**RFID Card Reader using ESP32**

**OBJECTIVE:**

Design and implement a secure and efficient RFID access control system using the ESP32 microcontroller, with the primary objective of enhancing physical security and access management for a specific application or environment. This project aims to integrate RFID technology with the ESP32 to provide a reliable, user-friendly, and customizable access control solution that ensures only authorized individuals can gain entry while maintaining a detailed access log for monitoring and auditing purposes.

**ABSTRACT:**

This project revolves around the development of an advanced access control system leveraging the capabilities of the ESP32 microcontroller in conjunction with RFID technology. The system aims to enhance security and streamline access management within various settings, such as homes, offices, or industrial facilities. By integrating the ESP32 with RFID technology, this project delivers a robust and customizable solution that enables only authorized individuals to gain access. The system not only ensures security but also maintains a comprehensive access log for monitoring and auditing purposes. This abstract provides a high-level overview of the project's objectives and expected outcomes, which encompass both security and accessibility improvements.

**INTRODUCTION:**

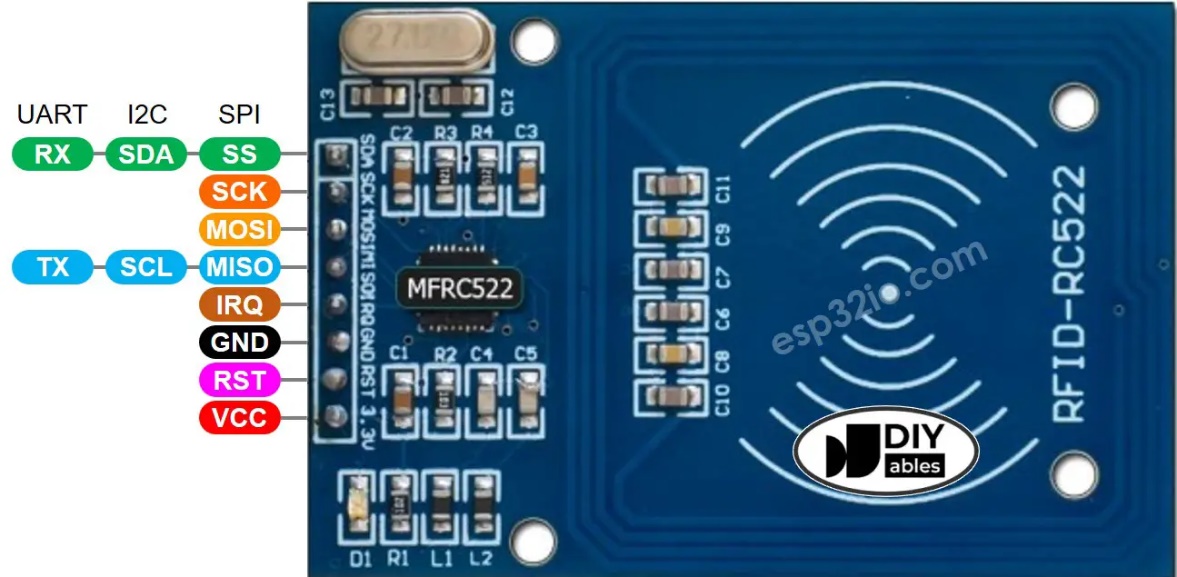
The integration of RFID/NFC technology with the ESP32 microcontroller offers a versatile and powerful solution for access control and security applications, spanning from homes to corporate environments, providing both security and adaptability.

**HARDWARE/SOFTWARE REQUIREMENT & DESCRIPTION:**

* 1x ESP-WROOM-32 Dev Module
* 1 x Micro USB Cable
* 1x RFID/NFC RC522 Kit (reader + tags)
* 1x RFID Key Fob
* 1 x Breadboard
* 1x Jumper Wires

**INTRODUCTION TO RFID-RC522 MODULE:-**

**PINOUT:**

****

The RFID-RC522 module has 8 pins, some pins are shared among three communication Interfaces: SPI, I2C, UART. At a time, only one communication mode can be used.

The pins are:

**GND pin**: connect this pin to GND (0V)

**VCC pin**: connect this pin to VCC (3.3)

**RST pin**: is a pin for reset and power-down. When this pin goes low, hard power-down is enabled. On the rising edge, the module is reset.

