**Interfacing RFID Module with Raspberry Pi Pico**

**W and Reading Writing RFID Data Using RC522**

**Contents**

[**1 Introduction** **2**](#_Toc6028)

[1.1 Project Overview 2](#_Toc6029)

[1.2 Objectives 2](#_Toc6030)

[**2 Setup and Wiring** **2**](#_Toc6031)

[2.1 Components Required 2](#_Toc6032)

[2.2 Wiring Diagram 2](#_Toc6033)

[2.3 Wiring Instructions 3](#_Toc6034)

[**3 Programming** **3**](#_Toc6035)

[3.1 Writing Data to RFID Card 3](#_Toc6036)

[3.2 Reading Data from RFID Card 4](#_Toc6037)

[**4 Results and Observations** **5**](#_Toc6038)

[4.1 Successful Write Operation 5](#_Toc6039)

[4.2 Successful Read Operation 6](#_Toc6040)

[**5 Conclusion** **6**](#_Toc6041)

[**6 References** **6**](#_Toc6043)

# Introduction

## Project Overview

This project involves interfacing an RFID module with the Raspberry Pi Pico W microcontroller to read and write data using the RC522 module. The goal is to enable the Raspberry Pi Pico W to read and write data to RFID cards, facilitating various applications such as access control and inventory management.

## Objectives

* Interface the RC522 RFID module with the Raspberry Pi Pico W.
* Write data to an RFID card.
* Read data from an RFID card.

# Setup and Wiring

## Components Required

* Raspberry Pi Pico W
* RC522 RFID Module
* RFID Cards/Tags
* Jumper wires
* Micro USB Cable
* Breadboard

## Wiring Diagram

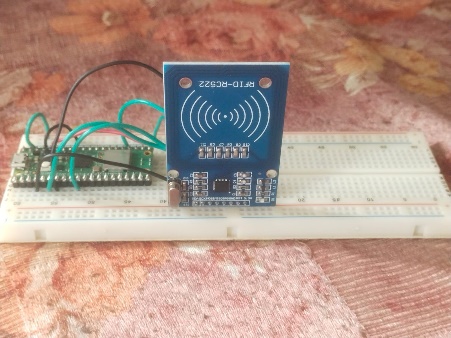
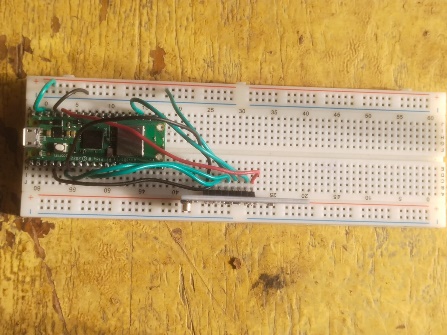
The following table details the connections between the Raspberry Pi Pico W and the RC522 RFID module.

|  |  |
| --- | --- |
| RC522 Pin | Raspberry Pi Pico W Pin |
| SCK | GPIO 2 |
| MOSI | GPIO 7 |
| MISO | GPIO 4 |
| RST | GPIO 18 |
| 3.3v | 3V3 |
| GND | GND |

## Wiring Instructions

1. Connect the SCK pin of the RC522 to GPIO 2 of the Raspberry Pi Pico w
2. Connect the MOSI pin of the RC522 to GPIO 7 of the Raspberry Pi Pico w
3. Connect the MISO pin of the RC522 to GPIO 4 of the Raspberry Pi Pico w
4. Connect the RST pin of the RC522 to GPIO 18 of the Raspberry Pi Pico w
5. Connect the GND pin of the RC522 to GND pin of the Raspberry Pi Pico w
6. Connect the 3.3V pin of the RC522 to the 3V3 (OUT) pin of the Raspberry Pi Pico W

## Circuit Diagram

****

# Programming

## Writing Data to RFID Card

Here is the code to write data (example Hello Devraj) to an RFID card:

from machine import Pin, SPI

from mfrc522 import MFRC522

import utime

# Pin numbers for SPI and RFID

sck = 6

mosi = 3

miso = 4

rst = 0

cs = 5

# Initialize the RFID reader

rdr = MFRC522(sck=sck, mosi=mosi, miso=miso, rst=rst, cs=cs)

# Key for authentication (default key for new cards)

key = [0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF]

print("Place your card near the reader...")

