ISSN 2347 - 3983

Volume 11. No.10, October 2023

International Journal of Emerging Trends in Engineering Research

Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter0211102023.pdf

https://doi.org/10.30534/ijeter/2023/0211102023

Developing a Remote Access System by Interfacing ESP32 Microcontroller with 4X4 Keypad

R. Padmasree[1], P. Harshitha[2], Shaik Muskan[3] Anokya Kalwala[4]

[1]Assistant Professor, Department of ECE, RGUKT-Basar, India; Email: r.padmasree3@gmail.com
[2]B. tech, Department of ECE, RGUKT-Basar, India; Email: harshithaponna@gmail.com
[3]B. tech, Department of ECE, RGUKT-Basar, India; Email: skmuskan1160@gmail.com
[4]B. tech, Department of ECE, RGUKT-Basar, India; Email:anokyakalwala@gmail.com

Received Date: August 27, 2023 Accepted Date: September 26, 2023 Published Date : October 07, 2023

ABSTRACT

Remote access with a microcontroller refers to the capability to control and communicate with a system based on a microcontroller from a remote location. The paper discusses the creation of a reliable remote access and control system using the ESP32 microcontroller and a 4x4 keypad. This system blends ease of use with robust security measures. With the increasing demand for secure access control solutions, this system makes a significant contribution to the realm of IoT-based authentication systems. Future endeavors may encompass scalability testing, integration with other biometric authentication methods, and additional refinements to expand its applicability.

Key words: ESP32 Microcontroller, IoT System, IoT Security, 4X4 Keypad, Remote access

1. INTRODUCTION

In this interconnected world, security is a top priority in both residential and commercial settings. The traditional lock-and-key systems are gradually giving way to advanced access control systems that provide enhanced security, convenience, and scalability. One such innovative system is the Remote Access System that utilizes the ESP32 microcontroller within the realm of IoT (Internet of Things) technology.

This Remote Access System harnesses the capabilities of IoT to establish a secure and efficient access control solution. It incorporates the ESP32 microcontroller as the central control unit, along with various hardware components and IoT connectivity, facilitating seamless communication and control. Through the incorporation of password-based authentication, this system ensures that only authorized individuals are granted access to protected areas or devices, bolstering security measures.

1.1 IoT and its Key components

The term IoT stands for the "Internet of Things." It signifies a network of interconnected physical objects or "things" that are equipped with sensors, software, and related technologies. These elements allow them to gather and exchange data with other devices and systems via the internet. The primary objective of IoT [1] is to empower these devices to communicate, analyze data, and autonomously make decisions, often with minimal human intervention.

Key aspects and principles associated with IoT encompass:

A. *Connectivity*: IoT devices are typically linked to the internet, enabling them to transmit and receive data from remote locations. They can utilize various communication protocols like Wi-Fi, Bluetooth, cellular networks, and energy-efficient, long-range technologies such as LoRaWAN.

B. *Sensors and Data*: IoT devices are furnished with sensors that capture data from their surroundings. This data can encompass details like temperature, humidity, location, motion, light, and more.

C. *Data Processing:* IoT devices often possess onboard or cloud-based processing capabilities to analyze the data they collect. They can derive insights, identify patterns, and make decisions based on this information.

D. *Remote Management*: IoT devices can be remotely supervised and controlled through web interfaces or mobile applications. Users can interact with and oversee these devices from any location with internet access.

E. *Automation*: IoT enables automation and the development of "smart" environments. For instance, smart homes can adjust heating and lighting based on occupancy and user preferences, while industrial IoT can optimize manufacturing processes in real-time.

F. *Scalability*: IoT systems have the capacity to expand from a small number of devices to encompass millions, making them suitable for a wide array of applications, ranging from home automation to expansive industrial and urban deployments.

G. *Security:* Ensuring the security and confidentiality of IoT devices and data is of paramount importance. Proper authentication, encryption, and security measures are essential to prevent unauthorized access and data breaches.

