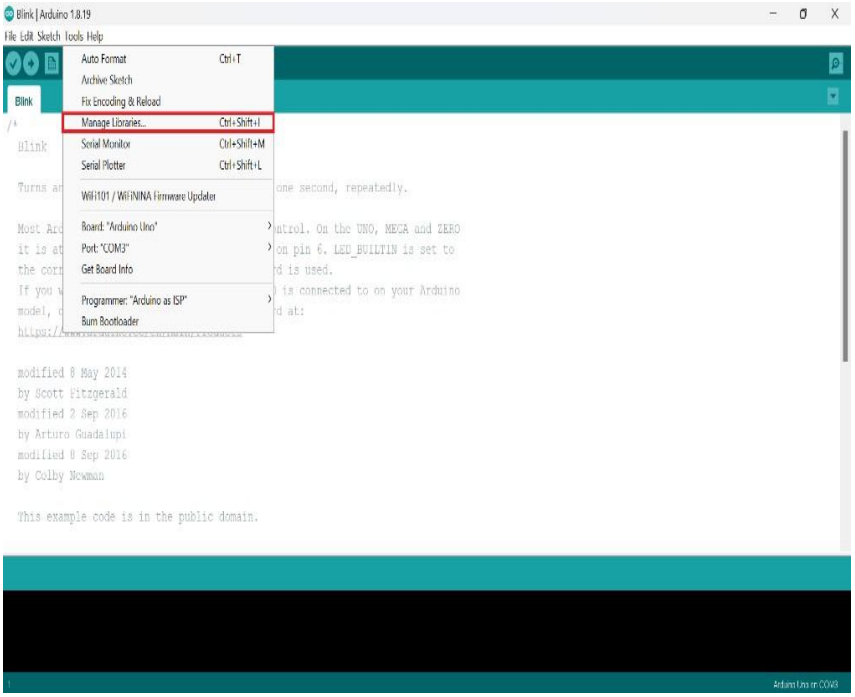


Fig: Arduino IDE step 2

• Now we can observe a dialogue box opening with the name "Library manager".

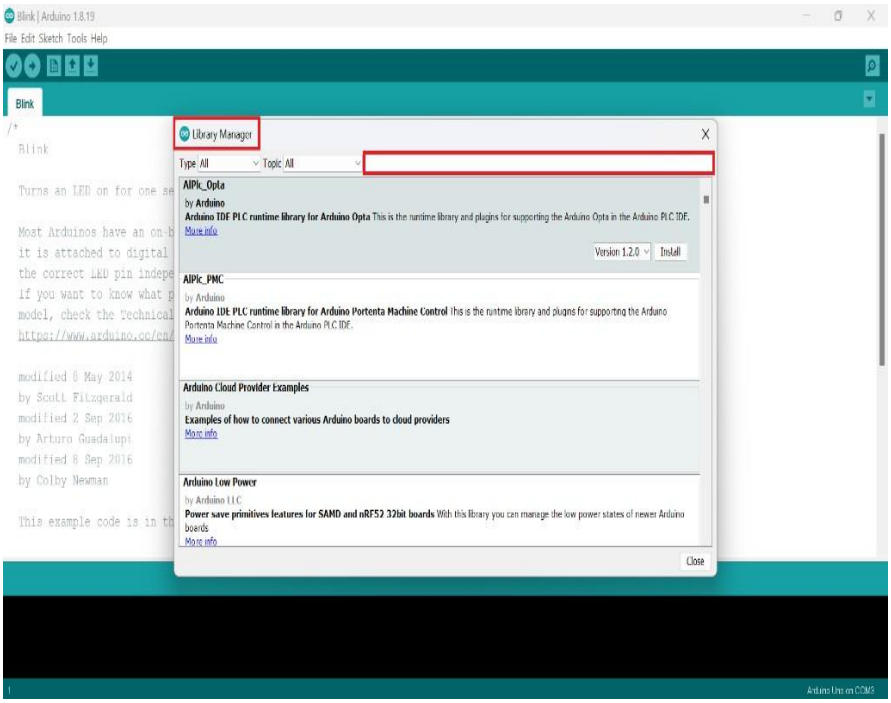


Fig: Arduino IDE step 3

- Search for the library you want to install and press enter.

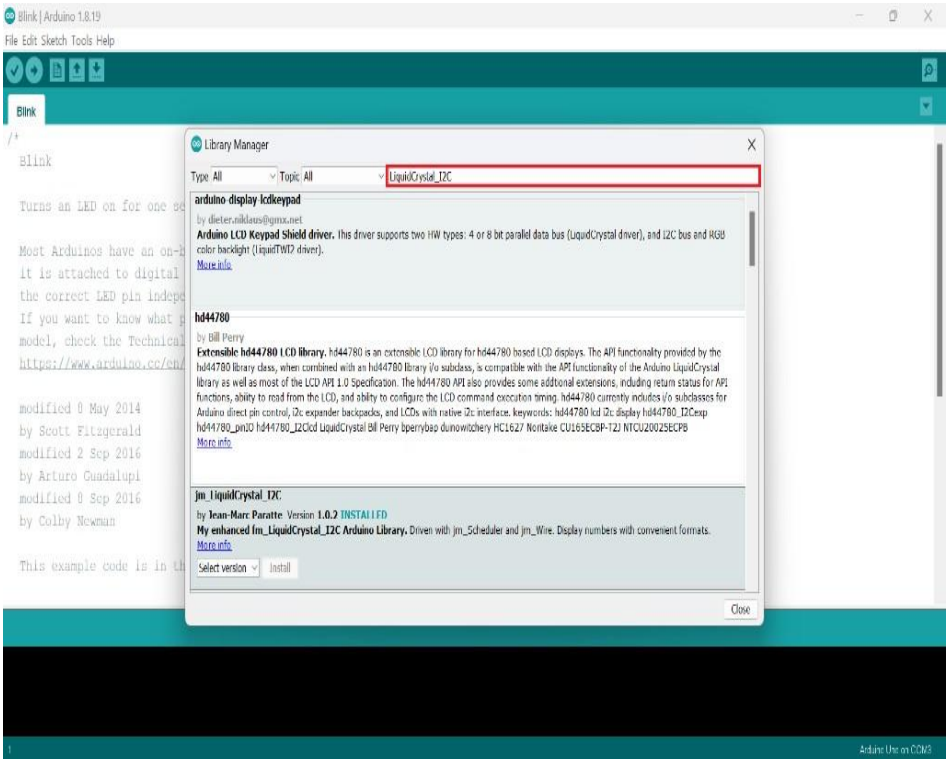


Fig: Arduino IDE step 4

- Select the Library and version you want and click on “Install”.

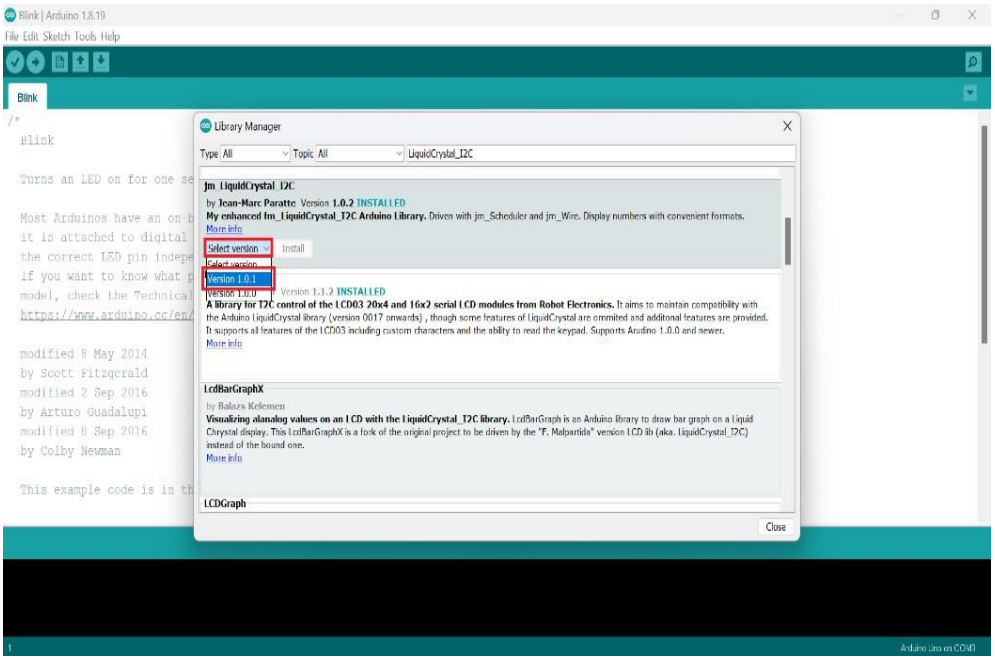


Fig: Arduino IDE step 5

Open the Arduino IDE and go to the Tools menu, where you will find the Board option. From the list, choose the specific model of your Arduino, such as Arduino Uno, Mega, or Nano. After selecting the board, go to the Port option under the same Tools.

such as Arduino Uno, Mega, or Nano. After selecting the board, go to the Port option under the same Tools menu.

