

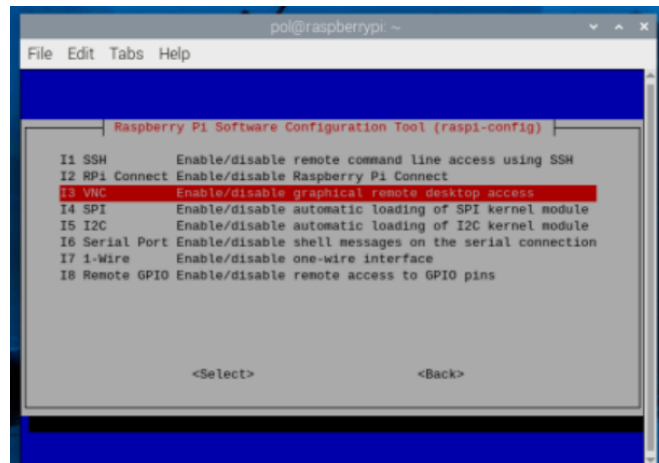
# **Memòria puzzle 1**

Pol Rodríguez Belló

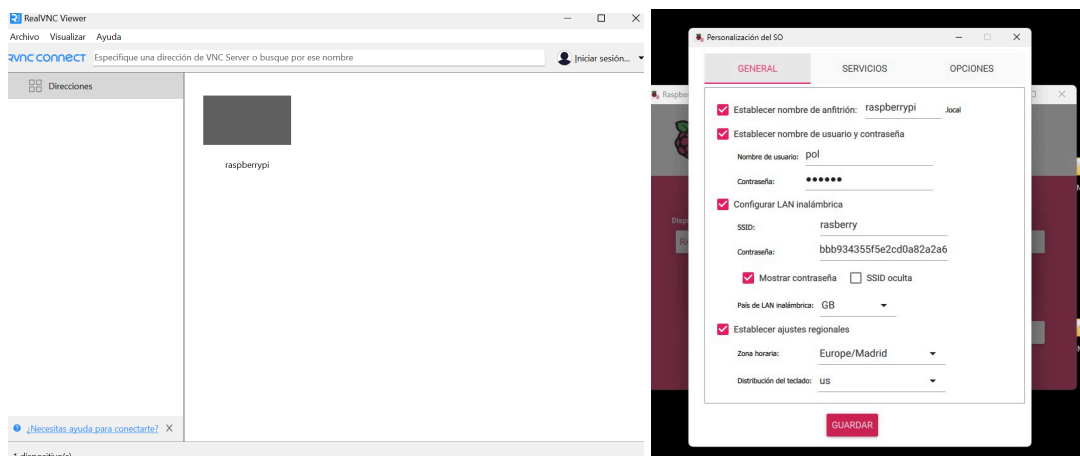
PBE

## 1. Programació de la placa

Primer de tot vaig comprar un micro sd i mitjançant l'aplicació "Raspberry Pi Imager" vaig poder configurar i instal·lar un sistema operatiu compatible amb la meua placa (Raspberry Pi 4). Després, mitjançant un adaptador micro hdmi a hdmi, vaig connectar la placa a un monitor i vaig poder activar el VNC (Virtual Network Computing), que permet utilitzar l'aplicació "RealVNC viewer".



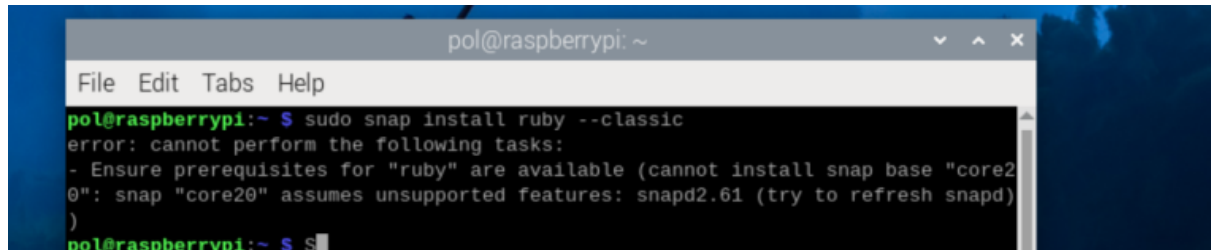
Llavors, vaig obrir aquesta aplicació i, tenint en compte les dades que havia posat en el "Raspberry Pi Imager", vaig poder connectar la placa al portàtil.