IoT [2] boasts a wide-ranging spectrum of applications across diverse industries, including smart homes, healthcare, agriculture, transportation, manufacturing, and more. It holds the potential to transform how we interact with our surroundings, enhance efficiency, and create fresh business opportunities by harnessing the extensive data generated by connected devices.

1.2 ESP32 Module and its key features.

The ESP32 is a highly capable and versatile microcontroller module that has gained considerable recognition in the realm of IoT (Internet of Things) and embedded systems. It is the creation of Espressif Systems, a prominent semiconductor company renowned for its wireless connectivity solutions. The ESP32 module [3] boasts an extensive array of features and functionalities, rendering it well-suited for a wide spectrum of applications.

Key Features of ESP32:

A. *Dual-core Processor*: The ESP32 module is furnished with a dual-core Xtensa LX6 microprocessor, enabling concurrent processing and enhanced performance, particularly in multitasking scenarios.

B. *Wi-Fi and Bluetooth Connectivity*: One of the standout strengths of the ESP32 lies in its built-in Wi-Fi and Bluetooth capabilities. It supports both Wi-Fi 802.11 b/g/n and Bluetooth 4.2, affording wireless connectivity for IoT applications and facilitating communication with other devices and robotics.

C. *Low Power Consumption*: Designed with an emphasis on energy efficiency, the ESP32 is well-suited for battery-powered applications. It incorporates power management features and various sleep modes to optimize power consumption.

D. *Integrated Peripherals*: The ESP32 module provides a comprehensive array of integrated peripherals, encompassing digital and analog input/output pins, SPI (Serial Peripheral Interface), I2C (Inter-Integrated Circuit), UART (Universal Asynchronous Receiver-Transmitter), ADC (Analog-to-Digital Converter), DAC (Digital-to-Analog Converter), and more. These integrated peripherals simplify the integration of sensors, actuators, and other external components.

E. *Secure Communication*: The ESP32 extends support for various security features, including hardware accelerators for cryptographic operations, secure boot mechanisms, and flash encryption. These features play a pivotal role in safeguarding sensitive data and ensuring secure communication within IoT applications.

F. *Programming and Development*: The ESP32 can be programmed utilizing the Arduino IDE, alongside other popular development frameworks like Espressif's ESP-IDF (IoT Development Framework) and Micro Python. This flexibility empowers developers to select their preferred programming language and development environment, facilitating ease of use and innovation.

The ESP32's impressive combination of capabilities and Espressif Systems' commitment to wireless technology solutions have cemented its position as a versatile and robust microcontroller platform for IoT and embedded system projects.

2. REQUIRED COMPONENTS

The components necessary to enable remote access typically include ESP32 Microcontroller, 4X4 Keypad, Buzzer, Servomotor as shown in figure 1.

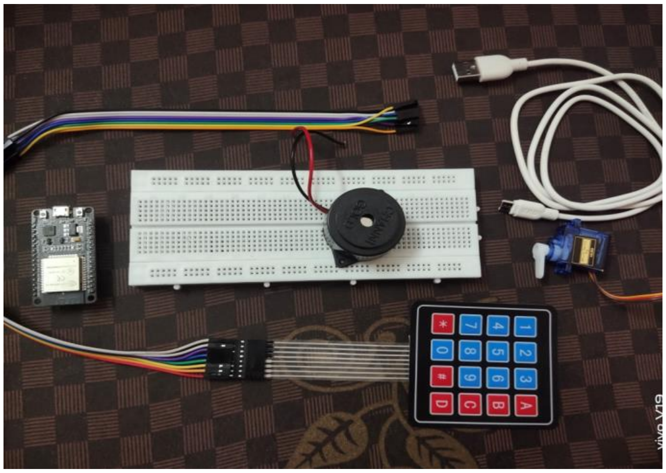


Figure 1: Required components for Interfacing

2.1 ESP32 module.

This module is a sturdy microcontroller with impressive features. It integrates a dual-core Xtensa LX6 microprocessor to boost performance through parallel processing. Equipped with built-in Wi-Fi and Bluetooth connectivity, it enables smooth communication with other devices. Moreover, the module is optimized for energy efficiency, which makes it well-suited for battery-powered applications. It also comes with a variety of integrated peripherals, including 12 digital and analog input/output pins, SPI, I2C, UART, ADC, and DAC.