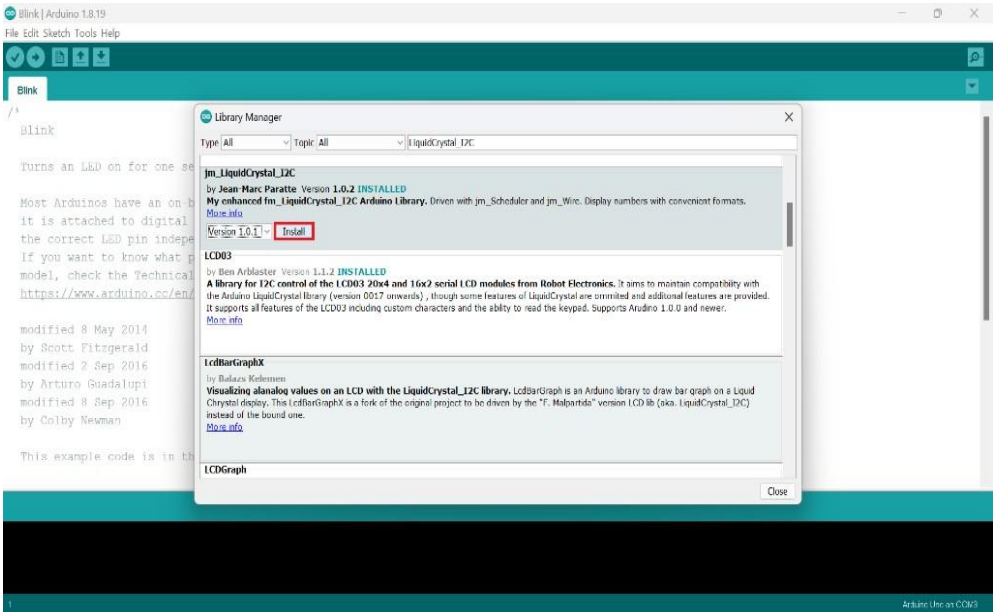


Fig: Arduino IDE step 6

Step-by-Step Guide to Install Raspberry Pi Pico in Arduino IDE

Step 1: Install Arduino IDE

Download and install Arduino IDE

Download the latest version for your operating system (Windows, macOS, or Linux)

Install and open the Arduino IDE.

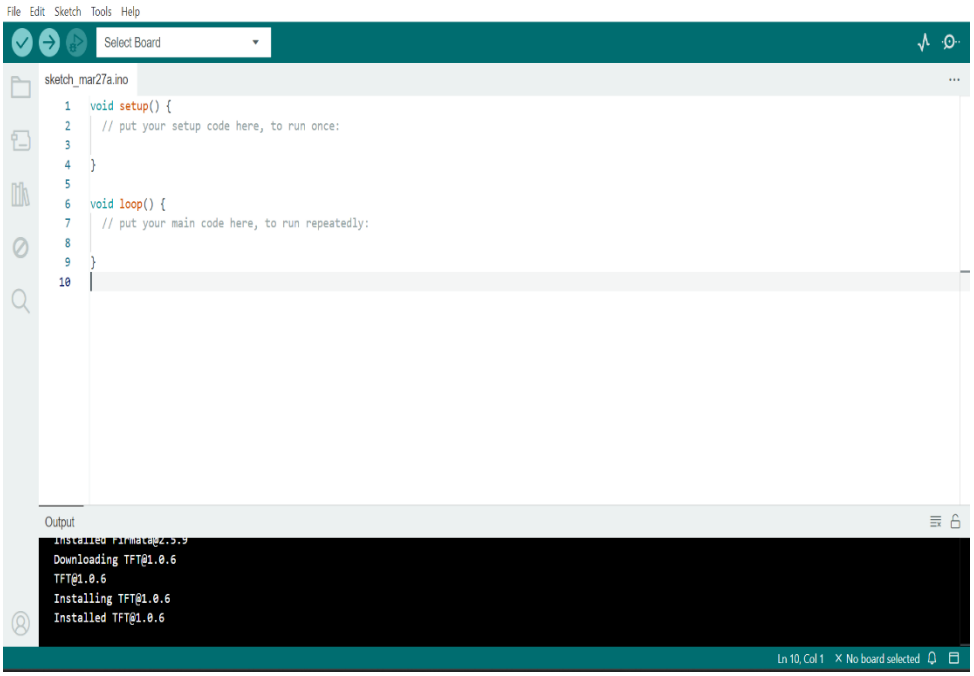


Fig 4.2 Raspberry Pi Pico in Arduino IDE

On Windows, this will typically be labeled as COM followed by a number, while on macOS or Linux, it will appear as a device path like /dev/try USB x or /dev/try AC Mx. If the correct board and port are selected, the Arduino IDE will be able to upload programs to the board.

