International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

IoT Based Door Locking System

Mr. Ashok Navale[1], Ayush Ganke[2], Aditya Mane[3], Vinish Sharma[4]

[1] Lecturer, Electronics and Tele-communication Engineering, Vivekanand Education Society's Polytechnic, Chembur, Mumbai, 400071
[2,3,4,]Student, Electronics and Tele-communication Engineering, Vivekanand Education Society's Polytechnic, Chembur, Mumbai, 400071

ABSTRACT :

The IoT-based smart door locking system using ESP8266 and the Blynk IoT platform provides a secure and efficient solution for remote access control. The system enables users to lock and unlock doors remotely using a smartphone app connected via Wi-Fi. The ESP8266 microcontroller processes the user's command and triggers a relay module, which in turn controls a solenoid lock. This eliminates the need for traditional keys and enhances security through real-time monitoring and control. The Blynk app allows seamless communication between the user and the system, offering features like status updates and remote access. The system operates on a 12V power supply and ensures reliable locking and unlocking through the relay mechanism

Keywords— IoT, smart door lock, ESP32, Wi-Fi, solenoid lock, relay, remote access, real-time monitoring, security, automation

1. Introduction :

The IoT-Based Smart Door Locking System is an advanced security solution designed to replace traditional mechanical locks with a more secure and convenient access control system. Traditional door locking systems rely on physical keys, which are prone to loss, duplication, and unauthorized access. These systems also lack the ability to monitor door status remotely, making them less effective in ensuring security.

1.1 Define User-Based Problem

Traditional door locking systems present several security and convenience issues for users. The reliance on physical keys introduces risks such as key loss, unauthorized duplication, and damage, which can compromise the security of the premises. Losing a key or forgetting to lock the door after leaving increases the risk of theft or unauthorized access.

- Inconvenience**:** Managing multiple keys or granting access to guests is difficult.
- Tampering and Mechanical Failure**:** Traditional locks are vulnerable to tampering and wear over time.
- Limited Access Management**:** Traditional locks do not allow for temporary or restricted access for specific individuals.

1.2 Problem Definition

The objective of this project is to develop a wireless EV charging station that can automatically detect an EV and initiate charging using an Arduino-based control system. The system comprises:

1. Users do not receive notifications or alerts about suspicious activity or access attempts.
2. Mechanical failure of locks can compromise security and create inconvenience
3. Existing smart locking systems often depend on external hubs, increasing complexity and cost**.**
4. Manual operation of traditional locks is inconvenient, especially for elderly or disabled individuals.

In short, the relay acts as a bridge between the ESP32 and the solenoid lock, using a small control signal from the ESP32 to switch the high-power solenoid circuit ON and OFF.

2. Literature survey :

The evolution of door locking systems has been influenced by the increasing need for enhanced security and convenience in modern living and workspaces. Traditional mechanical locks have served for centuries but suffer from significant limitations such as key loss, unauthorized duplication, and the lack of remote monitoring. As security challenges have evolved, researchers have explored innovative ways to improve door access control through technology.

- Early research focused on electronic locking systems that used RFID or Bluetooth technologies. These systems allowed users to unlock doors without physical keys, but they were limited by short operating ranges and susceptibility to signal interference.
- With the advent of the Internet of Things (IoT), attention shifted to Wi-Fi-based systems. Studies demonstrated that Wi-Fi connectivity could enable remote access and real-time monitoring, allowing users to control locks via smartphones or web interfaces. Researchers found that integrating IoT with door locks significantly improved user convenience and security.
- Recent literature has emphasized multi-layer authentication methods to enhance security. Studies have incorporated password-based access and biometric verification (e.g., fingerprint recognition) to ensure that only authorized users can gain entry. Although not all systems include biometric methods, the concept of multi-factor authentication has become a cornerstone in modern access control research.
- Several studies have also explored the role of automatic locking mechanisms to mitigate human error, such as forgetting to lock the door. These systems often use delay-based locking functions and real-time notifications to alert users about door status changes.
- The use of microcontrollers, particularly the ESP32, has been highlighted in many research works due to its low power consumption, built-in Wi-Fi capabilities, and robust processing power. ESP32-based systems have demonstrated reliable performance in controlling locking mechanisms, managing sensor inputs, and ensuring secure data transmission through encryption protocols.
- Security remains a primary concern in IoT-based door locking systems. Research in this area has focused on implementing advanced encryption protocols to protect data from hacking attempts and unauthorized access. Studies have shown that secure communication channels and regular firmware updates are essential for maintaining system integrity.