The ESP32 microcontroller [4] is equipped with a variety of pins, each designed for specific purposes and functionalities. Here is a concise overview of the primary pin categories on the ESP32 as shown in figure 2.

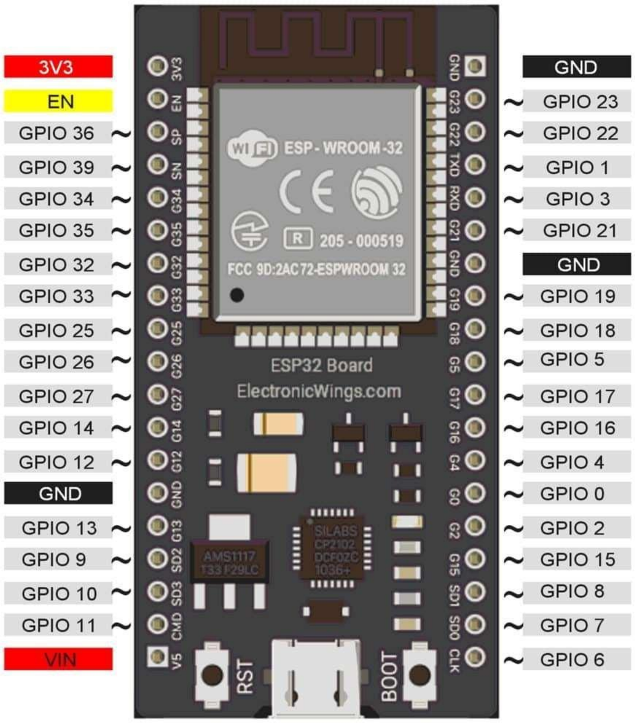


Figure 2: PIN description of ESP32

1. Power Supply Pins:
   VCC: This pin serves as the power supply input, typically connected to a 3.3V power source.
   GND: These pins function as ground connections, establishing the reference voltage for the entire system
2. Digital Input/output Pins:
   GPIO Pins (General-Purpose Input/Output): These pins can be configured as either digital inputs or outputs. They are identified by numerical labels (e.g., GPIO0, GPIO2, GPIO26) and can be applied for diverse purposes, such as connecting sensors, controlling LEDs, or managing external devices
3. Analog Input Pins:
   ADC Pins: The ESP32 offers several pins designed for analog-to-digital conversion (ADC). These pins are denoted by ADC numbers (e.g., ADC1_0, ADC2_4) and are capable of reading analog voltage levels from sensors.
4. Communication Interface Pins:
   SPI Pins: These pins facilitate Serial Peripheral Interface (SPI) communication, encompassing connections for MOSI (Master Out Slave In), MISO (Master In Slave Out), SCK (Clock), and CS (Chip Select).
   I2C Pins: These pins support Inter-Integrated Circuit (I2C) communication and include interfaces for SDA (Serial Data) and SCL (Serial Clock).
   UART Pins: UART pins are intended for serial communication and consist of transmit (TX) and receive (RX) pins.
   CAN Pins: Some ESP32 modules are equipped to handle Controller Area Network (CAN) communication, featuring dedicated CAN pins.
5. PWM Pins:
   PWM (Pulse Width Modulation) pins permit the generation of analog-like signals with adjustable duty cycles. They prove valuable for tasks such as motor control, LED dimming, or sound generation.
6. Internal Components:
   RTC (Real-Time Clock) Pins: These pins are designated for use with the internal RTC module.
   LED Control Pins: In specific ESP32 modules, dedicated pins are allocated for controlling onboard LEDs.
7. Special Function Pins:
   Certain pins are tailored for specialized functions, such as the Touch pins (T0, T1, etc.), designed for touch-sensitive applications.
   Some pins possess multifunctional capabilities contingent upon their configuration.