Step 2: Add Raspberry Pi Pico Board to Arduino IDE

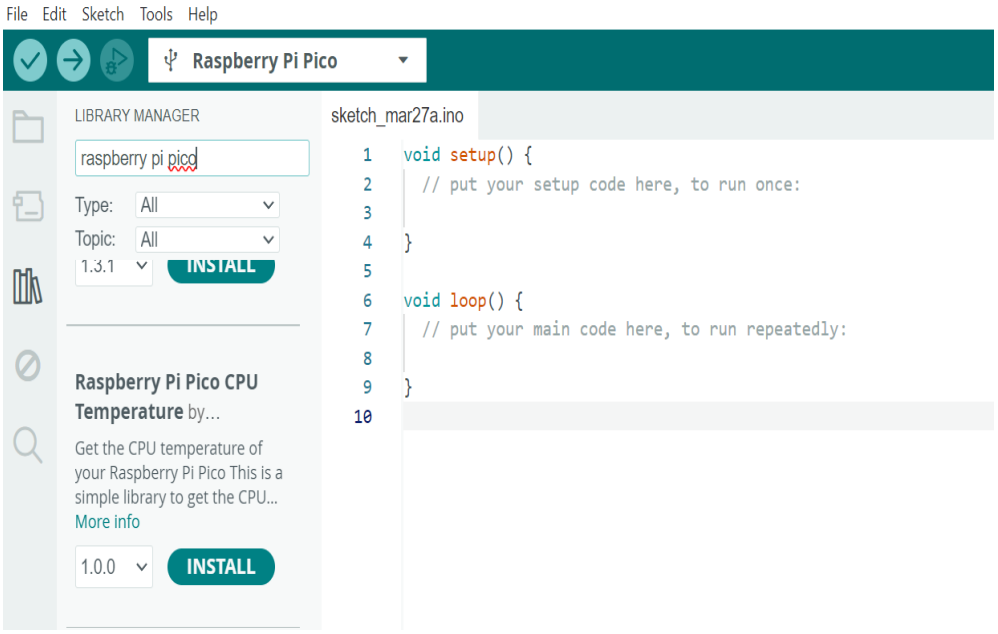


Fig Raspberry Pi Pico in Arduino ide step-2

Step 3: Install Raspberry Pi Pico Board

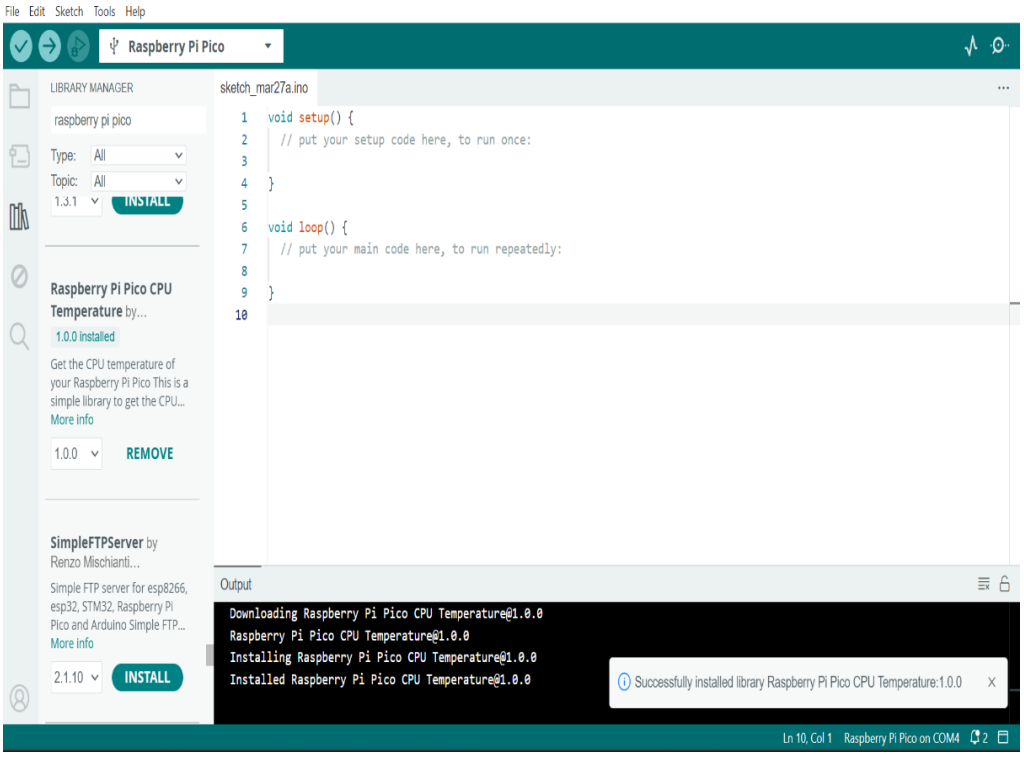


Fig Raspberry Pi Pico in Arduino IDE STEP 3

Step 4: Connect Raspberry Pi Pico to Your Computer

Step 5: Select the Raspberry Pi Pico Board

4.2 RESEARCH

The embedded systems industry was born with the invention of microcontrollers and since then it has evolved into various forms, from primarily being designed for machine control applications to various other new verticals with the convergence of communications. Today it spans right form small metering devices to the multi-functional smartphones. I will cover the areas that are currently focused for development in embedded systems and state what are the ongoing research opportunities in that particular area.

Security

Security remains a great challenge even today. Ongoing Research is to sustain physical tampering, mechanisms to trust the software, authenticate the data and securely communicate over internet. With the advent of IoT/IoE, not only the number devices will continue to increase but also will the number of possible attack vectors. Many challenges remain ahead to get the connected devices on a billion scale.

Connectivity

Wi-Fi, BLE, ZigBee, Thread, ANT, etc have been adapted by embedded system experts from considerable time. Head-on competition between these groups is in progress to determine as to who will emerge as the best solution provider to this huge estimated market of IoT/IoE. 4G/5G on low power devices is the ongoing experimentation which will make embedded systems easily and robustly connect to the internet. Communication using GSM/LTE in licensed/unlicensed communication bands with the cloud can change the ball game of IoE all together.

Memory

Various type of volatile/non-volatile memories with variable sizes and speeds are widely available today. Research is more towards organizing them in best possible architecture to reach closer to the design goal of optimal power-performance-cost.

Energy

Power/Battery management has been under focus for some time. Usage of renewable resources to power device's lifetime is currently the challenge that is tried to address; especially for wearables. Optimal power usage to get Longer Battery Life with new Hardware/Software architectural designs will continue for some time.

System

Multicore (Symmetric/Asymmetric) architectures are experimented since long.

Energy

Power/Battery management has been under focus for some time. Usage of renewable resources to power device's lifetime is currently the challenge that is tried to address; especially for wearables. Optimal power usage to get Longer Battery Life with new Hardware/Software architectural designs will continue for some time.

System

Multicore (Symmetric/Asymmetric) architectures are experimented since long. Addition of GPUs to systems for VR/Gaming/Machine learning is addressed currently.

Programmable SOCs (PSOCs) - (Configurable Hardware Capability) have been there for a long time now, but some has not yet gained momentum. Application-specific computer architectures is also in the pipeline in order to optimize the design matrix of power-performance-cost.

Performance

Real-time on-board Image/Video/Audio processing, feature enabled cameras, on board machine learning are all currently experimented with varied approaches. Commercialization of these technologies has already started but there is still some time to get the best out of these technologies and there is lot of scope to make them more user friendly Other than this, hardening of modular software functionalities (Yes lot of architectures are coming up with hardware performing redundant software functionalities). Ongoing research is to analyze the performance and determine the applications where this strategy can be fruitful.

Networking

Wireless Sensor Networks, Machine to Machine Communication/Interaction, Human Computer Interaction, Security Gateway protocols are still being improved. Light weight algorithms with optimal security will be targeted for embedded systems.

Real Time Operating Systems (RTOS)

Many companies are backing at least one Real Time Open-Source Operating System and there are many out there. Challenge is to cover the wide span of devices, there functionalities and variety of applications

CHAPTER 5

WORKING MODEL AND COMPONENTS

5.1 BLOCKDIAGRAM

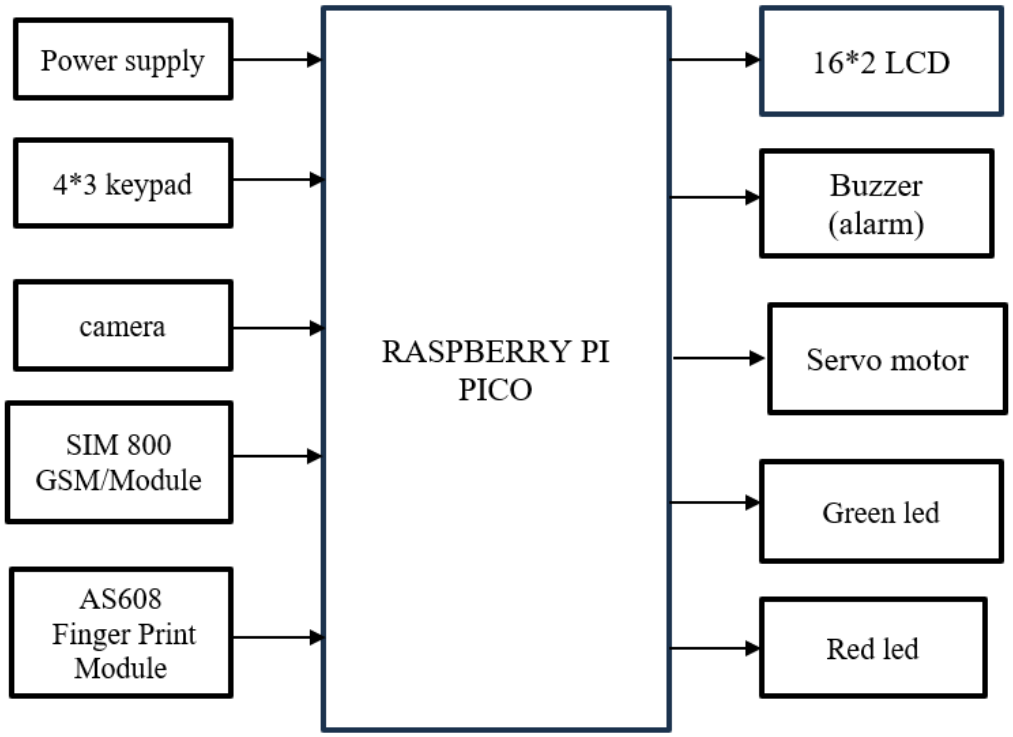


Fig: 5.1 Block Diagram

5.2 WORKING

Advanced security system with multilevel authentication using raspberry pi is designed to provide comprehensive protection for bank locker rooms through a combination of advanced technologies and layered security measures. The system operates as follows.

