





















Pacchetto su RF24: conversione dei programmi da seriale a RF24

(Parte Raspberry)

1. Attività preliminari

1.1. Piedinatura versione 3 e 4

Vedi anche <https://www.vincenzov.net/tutorial/RaspberryPi/connettore-GPIO.htm>

Raspberry Pi 3 GPIO Header				
Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I²C)		DC Power 5v	04
05	GPIO03 (SCL1 , I²C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I²C ID EEPROM)		(I²C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

Rev. 2
29/02/2016

www.element14.com/RaspberryPi

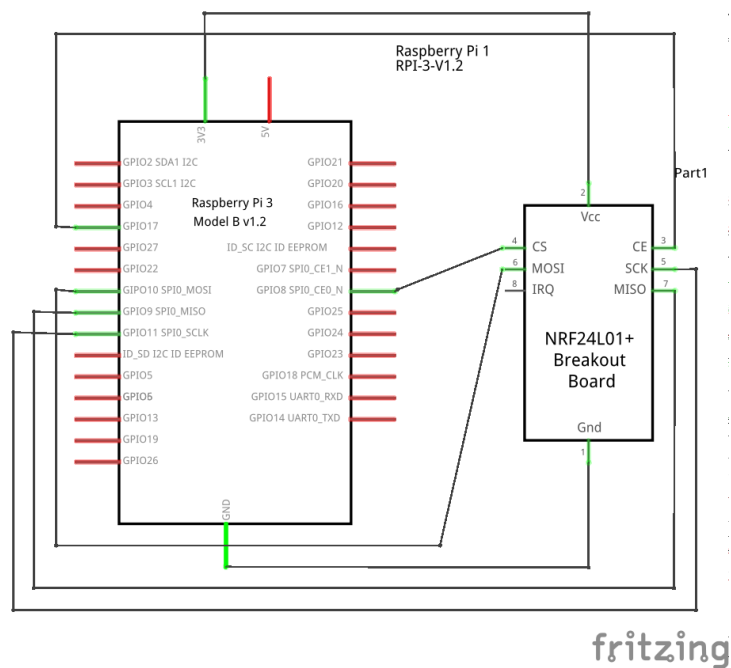
1.2. Uso dei piedini da shell

- Provare l'esempio del [video \(5:56\)](#)
- Provare lo script bash della sezione "Accendere un LED" al link <https://www.vincenzov.net/tutorial/RaspberryPi/helloREALworld-sh.htm>

1.3. Connessione con nRF24L01



nRF24L01	Pi (BCM Notation)	Pi (BOARD Notation)
Vcc	3.3v	Pin 1
GND	Ground	Pin 6
CSN	GPIO08	Pin 24
CE	GPIO17	Pin 11
MOSI	GPIO10	Pin 19
MISO	GPIO09	Pin 21
SCK	GPIO11	Pin 23



1.4. Installazione librerie

- **pigpiod**: è il processo di sistema che fa da interfaccia verso GPIO

```
sudo apt install pigpiod
```

- **nrf24**: è la libreria a che fa da interfaccia fra l'applicazione Python e nRF24L01

```
python3 -m pip install nrf24
```

2. Ricezione dati sensore

2.1. Libreria e costanti

```
import time
import sys
import struct

import pigpio
from nrf24 import *

#-----
# costanti
#-----
PIGPIONAME='localhost'
PIGPIOPORT=8888
READINGPIPE='00001'

MIO_ID=b"AB"
MIO_INDIRIZZO=b"P001"
MIO_TIPO=b"S1"
```

2.2. Settaggi e apertura pipe

```
# connessione a pigpiod
pi = pigpio.pi(PIGPIONAME, PIGPIOPORT)
if not pi.connected:
    print("Pigpiod non connesso. Lanciare: SUDO PIGPIOD")
    sys.exit()

# Crea l'oggetto NRF24
nrf = NRF24(pi, ce=17, payload_size=32, channel=76,
data_rate=RF24_DATA_RATE.RATE_1MBPS, pa_level=RF24_PA.LOW)

# apre la pipe
nrf.set_address_bytes(5)
nrf.open_reading_pipe(RF24_RX_ADDR.P1, READINGPIPE)
```

2.3. Lettura pacchetto

```
while True:
    if nrf.data_ready():
        msg=(struct.unpack("2s 4s 4s 2s 4s 16s",nrf.get_payload()))
        print(msg)
```

3. Invio comandi motore

3.1. Libreria e costanti

```
import time
import sys

import pigpio
from nrf24 import *
import struct

#-----
#  costanti
#-----
PIGPIONAME='localhost'
PIGPIOPORT=8888
WRITINGPIPE='00001'
ID=b"AB"
MITTENTE=b"P001"
DESTINATARIO=b"A328"
TIPO=b"A1"
VUOTO=("."*16).encode()
```

3.2. Settaggi e apertura pipe

```
# connessione a pigpiod
pi = pigpio.pi(PIGPIONAME, PIGPIOPORT)
if not pi.connected:
    print("Pigpiod non connesso. Lanciare: SUDO PIGPIOD")
    sys.exit()

# Crea l'oggetto NRF24
nrf = NRF24(pi, ce=17, payload_size=32, channel=76,
data_rate=RF24_DATA_RATE.RATE_1MBPS, pa_level=RF24_PA.LOW)

# apre le pipe
nrf.set_address_bytes(5)
nrf.open_writing_pipe(WRITINGPIPE)
```

3.3. Scrittura pacchetto

```
msg=struct.pack("2s 4s 4s 2s 1s 3s 16s",
                ID,MITTENTE,DESTINATARIO,TIPO,direzione,velocita,VUOTO)
nrf.send(msg)
print(msg)
```