Overall, the literature indicates that integrating IoT technology with traditional locking mechanisms leads to a more secure, flexible, and user-friendly access control system. The proposed project builds on these findings by developing a Wi-Fi-based smart door locking system using an ESP32 microcontroller. This system aims to combine remote control, multi-layer authentication, real-time monitoring, and automatic locking to overcome the limitations of traditional door locks, offering a reliable solution for modern security needs.

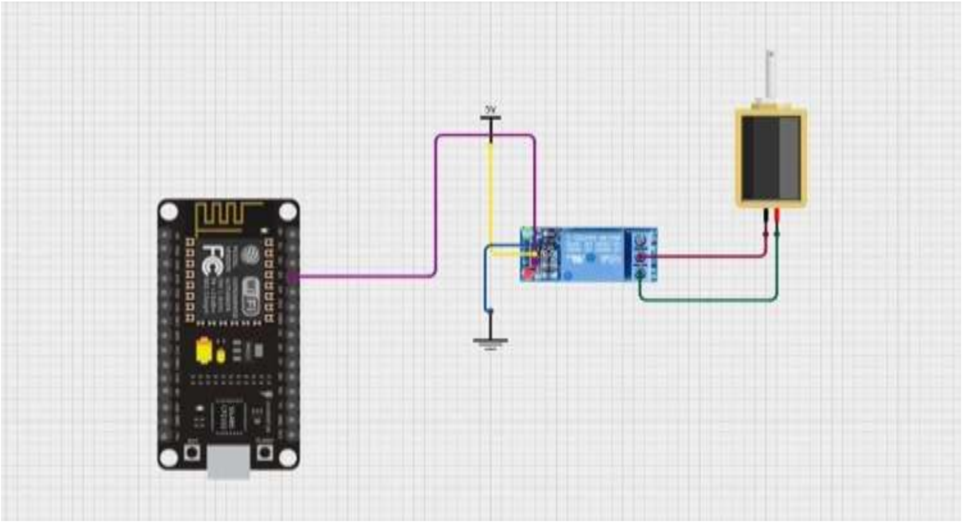

Figure 2: Design of Charging circuit

Working of the Circuit:

Below is a step-by-step explanation of the circuit diagram showing an ESP (NodeMCU/ESP32) controlling a solenoid lock through a 5V relay:

- **ESP Board (Left):**
  - The microcontroller provides the control signal to the relay and shares a common ground.
  - One of the ESP's GPIO pins (shown in pink) is used to drive the relay input.
  - The ESP's GND pin (black line) is connected to the relay's ground.
- **Relay Module (Center):**
  - Powered by a 5V supply (yellow line) from the ESP or an external 5V source.
  - The relay's input pin (pink line) is connected to the ESP's GPIO pin. When driven HIGH or LOW, it energizes or de-energizes the relay coil.
  - The relay's GND pin (black line) must be tied to the ESP's GND.
  - The relay switches the solenoid lock circuit ON or OFF.
- **Solenoid Lock (Right):**
  - One terminal of the solenoid (top wire) is connected to the +5V supply.
  - The other terminal (green line) is connected to the relay's Normally Open (NO) output.
  - When the relay is activated, the NO terminal connects internally to the relay's Common (COM), completing the circuit to ground (or the negative side), thus energizing the solenoid.
- **Power Connections:**
  - A stable 5V power source is needed to drive both the relay coil and the solenoid lock.
  - The ESP itself can be powered either by 5V (through USB or a regulator) but typically operates at 3.3V logic.
  - Ensure the relay is 5V-compatible, and that the solenoid lock matches the available power supply.
- **Operation:**
  - When the ESP GPIO pin is set HIGH (or LOW, depending on relay type), the relay coil energizes.
  - This closes the relay's switch (COM to NO), allowing current to flow from the 5V supply, through the solenoid, and back through the relay to ground, activating the lock (or unlocking, depending on solenoid type).
  - When the GPIO pin changes state, the relay opens the circuit, cutting power to the solenoid.

This setup allows the ESP to control the solenoid lock electronically, enabling IoT-based remote locking or unlocking through Wi-Fi.