2.2 4X4 Keypad

The 4x4 matrix keypad functions as an input device commonly employed for input purposes in a wide array of projects. It encompasses a grand total of 16 keys organized in a grid-like pattern with rows and columns. These keys are linked through conductive traces. Typically, there is no electrical connection between the rows and columns. However, when a key is pressed, it initiates a connection between a specific row and column. Interestingly, it operates efficiently with just 8 GPIO pins from a microcontroller, making it an economical and space-saving input solution.

2.3 Buzzer

A buzzer is an electronic apparatus designed to produce either a continuous or intermittent acoustic signal. Typically, it incorporates a piezoelectric component responsible for generating sound when triggered by a microcontroller or other electronic equipment. Buzzers are frequently equipped with built-in oscillator circuits, simplifying the process of activation and control, and enabling precise management of sound production.

2.4 Servo Motor

A servo motor is a type of rotary actuator known for its ability to offer precise control over angular position, velocity, and acceleration. These motors are characterized by their exceptional precision and find extensive use in applications where accurate position control, robust torque output, and variable speed are essential requirements. They play a critical role in various fields, including robotics, automation, and other systems that necessitate meticulous and controlled movements.

3. SYSTEM MODEL

The operational concept of a remote access system utilizing IoT encompasses the integration of several components, which include a microcontroller like the ESP32, a keypad for entering passwords, and a cloud platform for remote access and control [5]. Figure 3 shows visual representation of a circuit diagram designed for the purpose of connection or interaction.

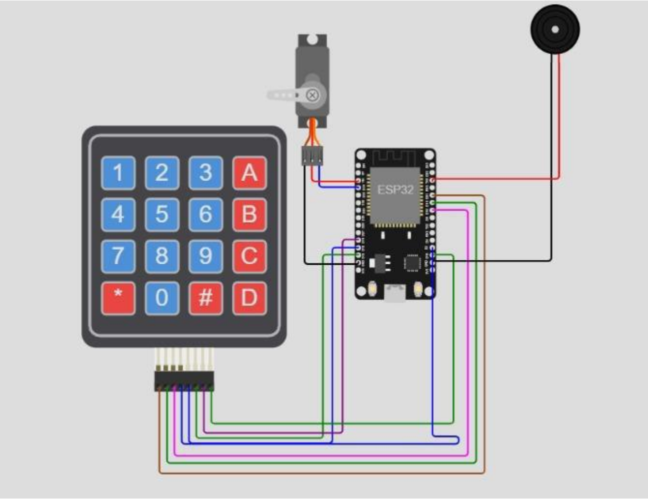


Figure 3: Interfacing of ESP32 Microcontroller with various components for Remote access

The schematic for the Password-Based Door Lock Security System is relatively straightforward. To begin, establish a connection with a 4x4 keypad by utilizing GPIO pins on the ESP32 module. Connect all eight pins of the keypad to the following ESP32 module GPIO pins: GPIO 19, GPIO 18, GPIO 5, GPIO 17, GPIO 16, GPIO 4, GPIO 0, and GPIO 2.

For the servo motor connection to the ESP32 module, use GPIO 25 to output the servo motor's PWM signal. Connect the positive wire to the 3.3 volts pin on the ESP32 module and the negative wire to the ground. Next, connect the positive wire of the buzzer to the 3.3 volts supply and the negative wire to the ground. Finally, establish a USB connection from a PC to the ESP32 module for uploading the code to the ESP32 module.

The operational concept behind an IoT-based password access system [6] includes user input, password validation, access management, IoT connectivity, transmitting data to a cloud platform, incorporating advanced security elements, and the opportunity to integrate and extend connections with other IoT devices and systems. This holistic strategy not only improves security but also provides the flexibility to adapt to a wide range of applications. The working principle is as shown in figure 4.