1. **User Authentication**:

Camera Integration:

- If any authentication attempt fails, the camera captures an image of the user. The image can be stored or sent for further verification.
- Biometric Verification: Users seeking access to the locker room are first authenticated through biometric systems. This includes fingerprint and iris scanners, which ensure that only authorized individuals can proceed.

RFID Access Control:

After biometric authentication, users must present an tag to gain entry. This additional layer of security verifies the user's credentials and controls access to the locker room area.

1. **Access Control:**

- **Main Door Access**: The main door to the locker room is controlled by both biometric and RFID systems. Only users whose biometric data and RFID credentials match the system's records can unlock the door.

2. **Surveillance and Monitoring**:

- CCTV Cameras: Surveillance cameras continuously monitor the locker room area, recording all activities. This footage is used to ensure ongoing security and can be reviewed in case of any incidents.
- **Passive Infrared Sensors:** These sensors detect unauthorized motion within the locker room. If movement is detected outside of authorized hours or areas, the system triggers alarms and captures images of the intruder.

3. **Incident Response:**

- **Alarm Activation:** In the event of unauthorized access or suspicious activity, alarms are triggered to alert local security personnel.
- **Real-Time Alerts**: Images and information about the incident are sent via email or other communication channels to security officials for immediate response.

4. **Activity Logging**:

- **Access Logs**: The system maintains detailed logs of all user activities, including check- ins and check-outs. This log helps track and audit access history and provides transparency for security management.

In summary, the multi-level bank security system integrates biometric and RFID technologies, along with OTP verification and real-time monitoring, to create a robust and secure environment for protecting bank assets and customer funds. Each component works together to ensure that access is strictly controlled and monitored, minimizing the risk of unauthorized entry and enhancing overall security.

The main goal of this project is to design and implement a bank locker security system based on Finger print. This can be organized in bank, offices and homes. In this system only the authenticate person recover the documents or money from the locker Surveillance cameras continuously monitor the locker room area, recording all activities. This footage is used to ensure ongoing security and can be reviewed. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip.

1. System Components and Their Functions

The Raspberry Pi Pico acts as the central controller, receiving inputs, processing authentication, and controlling outputs based on verification success or failure.

Inputs (Authentication Mechanisms)

1. **4x3 Keypad:**
    - Used for entering a PIN or password.
    - Serves as the first authentication level.
2. **Camera:**
    - Captures an image for facial recognition.
    - Acts as the second authentication level.
3. **AS608 Fingerprint Module:**
    - Scans fingerprints for biometric authentication.
    - Acts as the third and final level of authentication.
4. **SIM800 GSM Module:**
    - Sends alerts if unauthorized access is detected.
    - Can be used for remote authentication via OTP (One-Time Password).

Outputs (Security Responses)

1. **16x2 LCD Display:**
    - Displays authentication status and prompts for user input.
2. **Buzzer (Alarm):**
    - Sounds an alert if an unauthorized attempt is detected.
3. **Servo Motor:**
    - Controls door locking and unlocking based on authentication success.
4. **Green LED:**
    - Indicates successful authentication and access granted.
5. **Red LED:**
    - Indicates failed authentication and access denial.
6. **Power Supply:**
    - Provides power to all components for seamless operation.
    - The baseband engine acts as the control unit, handling signal processing.

2. Three-Level Authentication Process

The system implements three levels of authentication to ensure only authorized users gain.

Level 1: PIN or Password Verification (4x3 Keypad)

- The user enters a predefined PIN using the keypad.

- If the PIN is correct, the system proceeds to the next level.
- If the PIN is incorrect, the red LED lights up, and access is denied.

Level 2: Facial Recognition (Camera)

- The camera captures an image of the user and compares it with stored images.
- If the face matches a registered user, the system proceeds to the next level.
- If there is no match, an alert is sent via the GSM module, and access is denied.

Level 3: Fingerprint Authentication (AS608 Module)

- The user places their finger on the fingerprint scanner.
- If the fingerprint matches a stored template, authentication is successful.
- If the fingerprint does not match, the system denies access and triggers the buzzer.

3. Additional Security Features

- **GSM-Based Alert System:** Sends SMS notifications to the owner in case of multiple failed authentication attempts.
- **Time-Based Lockout:** If incorrect credentials are entered multiple times, the system locks for a specified duration.
- **Remote Authentication via GSM Module:** Users can verify their identity remotely by entering an OTP received via SMS.

CHAPTER 6

RESULTS

Enhanced Security: The integration of biometric authentication, OTP technology has substantially improved the security of bank locker rooms. Unauthorized access attempts Are effectively prevented, and the multi-layered approach ensures a higher level of protection compared to traditional methods.

Effective Monitoring and Response: The deployment of passive infrared sensors and surveillance cameras has enabled real-time monitoring of the locker room area. Unauthorized sum movements are promptly detected, and immediate alerts with captured images are sent to security officials, facilitating swift response actions.

Improved User Authentication: The combination of biometric verification and OTP technology has streamlined the authentication process. The system ensures that only verified individuals can access the lockers, minimizing the risk of fraudulent activities

Detailed Activity Logging: The logging system provides a comprehensive record of user activities, including entry and exit times This feature enhances transparency and accountability, aiding in the investigation of any security breaches or anomalies.

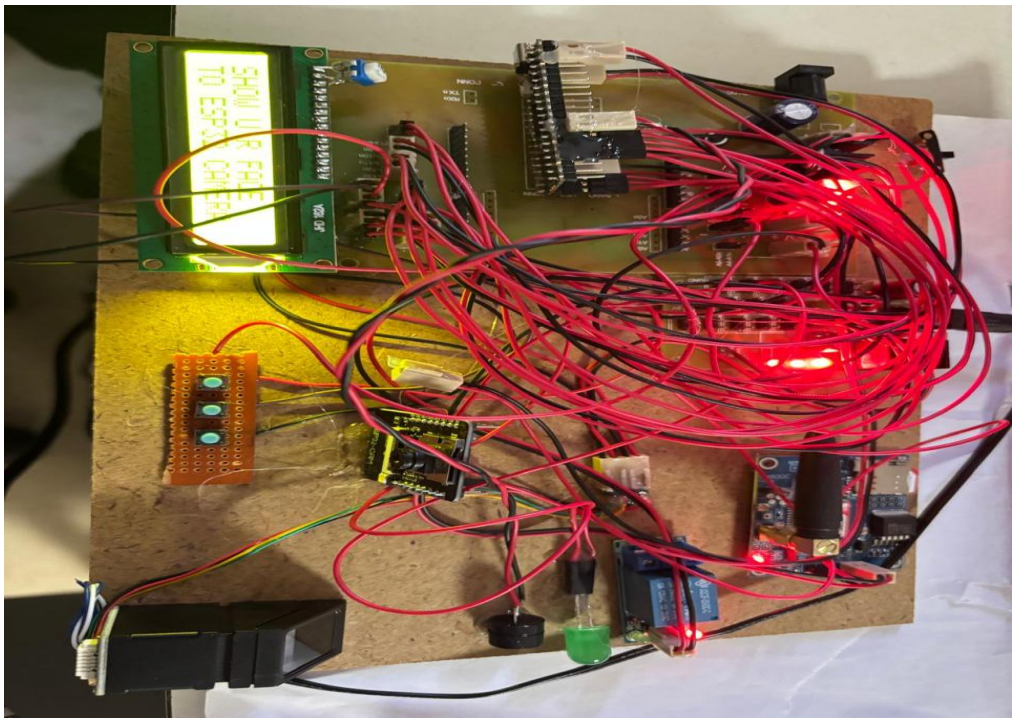


FIG:6.1 Advanced Security System with Multi Level Authentication Using Raspberry pi PICO

The first level of authentication, a 4x3 keypad is used to enter a PIN or password. Step ensures that only users who know the correct PIN can proceed to the next security level. The keypad is connected to the Raspberry Pi Pico, which reads the input and verifies it against a stored PIN. The authentication process consists of multiple levels to enhance security. The first level involves PIN code verification using a 4x3 keypad. The user enters a security code, and the system checks it against pre-stored values. If the PIN is correct, the system moves to the next authentication step. If incorrect, limited retries are allowed before triggering an alert.