3. Block Diagram :

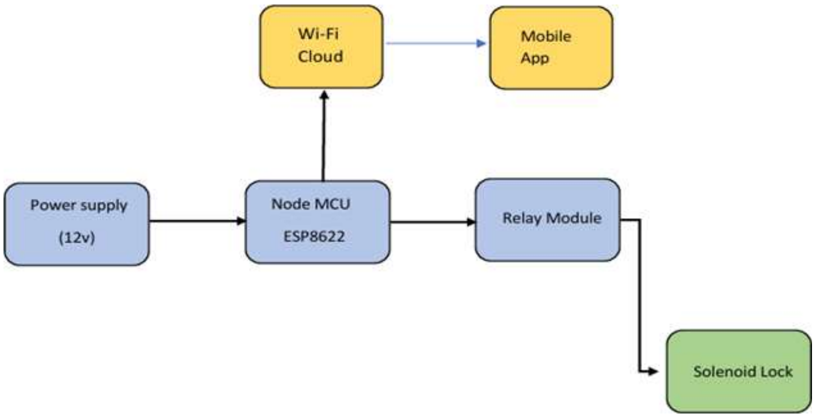

Fig 3. Block diagram

Working of Block Diagram:

1. **Power Supply (12V):**
   - The power supply provides the necessary operating voltage to the NodeMCU (ESP8266) and the relay module.
   - A 12V power source is used to drive the solenoid lock through the relay module.
2. **Node MCU (ESP8266):**
   - The ESP8266 microcontroller acts as the main control unit of the system.
   - It receives power from the power supply and manages the communication between the Wi-Fi cloud, relay module, and solenoid lock.
   - The ESP8266 processes user commands from the mobile app through the Wi-Fi cloud and triggers the relay module accordingly.
3. **Wi-Fi Cloud:**
   - The ESP8266 connects to the internet via Wi-Fi.
   - The Wi-Fi cloud allows real-time communication between the mobile app and the ESP8266.
   - It enables remote access and control of the door lock from anywhere.
4. **Mobile App:**
   - The mobile app sends user commands (lock/unlock) through the Wi-Fi cloud to the ESP8266.
   - The app allows users to monitor and control the door lock in real time.
5. **Relay Module:**
   - The relay module receives a control signal from the ESP8266 to open or close the circuit connected to the solenoid lock.
   - It acts as an electrically controlled switch, allowing low-power signals from the ESP8266 to control high-power devices.
6. **Solenoid Lock:**
   - The solenoid lock physically locks or unlocks the door based on the relay's state.
   - When the relay is triggered, it allows current to flow to the solenoid lock, causing it to engage or disengage.

Working Process:

- The user sends a lock/unlock command from the mobile app.
- The command travels through the Wi-Fi cloud to the ESP8266.
- The ESP8266 processes the command and signals the relay module.
- The relay module switches the solenoid lock ON or OFF based on the signal.
- The solenoid lock engages or disengages accordingly, locking or unlocking the door.

2. Hardware Description :

1. **ESP8266 (Node MCU) :**

- The ESP8266 microcontroller is the core component responsible for controlling the system.
- It comes with built-in Wi-Fi, which allows the system to connect to the internet and communicate with the mobile app.
- It receives commands from the user through the mobile app and sends signals to the relay module to control the solenoid lock.

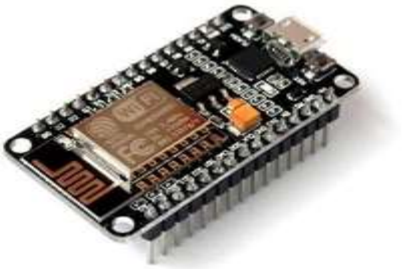

Fig. 4.1 Node MCU 8622

2. **Relay Module (12V):**

- The relay module acts as a switch that controls the solenoid lock.
- It allows the low-power signal from the ESP8266 to control the high-power solenoid lock.
- When the ESP8266 sends a signal, the relay completes the circuit and allows current to flow to the solenoid lock, activating it.

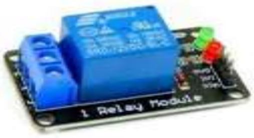

Fig. 4.3 relay module

3. **Solenoid Lock:**

- The solenoid lock is an electrically operated lock that engages or disengages when current flows through it.
- When the relay is activated, it allows current to flow to the solenoid lock, locking or unlocking the door.