Pel que fa a la connexió a internet, he obtat per fer-ho via WI-FI amb un punt d'accés des del portàtil.

## 2. Instal·lacions prèvies

Per començar, com que m'havia tocat programar en ruby, vaig instal·lar-me'l. Primer vaig decidir fer-ho via "Snap", que és un sistema de paquets que permet instal·lar aplicacions de manera senzilla, segura i amb actualitzacions automàtiques, però em va donar problemes.



```
pol@raspberrypi: ~  
File Edit Tabs Help  
pol@raspberrypi:~ $ sudo snap install ruby --classic  
error: cannot perform the following tasks:  
- Ensure prerequisites for "ruby" are available (cannot install snap base "core20": snap "core20" assumes unsupported features: snapd2.61 (try to refresh snapd)  
)  
pol@raspberrypi:~ $
```

Així que ho vaig fer mitjançant "rbenv" (Ruby Environment) amb els següents comandos, que no em van donar cap tipus de problema.



```
pol@raspberrypi:~ $ sudo apt install -y git  
pol@raspberrypi:~ $ git clone https://github.com/rbenv/rbenv.git ~/.rbenv  
pol@raspberrypi:~ $ echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashrc  
pol@raspberrypi:~ $ echo 'eval "$(rbenv init -)"' >> ~/.bashrc  
pol@raspberrypi:~ $ source ~/.bashrc  
pol@raspberrypi:~ $ rbenv install 3.4.2
```

Després, per a treballar amb les diferents biblioteques, vaig instal·lar "Bundler", ja que facilita l'ús d'aquestes.



```
pol@raspberrypi:~ $ gem install bundler
```

Lavors, vaig trobar a github ([https://github.com/atitan/MFRC522\\_Ruby](https://github.com/atitan/MFRC522_Ruby)) una biblioteca per a treballar amb el meu perifèric (RFID-RC522). Pel que me la vaig instal·lar i vaig comprovar els diferents mètodes que tenia.



```
pol@raspberrypi:~ $ gem install mfr522
```

```
pol@raspberrypi: ~/Desktop
File Edit Tabs Help
=>
[:mifare_crypto1_deauthenticate,
:write_spi_clear_bitmask,
:transceiver_baud_rate,
:internal_timer,
:read_spi,
:write_spi_set_bitmask,
:antenna_off,
:antenna_gain,
:buffer_size,
:picc_select,
:reestablish_picc_communication,
:mifare_crypto1_authenticate,
:picc_request,
:communicate_with_picc,
:picc_halt,
:picc_transceive,
:read_spi_bulk,
:soft_reset,
:pcd_config_reset,
:antenna_on,
:write_spi,
:to_yaml,
```