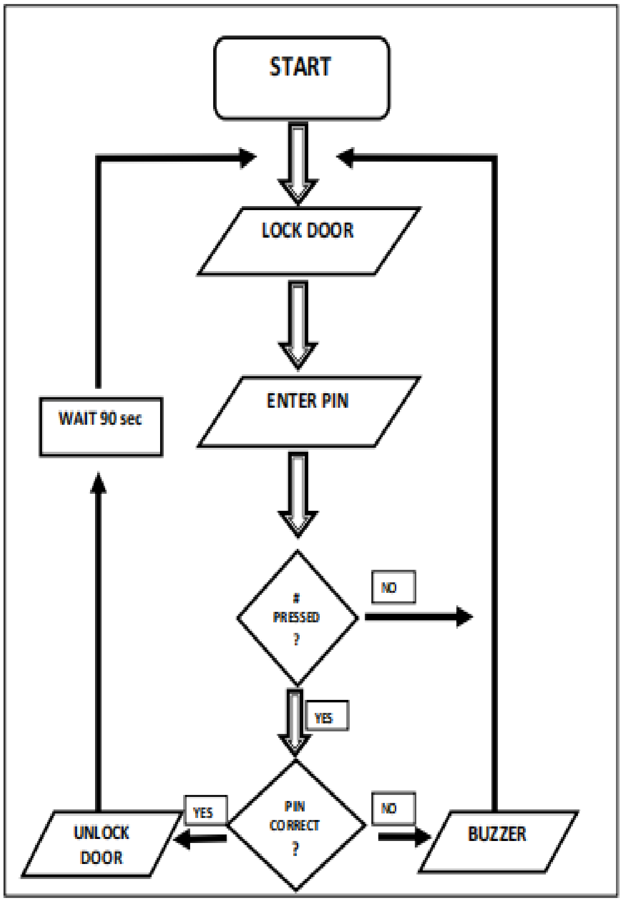


Figure 4: Flow chart of working principle for Remote access

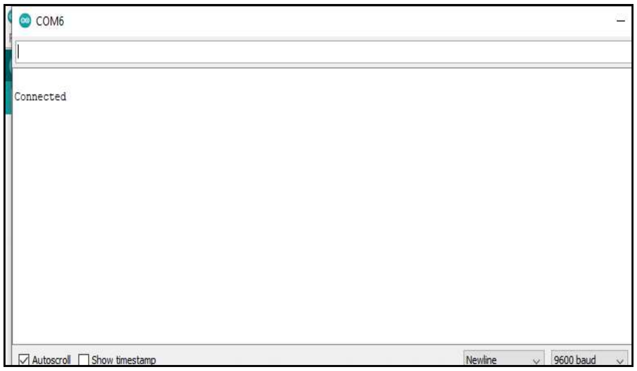
1. *User Input*: The initial step involves the user entering a password via a keypad connected to the microcontroller. This keypad typically contains buttons representing alphanumeric characters and symbols.
2. *Password Verification*: Following the user's input, the microcontroller retrieves the password and compares it to a stored password within its memory or a designated database. A successful match confirms the password's validity.
3. *Access Control*: Upon successful verification, the microcontroller takes action to control access to the targeted system or device. For instance, in our project, it activates a servo motor to open a door.
4. *IoT Connectivity*: Equipped with an IoT module like Wi-Fi or Bluetooth, the microcontroller [7] establishes a connection with either the internet or a local network.
5. *Data Transmission*: Subsequently, the microcontroller uses the established IoT connection to transmit access-related data or status information to a cloud platform [8]. This data includes details of successful authentications or any failed attempts.
6. *Enhanced Security Features*: To bolster security, additional measures like encryption, two-factor

authentication, or biometric verification can be seamlessly integrated into the system.

7. *Future Integration and Expansion*: The IoT-based password access system [9] has the potential for further expansion and integration with other IoT devices and systems. For instance, it can be linked to surveillance cameras, alarm systems, or home automation devices to create a comprehensive security solution.