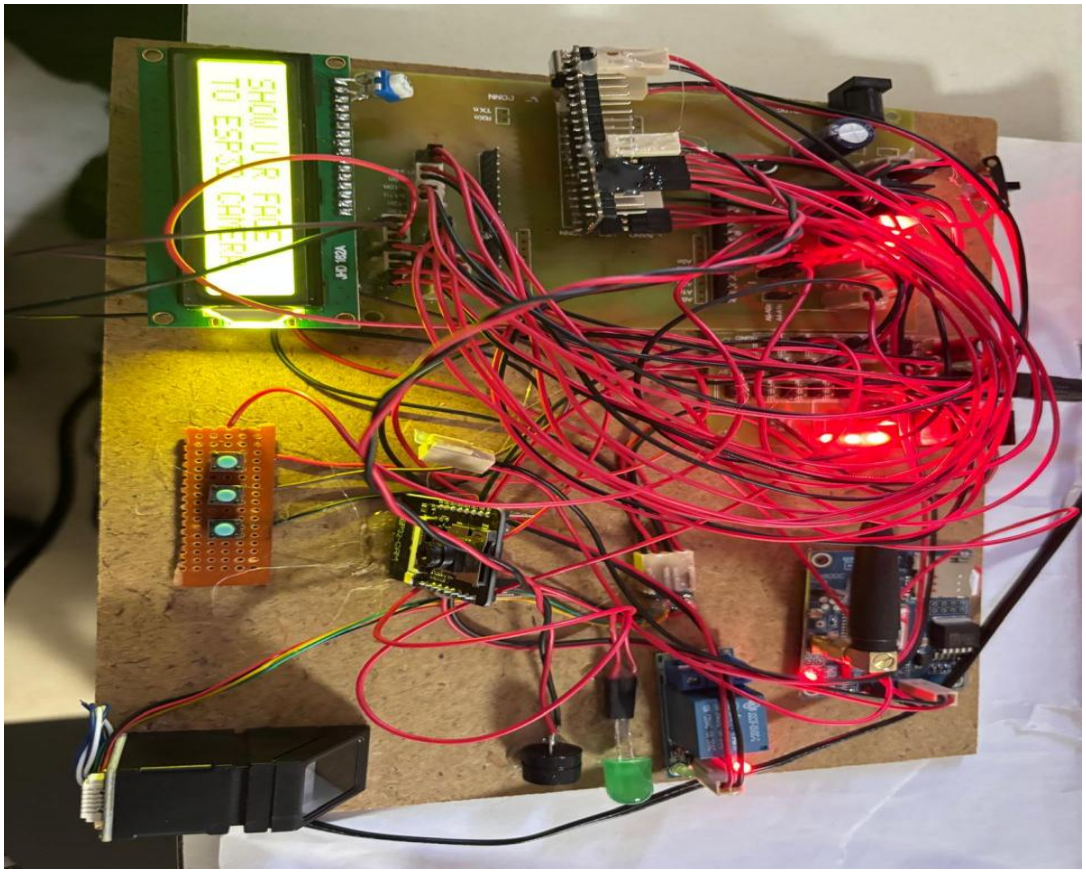


FIG:6.2 LEVEL 1 Pin or Password Verification Using a 4x3 keypad

If the entered PIN matches, the system moves to the next authentication stage; otherwise, access is denied. The 4x3 keypad consists of rows and columns, and when a key is pressed, it connects a specific row and column, allowing the microcontroller to detect which key was pressed. The Raspberry Pi Pico processes this input and checks if it matches the predefined PIN. A 16x2 LCD display is used to prompt the user by showing "Enter PIN" and provides. A buzzer is included to alert in case of multiple incorrect PIN attempts. If the wrong PIN is entered several times, the system triggers an alarm for security purposes. Additionally, red and green LEDs indicate the verification status.

The red LED lights up when the entered PIN is incorrect, while the green LED lights up when the correct PIN. The second level of authentication is biometric verification using the AS608 fingerprint module. When a user places their finger on the sensor, the system compares the scanned fingerprint with the stored templates. If the fingerprint is recognized, access is granted. Otherwise, the user can retry or use an alternative method. In the second level of authentication, the system uses a fingerprint module to verify the identity of the user. After successfully entering the correct PIN in Level 1, the user is prompted to place their finger on the fingerprint sensor. The AS608 fingerprint module captures the fingerprint data and compares it with the stored fingerprint templates in its memory.

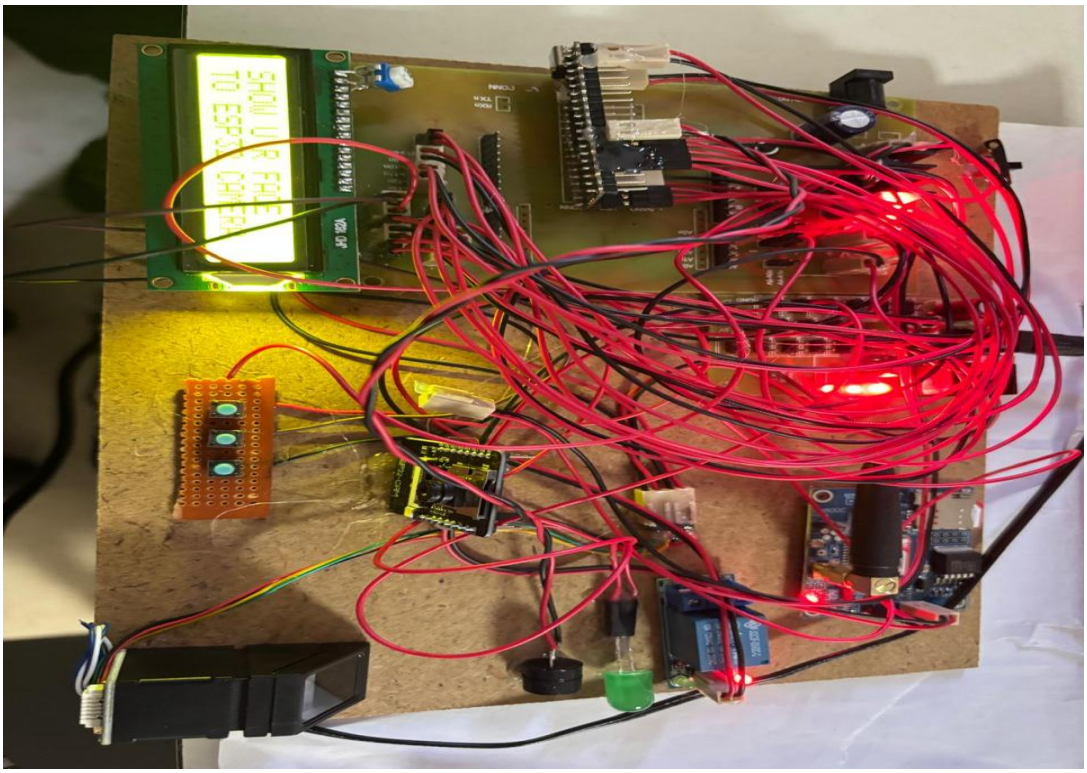


FIG 6.3 LEVEL 2 Facial Recognition (camera)

The Raspberry Pi Pico processes the fingerprint data by sending it to the fingerprint module, which then checks whether the scanned fingerprint matches any of the authorized. If a match is found, the authentication is successful, and the user is allowed to proceed to the final level. If the fingerprint does not match, access is denied, and the system may trigger an alert or reset the process. During this stage, the 16x2 LCD display provides real-time feedback. It first prompts the user to place their finger on the scanner. If the fingerprint is recognized, a message like "Fingerprint Matched" is displayed, and the green LED turns on. If the fingerprint is not found in the database, the red LED lights up.