### 3. Codi

```
GNU nano 7.2
require 'mfr522'

lector = MFRC522.new

puts "Posa el teu RFID tag aprop del lector"
sleep(5)

begin
  lector.picc_request(MFRC522::PICC_REQA)
  uid = lector.picc_select

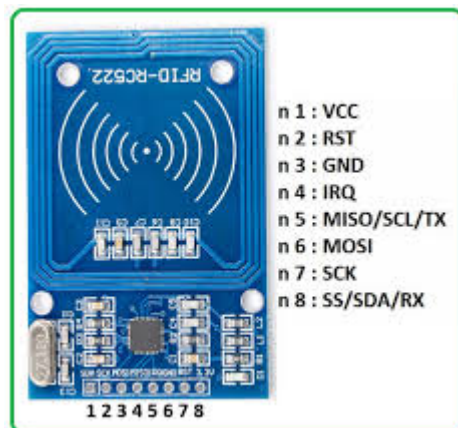
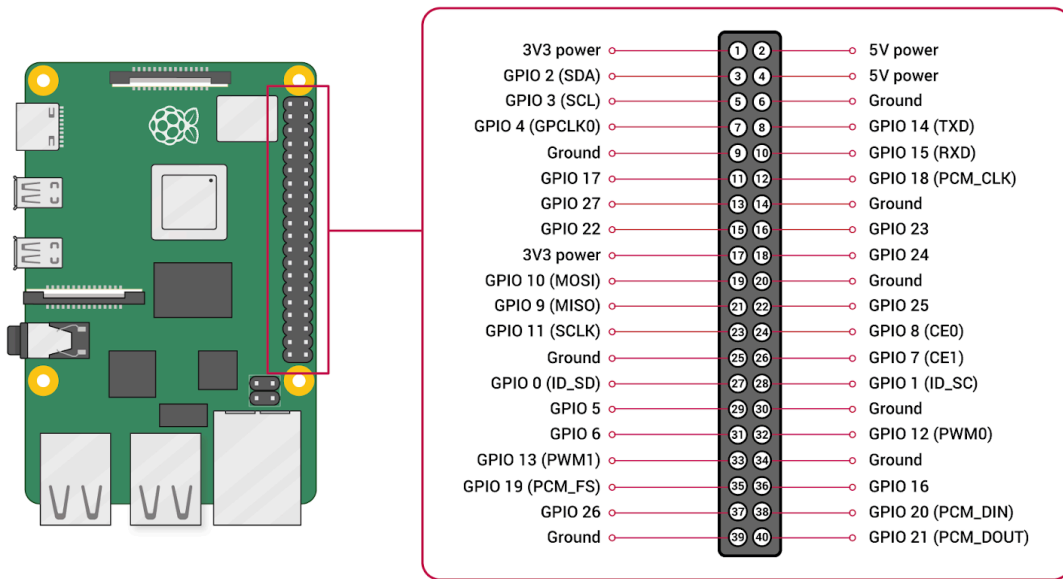
  if uid.is_a?(Array) && !uid.empty?
    uid_1 = uid[0]
    uid_hex = uid_1.map { |byte| "%02X" % byte }.join
    puts "Tag ID: #{uid_hex}"
  else
    puts "No s'ha detectat cap tag"
  end
rescue => e
  puts "Error: #{e.message}"
end
```

Com es pot observar, aquest codi primer de tot crea un objecte de la classe MFRC522 que permet controlar el perifèric. Després, utilitzant els mètodes `picc_request` i `picc_select`, llegeix l'ID del pin o targeta que s'apropi al lector. Llavors, verifica que hagi captat l'ID i si l'array no està buida mostra per pantalla el tag en format hexadecimal de 8 dígit, per altre banda, si l'array està buida mostra per pantalla "No s'ha detectat cap tag". Per acabar, tenint en compte que durant el procés de lectura poden haver-hi errors, capta l'error i mostra per pantalla un missatge de l'error.

I en executar el codi podem observar el següent:

```
pol@raspberrypi:~/Desktop $ ruby primer_puzzle.rb
Posa el teu RFID tag aprop del lector
Tag ID: 1394A714
```

#### 4. Com connectar la placa amb el RDIF-RC522



Pin Raspberrypi	Pin RFID-RC522
1 - 3V3 power	1 - VCC
22 - GPIO 25	2 - RST
6 - GND	3 - GND
Cap	4 - IRQ
21 - MISO	5 - MISO
19 - MOSI	6 - MOSI
23 - SCK	7 - SCK
24 - CEO	8 - SDA