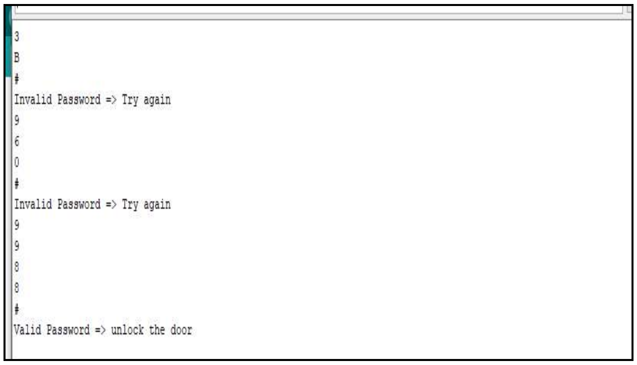
4. SIMULATION

The simulation is conducted using the Arduino IDE, with the requisite libraries for the ESP32 Module, specifically ESP32Servo, being installed, and the Node32s Board is chosen as the designated platform. Initially, the code establishes a Wi-Fi connection and proceeds to declare hardware component pins according to their respective connections. The code also contains predefined valid passwords. Upon code compilation, if the user enters a password on the keypad that matches one of these valid passwords, it triggers the servo motor to rotate, thereby unlocking the door. Conversely, if the entered password is invalid, the code activates the buzzer function. Simulation results are shown in Figure 5 and Figure 6.



```
Connected
```

Figure 5: Simulation Result 1



```
3
B
#
Invalid Password => Try again
9
6
0
#
Invalid Password => Try again
9
9
8
8
#
Valid Password => unlock the door
```

Figure 6: Simulation Result 2

5. CONCLUSION

The IoT-based remote access system presents a secure and user-friendly approach to access control, applicable across residential, commercial, and industrial contexts. Through the integration of IoT elements like microcontrollers and cloud platforms, this password-based access system leveraging IoT offers a robust and adaptable solution. It not only enhances security but also provides added convenience and scalability. With continual advancements and growing market interest, this system is positioned to play a pivotal role in the ever-evolving domain of access control systems.

.

REFERENCES

1. Salman, Tara & Jain, Raj, **A Survey of Protocols and Standards for Internet of Things**, in *Advanced Computing and Communications*, 2017.
2. Gul, S.; Asif, M.; Ahmad, S.; Yasir, M.; Majid, M.; Malik, M.S.A.; Arshad, S**. A survey on role of internet of things in education**. *Int. J. Comput. Sci. Netw. Secur.* 2017.
3. Maier, Alexander & Sharp, Andrew & Vagapov, Yuriy. **Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things,** in *7th International Conference on Internet Technologies and Applications*, UK,2017.
4. O. Barybin, E. Zaitseva and V. Brazhnyi, **On the Testing the Security ESP32 Internet of Things Devices,** in *IEEE International Scientific-Practical Conference Problems of Info communications, Science and Technology*, Ukraine, 2019, pp. 143- 146.
5. Al-Nabhi, Hashem, **Enhanced Security Methods of Door Locking Based Fingerprint**, *International Journal of Innovative Technology and Exploring Engineering*(IJITEE), 2020.
6. D. Aswini, R. Rohindh, K. S. Manoj Ragavendhara and C. S. Mridula, **On the Smart Door Locking System**, in *International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation*(ICAECA), Coimbatore, India, 2021, pp. 1-5.
7. M. Shanthini, G. Vidya and R. Arun, **On the IoT Enhanced Smart Door Locking System**, in *Third International Conference on Smart Systems and Inventive Technology* (ICSSIT), Tirunelveli, India, 2020, pp. 92-96.
8. Sridipta Misra, Muthucumaru Maheswaran, Salman Hashmi, **On Security Challenges and Approaches in Internet of Things,** Springer.com.
9. Hercog, Darko, Tone Lerher, Mitja Truntič, and Oto Težak, **Design and Implementation of ESP32-Based IoT Devices**, *MDPI Sensors*, 2023.