The system also incorporates a buzzer, which sounds in case of repeated failed attempts. If a user fails to authenticate after multiple tries, the system may either lock temporarily or send an alert via the GSM module, depending on the security. In the third level of authentication, the system employs the SIM800 GSM module to verify the user's identity remotely. After successfully passing the PIN entry (Level 1) and fingerprint verification (Level 2), the system sends a one-time password (OTP) to the registered mobile number of the authorized user via SMS using the SIM800 GSM module. Once the user receives the OTP, they must enter it using the 4x3 keypad.

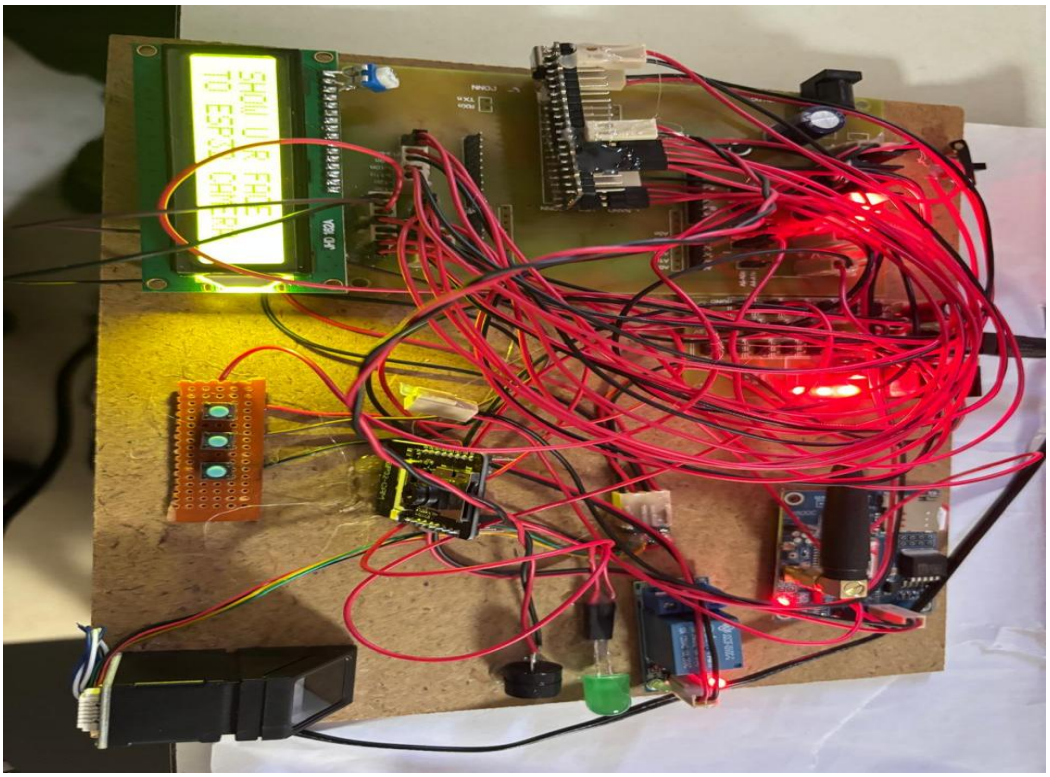


FIG 6.4 LEVEL 3: Fingerprint Authentication (as608 module)

The AS608 fingerprint module provides biometric security, ensuring that only registered users can gain access. Additionally, the 4x3 keypad adds an extra layer of security. The Raspberry Pi Pico successfully manages all components, processing authentication requests and controlling connected devices such as the LCD display, buzzer alarm, LEDs, and servo motor for door access. The Raspberry Pi Pico reads the entered OTP and compares it with the one sent to the mobile device. If the OTP matches, authentication is successful, and access is granted. The system then activates the servo motor to unlock a door, box, or any secured mechanism. If the OTP does not match or is not entered within a specific time limit, the system denies access and may trigger a security alert. The LCD display provides real-time feedback throughout this process.

First, it informs the user that an OTP has been sent. After receiving the OTP, the user is prompted to enter it via the keypad. If the OTP is correct, the display shows "Access Granted," and the green LED turns on. If the OTP is incorrect or not entered in time, the red LED lights up, and the display shows "Invalid OTP." To enhance security, multiple failed OTP attempts may trigger the buzzer as an alarm.

ADVANTAGES

The advanced security system with multi-level authentication using Raspberry Pi Pico enhances security by integrating multiple authentication methods such as PIN entry, fingerprint scanning, and speech recognition**.** This system is designed for high security, real-time authentication, and automated access control**,** making it suitable for applications like home security, office access control, and restricted area protection**.**

1. High-Level Security

By combining multiple authentication factors**,** the system ensures that unauthorized individuals cannot gain access easily. Even if an intruder bypasses one security layer, they must still pass the other authentication steps, making it more secure than traditional single-factor authentication systems. Multi-Authentication for Reliability

The system uses three authentication methods PIN entry (4×3 Keypad), fingerprint recognition (AS608 Module), and speech verification (VC-02 Module) to ensure only authorized users can access restricted areas. If any one of these methods is compromised, the other layers provide additional security.

1. **Fast and Accurate Authentication**

The use of a fingerprint module and speech recognition system enables quick user verification. This reduces wait times compared to manual access control systems and enhances efficiency in high-security environments.

2. **Real-Time Monitoring and Alerts**

The system incorporates a buzzer (alarm) and camera to capture unauthorized access attempts. If authentication fails, the system can trigger alerts via a GSM module (if integrated) to notify security personnel. The camera can take photos of unauthorized attempts for later review.

3. **Cost-Effective and Energy Efficient**

Using Raspberry Pi Pico**,** which is a low-cost and low-power microcontroller, makes this system affordable compared to other high-end biometric security solutions. It consumes minimal power, making it ideal for battery-powered security applications.

This reduces wait times compared to manual access control systems and enhances efficiency in high-security environments. By combining multiple authentication methods such as PIN entry, fingerprint recognition, speech recognition, and GSM-based alerts, the system significantly increases security, making unauthorized access extremely difficult. The integration of a GSM module ensures that security alerts can be sent in real-time to a registered user's phone, providing remote monitoring and quick response to potential security breaches.

4. **User-Friendly Interface**

The 16×2 LCD display provides real-time instructions, guiding users through the authentication process. Simple LED indicators (green for success, red for failure) visually inform users about their authentication status.

5. **Automatic Access Control**

   Upon successful authentication, the servo motor automatically opens doors, safes, or other restricted access points, eliminating the need for manual intervention.

6. **Highly Scalable and Customizable**

   The system can be upgraded to support additional authentication methods like RFID, facial recognition, or NFC-based access control**.** It can also be connected to a cloud platform for remote access control and monitoring.

APPLICATIONS

1. Home Security Systems

Used for smart door locks, allowing only authorized users to enter through multi-level authentication.

Can be integrated with IoT to send real-time security alerts to homeowners via mobile applications.

Helps in preventing unauthorized entry by triggering alarms and capturing images of intruders.

2. Office and Corporate Security

Used for employee access control to restricted office areas, ensuring only authorized personnel can enter. Can be implemented in server rooms and confidential file storage areas to enhance data protection. Maintains log records of entry and exit.

3. Banking and Financial Institutions

Provides multi-layered security for ATM rooms, bank vaults, and safe lockers, reducing the risk of unauthorized access. Can be used for secure authentication in online banking.

4. This security system integrates

multiple authentication methods, including fingerprint recognition, speech authentication, and PIN-based access, providing a robust and foolproof mechanism for securing industrial.

5. Educational Institutions

Used for biometric-based attendance systems, ensuring accurate recording of student and staff attendance. Can be implemented in libraries, laboratories, and exam control rooms to prevent unauthorized entry. Prevents identity fraud in exams by verifying students through fingerprint and speech authentication.

6. Healthcare and Hospitals

Ensures secure access to patient records, medicine storage rooms, and ICU units. Helps in doctor and nurse authentication for restricted the access to areas. Enhances security for automated medicine dispensing systems to prevent misuse.

7. Military and Defence Security Systems

Provides high-level authentication for entry into military bases, weapon storage rooms, and classified areas. Can be combined with facial recognition and RFID-based military ID systems for added security. Logs all access attempts for security monitoring.