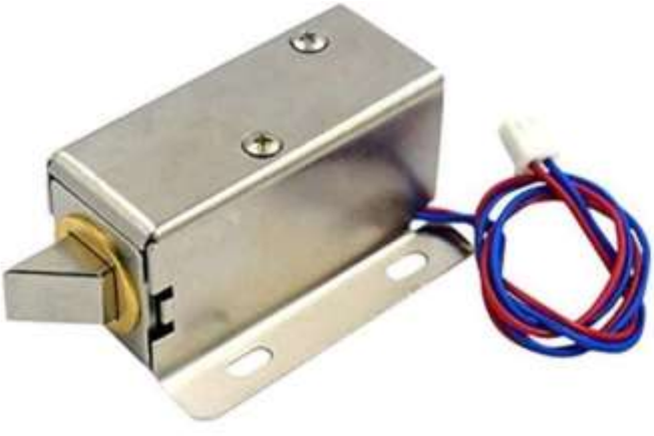

Fig. 4.4 solenoid lock

4. 12V Power Supply

- Function: Provides power to the entire system, including the Arduino, relay module, and wireless charging coils.
- Converts AC mains power to 12V DC.
- Delivers sufficient current for efficient charging.

- Ensures stable operation of all components.

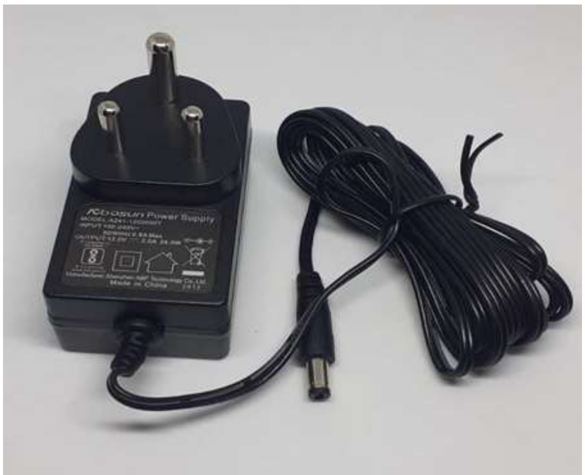

Fig. 4.5 12v power supply

5. **Blynk IoT Platform:**

- The Blynk app is used as the user interface for controlling and monitoring the smart lock.
- The ESP8266 is connected to the Blynk server through Wi-Fi.
- Users can send lock/unlock commands and receive real-time status updates through the Blynk app.

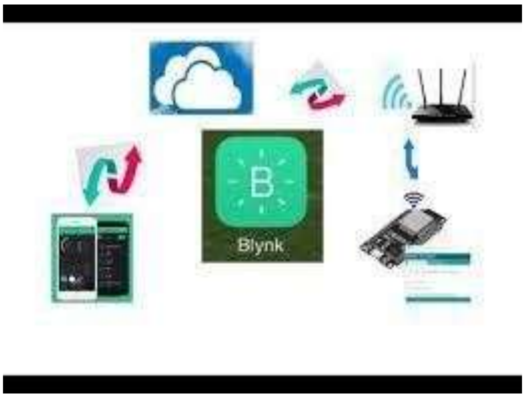

Fig. 4.5 Blynk IOT app

5. Future Directions

- **Integration with Voice Assistants:**

Future upgrades can include integration with popular voice assistants like Google Assistant and Amazon Alexa to enable hands-free control of the door lock.

- **Biometric Authentication:**

Adding fingerprint or facial recognition features can enhance security by allowing only authorized users to unlock the door.

- **Enhanced Security with Two-Factor Authentication (2FA):**

Implementing 2FA (e.g., OTP through mobile or email) will add an extra layer of protection, making the system more secure.

- **Battery Backup and Power Efficiency:**

Future versions can incorporate battery backup to ensure operation during power outages. Optimizing power consumption can also extend the system's lifespan.

6. Conclusion :

The IoT-based smart door locking system using the ESP8266 microcontroller and the Blynk IoT platform provides a secure, reliable, and convenient solution for remote door access control. The integration of Wi-Fi connectivity allows users to monitor and control the lock from anywhere using a smartphone app, improving both security and user convenience. The relay module effectively manages the high-power solenoid lock, ensuring reliable operation with minimal power consumption.

The system enhances security through real-time alerts and the ability to grant or revoke access remotely. It eliminates the risk associated with traditional key-based locks by enabling digital and automated control. The Blynk app ensures a user-friendly interface for easy operation.

REFERENCES :

- Espressif Systems, "ESP8266EX Datasheet," [Online]. Available: https://www.espressif.com.
- Blynk IoT Documentation, "Getting Started with Blynk IoT," [Online]. Available: https://docs.blynk.io.
- Mouser Electronics, "SPDT Relay Module Datasheet," [Online]. Available: https://www.mouser.com.
- Arduino, "Solenoid Lock Control Using Relay and ESP8266," [Online]. Available: https://www.arduino.cc.
- IEEE, "IoT-Based Smart Home Automation System," IEEE Xplore, 2022.