Sudo raspi-config

*Give a new hostname*

Enable SPI

Enable I2C

Enable SSH

Enable VNC

Set Resolution 800x600

**New Steps**

sudo apt-get update

sudo apt-get upgrade

sudo apt-get install python3-pip

sudo apt-get install ntp

sudo apt-get install python3-dev python3-pip

pip3 install spidev

pip3 install --upgrade pip setuptools

Pandas

sudo apt-get install python3-pandas

MariaDB

sudo pip3 install mysql-connector-python

sudo apt-get install mariadb-server-10.0

sudo apt install mariadb-client mariadb-server

# sudo apt-get install mariadb-server

sudo mysql\_secure\_installation

sudo mysql -u root -p -h localhost ß (-h) Connect to a different host

**bingo** ß MariaDB root password

A note about MariaDB and ROOT.

You cannot use root if you want to access the database remotely, you will need to create another user and grant it all the necessary privileges for that to function.

create user cubcaradmin identified by 'cubsrock'; (Replace XXXX with password)

grant all privileges on \*.\* to 'cubcaradmin' ;

If you have MariaDB installed on a raspberry pi and you want to access it remotely you will also have to edit:

sudo nano /etc/mysql/mariadb.conf.d/50-server.cnf

Change bind address to 0.0.0.0 instead of 127.0.0.1

MFRC522

Was originally using SPI connected RFID readers on the arduino, but due to numerous issues conflicting with the NRF24L01+ have switched over to using I2C connected RFID Tag readers.

Have not found i2C python code for RFID tag readers.

Note: The way that the RFID tags are read via Python is very different than how it is interpreted via the Arduino.

## How Arduino and Raspberry PI process RFID cards is very different so we need to go through a transformation to make the numbers work.

## On the RFID card the serial # of the card is stored in HEX in Sector 0.

## The arduino code only uses the first 8 HEX characters, although the codes are longer.

## Here is the conversion process steps:

## A) Convert the integer to HEX

## B) Remove the leading 0X and the last tuple of the code.

## C) Convert the digits back to integer

from mfrc522 import SimpleMFRC522

import os

import RPi.GPIO as GPIO

GPIO.setwarnings(False)

reader = SimpleMFRC522()

try:

id, text = reader.read()

# ut = hex(id)[:10]

ut = hex(id)[2:-2] ## strip off leading 0x and last two digits

print(ut)

mx = int(ut, 16) ## convert back to integer

print(mx)

finally:

GPIO.cleanup()

MFRC522 Tag Reader – PI Install

Two options:

Basic Install on CE0 Install libraries outlined above first then add:

sudo pip3 install mfrc522

OR

1. Create Directory MFRC522 on PI
2. Copy *MFRC522-python-master* from Onedrive\Raspberry Pi \ Libraries \ to Directory (Including all subdirectories)
3. Run python3 setup.py install from within the MFRC522 folder

The version on OneDrive includes changes from

<https://github.com/pimylifeup/MFRC522-python.git>

<https://github.com/pimylifeup/MFRC522-python/pull/11/commits/aa87188579ab5888264d84157d65ec1b13553e49>

<https://github.com/pimylifeup/MFRC522-python/pull/11/commits/0cef34c74c630a15ba838c9ba5bf2ece551597e9>

This allows for more flexibility with the [mfrc522/SimpleMFRC522.py](https://github.com/pimylifeup/MFRC522-python/pull/11/commits/aa87188579ab5888264d84157d65ec1b13553e49#diff-dd794cf84557ed9e5e84b9cf09d360a30c1701db1fb9f254f21dec71e9443840).

Allows for more options when launching Reader

*self.READER = MFRC522(bus=bus, device=device, pin\_rst=pin\_rst),*

*(self, bus=0, device=0, pin\_rst=-1) [defaults]*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Board PIN** | **RS522 Header** | **Colour** | **RPi Pin** | **RPi Function** |
| **1** | 3.3 V | Red | 17 | 3V3 |
| **2** | RST | Orange | 22 | GPIO25 |
| **3** | GND | Brown | 20 | GND |
| **4** | IRQ | Orange |  | Not Connected |
| **5** | MISO | Yellow | 21 | GPIO9 (SPI MISO) |
| **6** | MOSI | Green | 19 | GPIO10 (SPI MOSI) |
| **7** | SCK | Blue | 23 | GPIO11 (SPI CLK) |
| **~~8~~** | ~~SDA~~ | ~~Purple~~ | ~~24~~ | ~~GPIO8~~ (SPI CE0) |
| **8** | SDA | Purple | 26 | GPIO7 (SPI CE1) |

Tag Reader – i2c Arduino

#include "MFRC522\_I2C.h"

|  |  |  |  |
| --- | --- | --- | --- |
| **RS522 Header** | **Diagram Colour** | **Arduino** | **Notes** |
| 3.3 V | Red | 3.3V |  |
| RST/Reset | Orange | 9 |  |
| GND | Black | GND |  |
| I2C SDA/SS | Blue | SDA/SS A4 |  |
| I2C SCL | Yellow | MISO A5 |  |

PCB Layout info

Diagram, schematic

Description automatically generated

On Gen 2 boards (Green not Blue) Bank #1 LEDS is 16,18,7 Not 3,5,7 as pins 3 & 5 are used by the I2C display.