**IRQ pin**: is an interrupt pin that can alert the ESP32 when RFID tag comes into its detection range.

**MISO/SCL/TX pins** works as:

* MISO pin if SPI Interface is enabled
* SCL pin it I2C Interface is enabled
* TX pin if UART interface is enabled,

**MOSI pin**: works as MOSI if SPI interface is enabled.

**SCK pin**: works as SCK If SPI interface is enabled

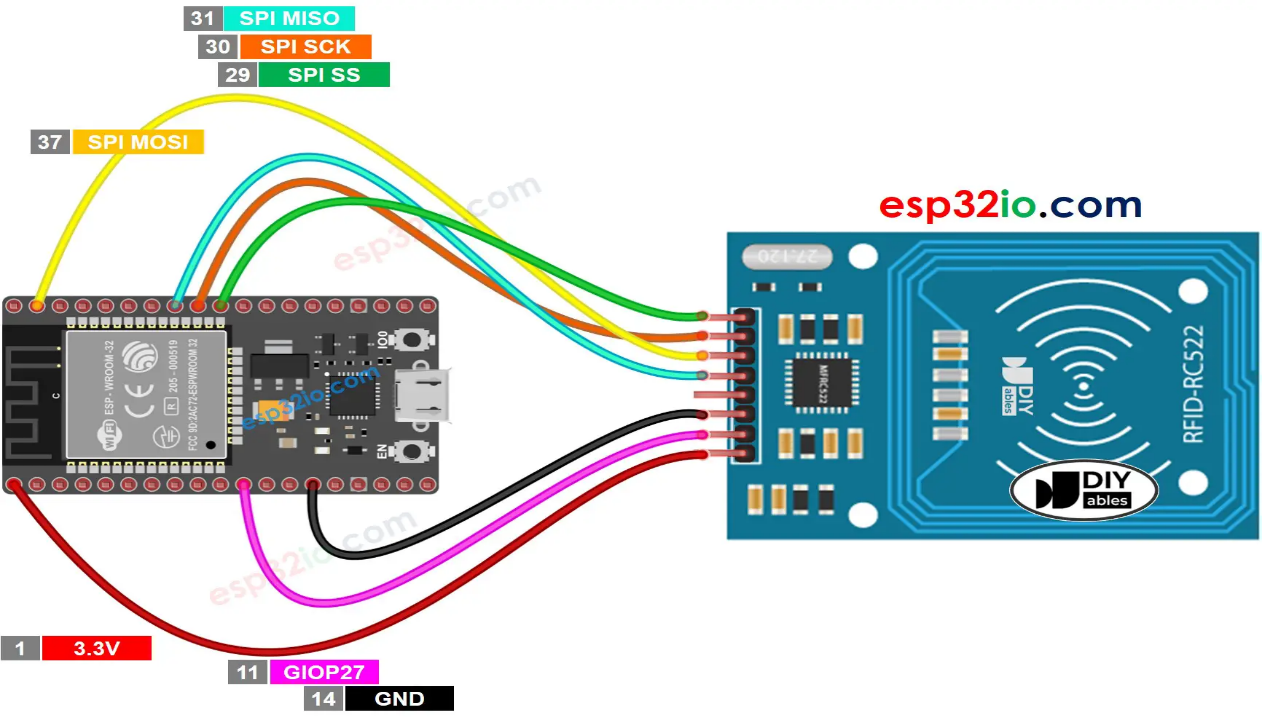
**SS/SDA/RX pin**: works as

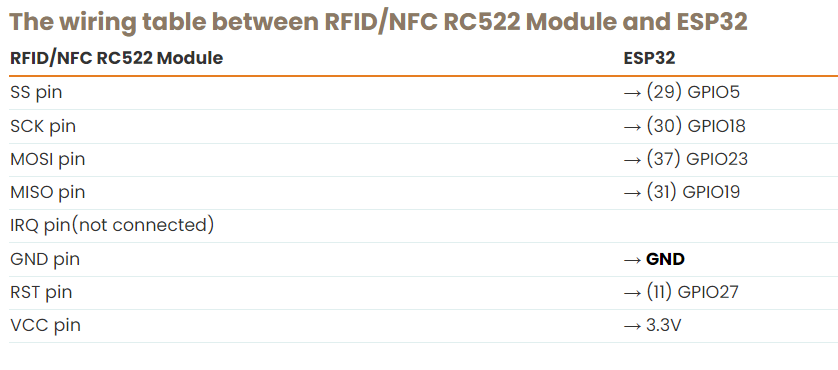
* SS pin if SPI interface is enabled
* SDA pin when I2C interface is enabled
* RX pin when UART interface is enabled

**Note that:**

* The pins order can vary according to manufacturers. ALWAYS use the labels printed on the module. The above image shows the pinout of the modules from **DIYables** manufacturer.
* The RFID-RC522 module works with 3.3V. Do not connect the RFID-RC522 module's VCC pin to 5V. 5V can burn the RFID-RC522 module.
* This project uses SPI Interface for communication between ESP32 and RFID-RC522 module.