def write\_long\_text(start\_block, text):

block\_size = 16

text\_bytes = [ord(c) for c in text] # Convert text to bytes

num\_blocks = (len(text\_bytes) + block\_size - 1) // block\_size # Calculate number of blocks needed

for i in range(num\_blocks):

block\_data = text\_bytes[i\*block\_size:(i+1)\*block\_size] # Extract block data

block\_data += [0x00] \* (block\_size - len(block\_data)) # Pad with zeros if needed

status = rdr.write(start\_block + i, block\_data)

if status == rdr.OK:

print(f"Data written to block {start\_block + i} successfully")

else:

print(f"Failed to write data to block {start\_block + i}")

try:

while True:

(status, tag\_type) = rdr.request(rdr.REQIDL) # Check if a card is near

if status == rdr.OK:

(status, uid) = rdr.SelectTagSN() # Get the card's UID

if status == rdr.OK:

print("Card detected with UID: ", uid)

# Choose the starting block number

start\_block = 8 # Change this to the starting block

# Authenticate for the chosen block

status = rdr.authKeys(uid, start\_block, keyA=key)

if status == rdr.OK:

print("Authentication successful")

# Prepare and write the long text

text = "Hello this is devraj" # Change this line to your new text

write\_long\_text(start\_block, text)

rdr.stop\_crypto1() # End the communication

else:

print("Authentication failed")

else:

print("Failed to select tag")

except KeyboardInterrupt:

print("Stopped by user")

utime.sleep\_ms(500)

## Reading Data from RFID Card

Here is the code to read data (example Hello Devraj) from an RFID card:

from machine import Pin, SPI

from mfrc522 import MFRC522

import utime

# Pin numbers for SPI and RFID

sck = 6

mosi = 3

miso = 4

rst = 0

cs = 5

# Initialize the RFID reader

rdr = MFRC522(sck=sck, mosi=mosi, miso=miso, rst=rst, cs=cs)

# Key for authentication (default key for new cards)

key = [0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF]

print("Place your card near the reader...")

def read\_long\_text(start\_block, num\_blocks):

text\_bytes = []

for i in range(num\_blocks):

status, block\_data = rdr.read(start\_block + i)

if status == rdr.OK:

text\_bytes.extend(block\_data)

else:

print(f"Failed to read data from block {start\_block + i}")

return None

# Remove padding zeros and convert bytes to string

text = ''.join([chr(b) for b in text\_bytes if b != 0x00])

return text

try:

while True:

(status, tag\_type) = rdr.request(rdr.REQIDL) # Check if a card is near

if status == rdr.OK:

(status, uid) = rdr.SelectTagSN() # Get the card's UID

if status == rdr.OK:

print("Card detected with UID: ", uid)

# Choose the starting block number

start\_block = 8 # Change this to the starting block

num\_blocks = 2 # Change this to the number of blocks written

# Authenticate for the chosen block

status = rdr.authKeys(uid, start\_block, keyA=key)

if status == rdr.OK:

print("Authentication successful")

# Read the long text

text = read\_long\_text(start\_block, num\_blocks)

if text is not None:

print(f"Data read from blocks {start\_block} to {start\_block + num\_blocks - 1}: {text}")

rdr.stop\_crypto1() # End the communication

else:

print("Authentication failed")

else:

print("Failed to select tag")

except KeyboardInterrupt:

print("Stopped by user")

utime.sleep\_ms(500)

# Results and Observations

## Successful Write Operation

When the code for writing data is executed, the message ”Hello Devraj” is successfully written to the RFID card. The console output indicates successful detection, authentication, and writing process.

## Successful Read Operation

When the code for reading data is executed, the message ”Hello Devraj” is successfully read from the RFID card. The console output indicates successful detection, authentication, and reading process.

# Conclusion

In this project, we successfully interfaced the RC522 RFID module with the Raspberry Pi Pico W. We demonstrated the ability to write data to an RFID card and read it back. This setup can be utilized for various applications such as access control, attendance systems, and inventory management.

# References

* Raspberry Pi Pico Documentation: [https://www.raspberrypi.org/docu](https://www.raspberrypi.org/documentation/pico/getting-started/)mentation/ [pico/getting-started/](https://www.raspberrypi.org/documentation/pico/getting-started/)
* MFRC522 Datasheet: [https://www.nxp.com/docs/en/data-sheet/MFRC](https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf)522. [pdf](https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf)
* MicroPython Documentation: <https://docs.micropython.org/en/latest/>