8. Smart Lockers and Storage Systems

Used in parcel delivery lockers, ensuring only the authorized recipient can retrieve the package. Helps in securing personal storage lockers in gyms, hotels, and shopping malls. Prevents theft and unauthorized access by using multi-level authentication.

9. Automated Attendance and Workforce Management

Helps in tracking employee attendance using fingerprint and speech recognition. Eliminates the risk of buddy punching (proxy attendance) in offices and factories. Can be integrated with payroll systems for automated salary processing based on attendance records.

10. Smart Vehicles and Transportation Security

Used for biometric car ignition systems, allowing only authorized users to start the vehicle. Helps in public transport access control, ensuring only registered passengers can board restricted areas of trains or buses. Used for biometric-based attendance systems, ensuring accurate recording of student and staff attendance. Can be implemented in libraries, laboratories, and exam control rooms to prevent unauthorized entry. Prevents identity fraud in exams by verifying students through fingerprint and speech authentication. Can be integrated with payroll systems for automated salary processing based on attendance.

LIMITATIONS:

1. **Limited Processing Power** – The Raspberry Pi Pico has a microcontroller-based architecture, which lacks the processing capability required for complex encryption algorithms and high-speed authentication methods.
2. **Memory Constraints** – The onboard RAM and storage are limited, which can restrict the number of stored fingerprints, OTP logs, and user data, reducing scalability.
3. **No Native Internet Connectivity** – Unlike Raspberry Pi boards with built-in Wi-Fi or Ethernet, the Pico lacks direct internet access. It requires additional modules like ESP8266 or GSM modules for cloud-based authentication and remote monitoring.
4. **Limited Number of GPIO Pins** – While it supports multiple peripherals, the GPIO pin count may limit the number of devices that can be connected simultaneously, such as fingerprint sensors, keypads, GSM modules, and LCD displays.
5. **Power Supply Dependence** – The Pico requires a stable power source, and any fluctuations or interruptions could disrupt authentication processes, causing system failures.
6. **Slower Communication with External Modules** – The UART, I2C, and SPI interfaces can introduce delays when communicating with fingerprint modules, GSM modules, or LCDs, especially if multiple devices are connected.
7. **Security Vulnerabilities** – Without advanced encryption or secure storage mechanisms, sensitive data like stored fingerprints and OTPs could be susceptible to tampering or hacking.
8. **Limited Multi-User Management** – Managing multiple users effectively, such as adding, deleting, or modifying authentication data, can be challenging due to the lack of a built-in user-friendly interface.
9. **Difficulty in Remote Management** – Since the Pico does not have an operating system like Raspberry Pi 4, implementing remote firmware updates or security patches requires additional hardware and manual intervention.
10. **Higher Latency in Multi-Level Authentication** – The step-by-step nature of PIN entry, fingerprint scanning, and OTP verification can introduce noticeable delays, making authentication slower compared to systems with integrated authentication modules**.** The Advanced Security System with Multi-Level Authentication using Raspberry Pi Pico provides a highly secure and efficient solution for access control and authentication. By integrating PIN entry, fingerprint recognition, and speech authentication**,** it significantly.

The Advanced Security System with Multi-Level Authentication using Raspberry Pi Pico provides a highly secure and efficient solution for access control and authentication. By integrating PIN entry, fingerprint recognition, and speech authentication, it significantly reduces the risk of unauthorized access. The use of Raspberry Pi Pico ensures a cost-effective, low-power, and compact implementation suitable for a wide range of security applications.

With features like real-time monitoring, alarm alerts, automatic access control via a servo motor, and camera-based security, the system enhances overall protection for homes, offices, banking institutions, industrial zones, healthcare facilities, and military areas**.** The LCD display and LED indicators improve user interaction, while the system's scalability allows for future upgrades**,** such as integration with IoT, cloud storage, or mobile-based remote access. In conclusion, this multi-level authentication system offers enhanced security, automation, and real-time monitoring**,** making it a reliable and adaptable solution for modern security needs. By leveraging embedded systems and biometric technologies, it ensures a high level of protection against unauthorized access**,** making it a valuable innovation for the future of security systems.

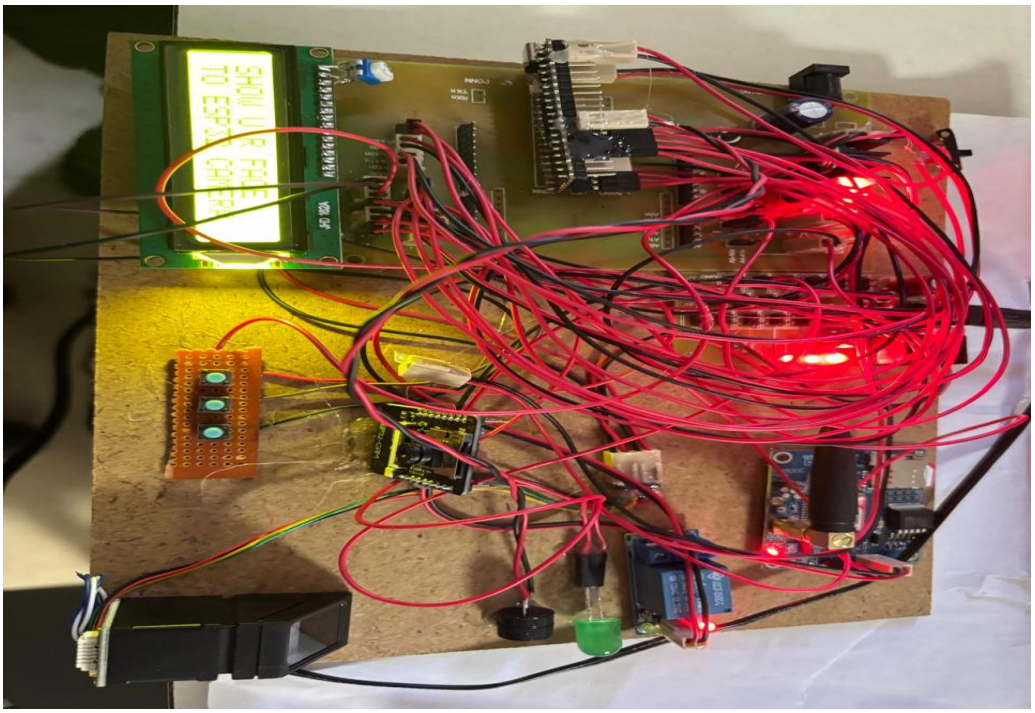


FIG: 6.5 Advanced Security System with Multi-level Authentication Using Raspberry pi PICO

FUTURE SCOPE

1. Integration with IoT and Cloud-Based Security

- The system can be connected to cloud storage to log authentication attempts and store security data for remote access.
- IoT-based smart home integration can allow users to control security access remotely via mobile apps or web dashboards.
- Real-time security alerts can be sent to users via SMS, email, or mobile notifications in case of unauthorized access attempts.

2. AI-Powered Facial Recognition

- A camera module can be integrated with AI-based facial recognition for an additional level of authentication.
- The system can recognize authorized individuals and detect unauthorized users in real time.
- AI algorithms can improve fraud detection by identifying suspicious behaviors or spoofing attempts.

3. Blockchain-Based Security

- Implementing blockchain technology can help in secure data storage and prevent tampering with authentication logs.
- Access control records can be stored in a decentralized and immutable ledger, enhancing trust and security.

4. Biometric Authentication Enhancements

- Additional biometric features such as iris scanning, vein recognition, or palm print scanning can be integrated for even higher security.
- Multi-modal biometrics (combining fingerprint, face, and speech recognition) can improve accuracy and reduce the chances of false acceptance.

5. RFID and NFC-Based Access Control

- The system can incorporate RFID (Radio Frequency Identification) or NFC (Near-Field Communication) for quick and contactless authentication.
- Employees or authorized users can use RFID/NFC cards or mobile devices for seamless access control.

6. AI-Powered Voice Authentication

- Advanced speech recognition AI can be implemented to enhance voice-based authentication accuracy.
- The system can differentiate between genuine users and recorded/spoofed voices.