**BLOCK DIAGRAM/ INTERFACE DIAGRAM:**

****



**APPROACH/METHODOLOGY:**

* **Understanding NFC Tech:** Study RFID/NFC principles.
* **Component Selection:** Choose ESP32, RFID-RC522.
* **Hardware Setup:** Interfaced ESP32 with RFID using SPI.
* **Software Development:** Used Arduino IDE for control and data logging.
* **Access Control Logic:** Allowed/denied access based on RFID.
* **Data Logging:** Recorded access events, timestamps.
* **User Interface:** Created user-friendly display.
* **Testing & Debugging:** Ensured system reliability.
* **Optimization:** Enhanced performance, minimized power use.
* **Documentation:** Maintained detailed records.
* **Integration:** Combined components for ease of use.
* **Security Assessment:** Evaluated and improved system security

**REALISTIC CONSTRAINTS:**

Realistic constraints for an ESP32 RFID project include:

* **Hardware Limitations**: The ESP32 has limited GPIO pins and available interfaces, which may constrain the number of RFID/NFC readers or other peripherals that can be connected simultaneously.
* **Power Consumption**: The ESP32's power requirements should be carefully managed, especially when running on battery power, as continuous operation can drain the battery relatively quickly.
* **Antenna Range**: The RFID/NFC antenna range is generally limited to a few centimeters to a few meters, which may limit the system's coverage area and the physical placement of readers.
* **Interference**: RFID/NFC systems may suffer from interference, particularly in crowded radio frequency environments, which can affect communication reliability.
* **Security**: Ensuring robust security in an RFID/NFC system is a challenge, and there may be constraints on the level of encryption and authentication that can be implemented.
* **Cost**: The cost of RFID/NFC components and ESP32 modules can add up, especially in larger-scale deployments, which can be a constraint for budget-limited projects.
* **Software Complexity**: Developing and maintaining software for an ESP32-based RFID/NFC system can be complex, and limitations in terms of available memory and processing power may affect the system's functionality.
* **Regulatory Compliance**: RFID/NFC systems must comply with regional radio frequency regulations, which can limit the frequencies and power levels at which the system can operate.
* **Data Storage**: Storing access logs and data on the ESP32 may be constrained by its onboard storage capacity, and additional storage solutions may be required for long-term data retention.
* **Environmental Factors:** Depending on the environment in which the system is deployed, factors like temperature, humidity, and physical wear and tear may pose constraints on the durability and reliability of the system.

Consideration of these constraints is essential when designing and implementing an ESP32 RFID system to ensure that the project meets its objectives while operating within realistic limitations.

**RESULTS AND DISCUSSION:**

The result of implementing an RFID reader using ESP32 is a system that effectively reads RFID tags, enabling access control, data logging, and user-friendly applications in various contexts.



**CONCLUSION:**

In conclusion, the RFID reader system using ESP32 provides a reliable and versatile solution for RFID tag reading, access control, and data management, enhancing security and efficiency across a range of applications.

**REFERENCES:**

<https://esp32io.com/tutorials/esp32-rfid-nfc>

**APPENDIX:**

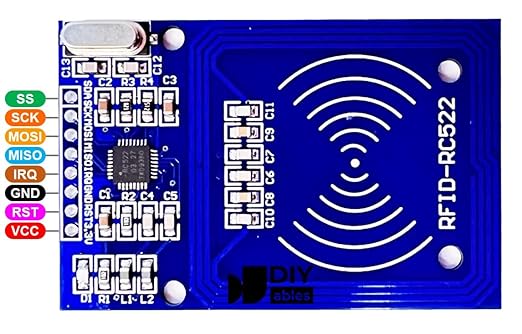
**ABOUT THE COMPONENTS USED**

1. **ESP-32S Development Board :-**



Product reference link:-<https://a.co/d/iOJdphy>

1. **RFID reader and 6 x RFID tags** (1 x S50 white card + 5 x key fobs, 5 colors) :-



Product reference link:-<https://amzn.to/458XFbp>

1. **Amazon Basics USB-A to Micro USB Fast Charging Cable**



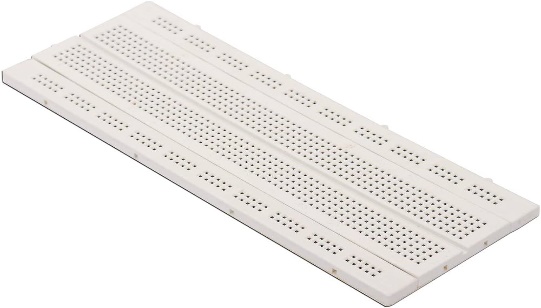
Product reference link:- <https://a.co/d/ahC2ovq>

1. **DIYables RFID Key fob :-**



Product reference link:- [**https://a.co/d/eodGkhl**](https://a.co/d/eodGkhl)

1. **Breadboard :-**



Product reference link:-<https://amzn.to/46Y5TnT>

1. **Jumper Wires for Prototyping**



Product reference link:-<https://amzn.to/3FcLcJH>