**Title:**

**Bluetooth Password-Protected Door Lock using ESP32 & Arduino**

**ABSTRACT**

In today's world, home security systems are of paramount importance due to the growing concerns over personal safety and property protection. Traditional lock-and-key systems, while widely used, are susceptible to issues such as key loss, duplication, or breakage. With the advancement of technology, smart locks have emerged as an innovative solution, allowing users to secure and access their homes conveniently. This project presents the design and implementation of a Bluetooth Password-Protected Door Lock using the ESP32 microcontroller, which leverages both Bluetooth communication and secure password verification to control a solenoid or servo motor-operated door lock.The main objective of this project is to create a cost-effective, easy-to-install, and user-friendly door lock system that enhances home security by integrating wireless Bluetooth connectivity and password-based authentication. The ESP32 microcontroller, which comes with built-in Bluetooth capabilities, serves as the central control unit of the system. A smartphone or tablet, acting as the user interface, is used to send the password via Bluetooth to the ESP32, which then verifies the entered password against a pre-stored password in the system. If the password is correct, the door lock (either a solenoid lock or a servo motor) is triggered to unlock the door. In case of incorrect password entry, the system remains locked and can send notifications of failed attempts, depending on the added features.

System Components:

* ESP32 Microcontroller: The ESP32 is a powerful microcontroller with integrated Wi-Fi and Bluetooth capabilities. It is widely used in IoT (Internet of Things) applications due to its versatility, low power consumption, and dual-mode Bluetooth functionality. In this project, the ESP32 acts as the brain of the system, handling both Bluetooth communication and the logic required to control the door lock mechanism.
* Bluetooth Module: While the ESP32 has built-in Bluetooth capabilities, this module acts as the communication bridge between the smartphone and the microcontroller. The user sends the password via a Bluetooth-enabled app on their phone, and the module transmits the data to the ESP32 for validation.
* Solenoid Lock or Servo Motor: The solenoid lock is an electromechanical locking mechanism commonly used in secure applications. When voltage is applied to it, the solenoid pulls in a metallic rod, unlocking the door. A relay module is used to interface the solenoid with the ESP32, as the solenoid requires higher current than what the ESP32 can provide. Alternatively, a servo motor can be used to simulate a lock by rotating a latch mechanism.
* Relay Module (for Solenoid Lock): The relay module allows the ESP32 to control the high-power solenoid lock using low-power digital signals. The ESP32 sends a control signal to the relay, which closes or opens the circuit that powers the solenoid.
* Smartphone (Bluetooth Interface): The smartphone application serves as the user interface. It allows the user to send the password to the ESP32 via Bluetooth, initiating the process of unlocking or locking the door. This app can be developed using platforms such as MIT App Inventor, or existing Bluetooth terminal apps can be used for testing.

System Design and Implementation:

The project design consists of both hardware and software components. The hardware aspect involves setting up the ESP32 microcontroller, relay, and solenoid lock in an appropriate configuration to physically control the locking mechanism. The software part focuses on implementing Bluetooth communication and password verification logic using the Arduino IDE, with the ESP32 programmed to receive password input and control the lock.

Step 1: Bluetooth Communication Setup The first step in implementing the system is setting up Bluetooth communication between the ESP32 and the smartphone. The ESP32 acts as a Bluetooth server, advertising itself to nearby devices. A Bluetooth-enabled smartphone can then pair with the ESP32 and establish a connection. This connection allows for data transfer, where the user can send a password string to the ESP32 for verification.

Step 2: Password Verification Once the ESP32 receives a password input from the smartphone, it compares the received password with a pre-stored password in its memory. This pre-stored password can be hardcoded or stored in EEPROM, allowing it to be persistent even when the ESP32 is powered off. If the entered password matches the stored password, the ESP32 will proceed to unlock the door by sending a signal to the relay module (for solenoid locks) or controlling a servo motor (for motor-based locks). In case of an incorrect password, the system will deny access and can optionally send feedback to the user.

Step 3: Controlling the Lock The solenoid lock operates by receiving an electrical current through the relay module, which controls the lock’s actuation. The ESP32 sends a digital signal to the relay, allowing or cutting off power to the solenoid, which physically unlocks or locks the door. Alternatively, in the case of a servo motor-based lock, the ESP32 sends pulse width modulation (PWM) signals to control the rotation of the motor, allowing for mechanical locking and unlocking of the door.

Step 4: Safety and Additional Features To enhance the security and functionality of the system, several additional features can be implemented:

1. Multiple Failed Attempts Lockout: If a user enters the wrong password multiple times consecutively, the system can trigger a lockout period, during which no further password entries are accepted.
2. Notification System: The ESP32 can be integrated with a Wi-Fi module to send notifications via email or SMS when a wrong password is entered, alerting the user of possible unauthorized access attempts.
3. Local Keypad Integration: For redundancy, a local keypad can be integrated into the system to allow manual password entry in case of smartphone malfunction or loss.

Step 5: Power Management Since the ESP32 and solenoid lock require a constant power supply, proper power management is essential to ensure system reliability. The solenoid lock, which typically operates at 12V, is powered separately through the relay module, while the ESP32 is powered by a 5V power supply. The relay ensures the low-power signals from the ESP32 can control the high-power requirements of the solenoid lock.

Advantages of the System:

1. Convenience: Users can unlock their doors wirelessly using their smartphones, eliminating the need for physical keys.
2. Enhanced Security: The system employs password-based authentication, ensuring that only authorized users with the correct password can unlock the door. The password can be easily changed via software updates.
3. Low Cost: The project is cost-effective as it uses readily available components such as the ESP32 microcontroller and a solenoid lock or servo motor, making it affordable for personal or small-scale commercial applications.
4. Scalability: The system can be scaled and modified to include additional features like biometric authentication, voice recognition, or Wi-Fi control for remote access.

Challenges and Considerations:

While the Bluetooth Password-Protected Door Lock offers many advantages, several challenges must be addressed during implementation. These include:

1. Security of Bluetooth Communication: While Bluetooth offers a convenient method of communication, it is susceptible to security vulnerabilities such as eavesdropping or man-in-the-middle attacks. To mitigate this risk, pairing authentication and secure password transmission techniques (such as encryption) can be implemented.
2. Power Consumption: The solenoid lock can consume significant power, especially if it remains energized for long periods. It is essential to design the system such that the lock only consumes power when transitioning between locked and unlocked states, reducing overall power consumption.
3. Physical Installation: The physical installation of the solenoid lock or servo motor must be carefully planned to ensure that it operates smoothly with the door’s locking mechanism.