7. Machine Learning for Security Analysis

- Machine learning algorithms can be used to analyses user authentication patterns and detect anomalies.
- It can help identify unauthorized access attempts, brute-force attacks, or unusual login behaviours.

8. Wireless and Remote Access Control

- The system can be connected to Wi-Fi or Bluetooth for wireless authentication.
- Remote access control via a smartphone app or web interface can allow administrators to manage entry permissions from anywhere.

9. Enhanced Data Encryption and Cybersecurity Features

- Stronger end-to-end encryption algorithms can be implemented to protect user authentication data.
- Secure boot and firmware protection can prevent hacking attempts and unauthorized modifications.

10. Smart Home and Vehicle Security Integration

- The system can be integrated with smart home automation to control doors, windows, and alarm systems.
- It can also be used in automobile security, ensuring only authorized users can start and operate a vehicle.

The future scope of speech recognition modules in security systems is vast, with advancements in artificial intelligence, deep learning, and biometric authentication making these systems more intelligent, secure, and adaptive. As technology evolves, speech recognition will become more accurate, efficient, and integrated with other authentication methods to provide a seamless and highly secure access control system.

One of the most promising developments is the integration of AI and machine learning algorithms, which will enhance voice recognition accuracy by adapting to different accents, speech patterns, and environmental conditions. Future speech recognition modules will be able to filter out background noise more effectively and detect attempts at voice spoofing or imitation. This will prevent unauthorized users from bypassing security using voice recordings or synthesized speech.

REFERENCES

[1]. E. Barker, L. Chen, A. Roginsky, A. Vassilev, R. Davis, and S. Simon, ‘‘Recommendation for pair-wise key establishment schemes using integer factorization cryptography,’’ U.S. Dept. Commerce, Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. NIST SP 800-56B Rev. 2, 2022.

[2]. Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography, U.S. Dept. Commerce, Nat. Inst. Standards Technol., Gaithersburg, MD, USA, 2023.

[3]. M. Mumtaz and L. Ping, ‘‘Forty years of attacks on the RSA cryptosystem: A brief survey,’’ J. Discrete Math. Sci. Cryptogr., vol. 22, no. 1, pp. 9–29, Jan. 2021.

[4]. P. W. Shor, ‘‘Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer,’’ SIAM Rev., vol. 41, no. 2, pp. 303–332, Jan. 2020.

[5]. V. Hassija, V. Chamola, V. Saxena, V. Chanana, P. Parashari, S. Mumtaz, and M. Guizani, ‘‘Present landscape of quantum computing,’’ IET Quantum Commun., vol. 1, no. 2, pp. 42–48, Dec. 2020, doi: 10.1049/ietqtc.2020.

[6]. D. Stebila and M. Mosca, ‘‘Post-quantum key exchange for the Internet and the open quantum-safe project,’’ in Selected Areas in Cryptography— SAC (Lecture Notes in Computer Science), vol. 10532, R. Avanzi and H. Heys, Eds. Cham, Switzerland: Springer, 2021

[7]. A. Ménard, I. Ostojic, M. Patel, and D. Volz, ‘‘A game plan for quantum computing,’’ McKinsey Q, 2020, pp. 7–9. Accessed: Aug. 8, 2024.

[8]. L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, ‘‘NIST released NISTIR 8105, report on postquantum cryptography,’’ Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep., 2019

[9]. L. S. Vailshery, ‘‘Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2021, with forecasts from 2022 to 2030,’’ Transforma Insights, Tech. Rep., 2020.

[10]. J. Barton, N. Pitropakis, W. Buchanan, S. Sayeed, and W. Abramson, ‘‘Post-quantum cryptography analysis of TLS tunneling on a constrained device,’’ in Proc. ICISSP, 2022, pp. 551–561.

[11]. T. Oder, T. Schneider, T. Pöppelmann, and T. Güneysu, ‘‘Practical CCA2- secure and masked ring-LWE implementation,’’ Cryptol. ePrint Arch., 2016.

[12]. S. Ebrahimi, S. Bayat-Sarmadi, and H. Mosanaei-Boorani, ‘‘Post-quantum crypto

processors optimized for edge and resource-constrained devices in IoT,'' IEEE Internet Things J., vol. 6, no. 3, pp. 5500–5507, Jun. 2019.

[13]. J. Hu, M. Baldi, P. Santini, N. Zeng, S. Ling, and H. Wang, ''Lightweight key encapsulation using LDPC codes on FPGAs,'' IEEE Trans. Comput, vol. 69, no. 3, pp. 327–341, Mar. 2020.

[14]. Y. Kim, J. Song, and S. C. Seo, ''Accelerating falcon on ARMv8,'' IEEE Access, vol. 10, pp. 44446–44460, 2022.

[15]. E. Rescorla, ''Diffie–Hellman key agreement method,'' Tech. Rep. rfc2631, 1999.

[16]. T. Saito, K. Xagawa, and T. Yamakawa, ''Tightly-secure key-encapsulation mechanism in the quantum random oracle model,'' in Advances in Cryptology—EUROCRYPT (Lecture Notes in Computer Science), vol. 10822, J. Nielsen and V. Rijmen, Eds. Cham, Switzerland: Springer, 2018. [Online].

[17]. M. Schöffel, F. Lauer, C. C. Rheinlander, and N. Wehn, ''On the energy costs of post-quantum KEMs in TLS-based low-power secure IoT,'' in Proc. Int. Conf. Internet-Things Design Implement., May 2021, pp. 158–168.

[18]. S. Saribas and S. Tonyali, ''Performance evaluation of TLS 1.3 handshake on resource-constrained devices using NIST's third round post-quantum key encapsulation mechanisms and digital signatures,'' in Proc. 7th Int. Conf. Comput. Sci. Eng. (UBMK), Sep. 2022, pp. 294–299.

[19]. K. Burstinghaus-Steinbach, C. Kraub, R. Niederhagen, and M. Schneider, ''Post-quantum TLS on embedded systems: Integrating and evaluating kyber and SPHINCS+ with mbed TLS,'' in Proc. 15th ACM Asia Conf. Comput. Commun. Secur., Oct. 2020, pp. 841–852.

[20]. P. Schwabe, D. Stebila, and T. Wiggers, ''Post-quantum TLS without handshake signatures,'' in Proc. ACM SIGSAC Conf. Comput. Commun. Secur., Oct. 2020, pp. 1461–1480.

APPENDIX

Appendix-1: Gather Components

Before beginning the project, ensure you have all necessary components:

1. Raspberry Pi Pico
2. 4x3 Keypad
3. Fingerprint Sensor
4. 16x2 LCD
5. Servo Motor
6. LEDs and Buzzer
7. Power Supply
8. Camera

Appendix-2: Circuit Design & Wiring

1. **Raspberry Pi Pico Datasheet**
   - RP2040 microcontroller chip
   - Dual-core Arm Cortex-M0+ processor
   - 264KB of SRAM
   - 2MB of onboard Flash memory
   - 26 multifunction GPIO pins
2. **AS608 Fingerprint Sensor Datasheet**
   - Optical fingerprint sensor
   - 512-byte fingerprint template storage
   - 0.1s recognition speed
   - UART interface
3. **16x2 LCD Display Specifications**
   - Interface: Parallel (4-bit or 8-bit)
   - Operating Voltage: 5V
   - Controller: HD44780 compatible

PIN Authentication Test

- Enter correct PIN (1234) → System proceeds to next level
- Enter incorrect PIN → System denies access and sounds alarm

Fingerprint Authentication Test

- Scan registered fingerprint → System grants access
- Scan unregistered fingerprint → System denies access

System Integration Test

- Complete all authentication steps successfully → Servo motor activates
- Fail any authentication step → Alarm sounds and red LED lights up

Appendix-3 Setting Up the ESP32 Environment.

Appendix-4: Setting Up the Raspberry Pi Pico Environment

Appendix-5: Writing the Firmware for Raspberry Pi Pico

Appendix-6: Testing the System

Appendix-7: Install the System in the Target Environment

Appendix-8: Final Testing and Optimization

1. Regularly update the firmware for improvements.
2. Check sensor accuracy and recalibrate if needed.
3. Maintain documentation for troubleshooting and future upgrades.