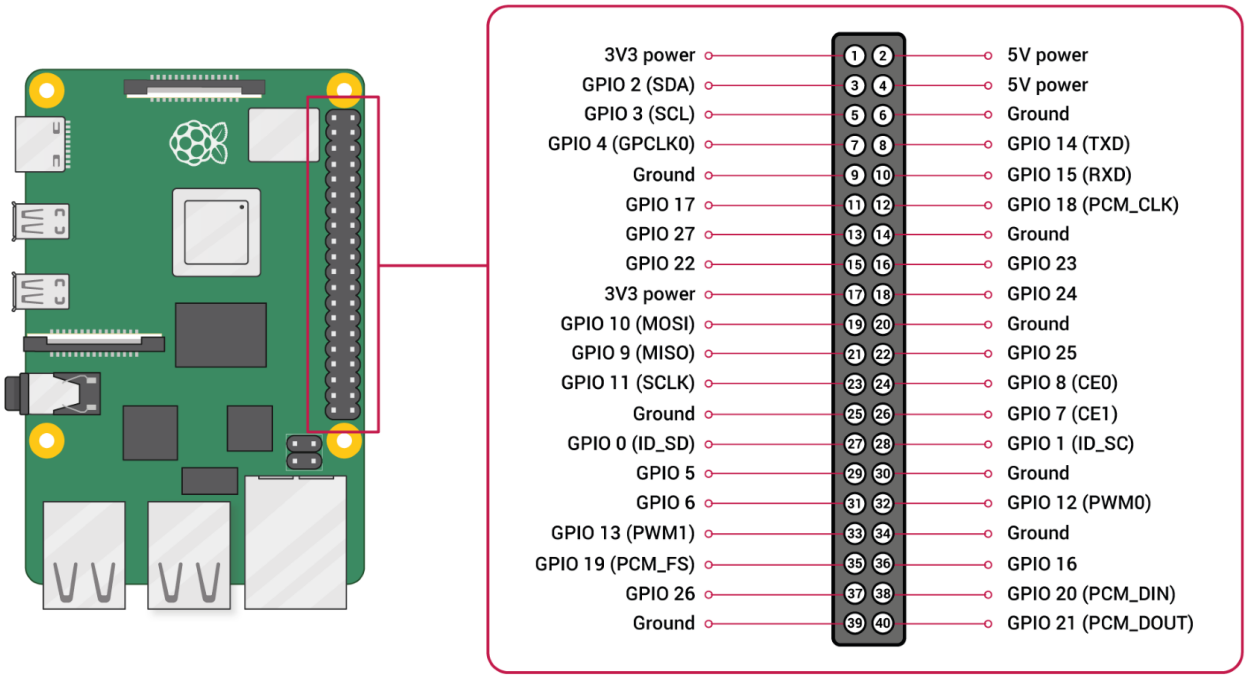
GPIO & Python sur Pi 4

GPIO & Python sur Pi 4

GPIO signifie General Purpose Input Output. Les broches GPIO du Raspberry Pi peuvent se configurer soit en entrée, soit en sortie. Certaines broches sont associées à des fonctions de type PWM, I²C, SPI, Serial, 1 wire, etc. Comparé à un microcontrôleur MicroChip, l'utilisateur ne rentrera pas dans les détails de configuration de registre bas-niveau (TRIS, LAT, PORT...) le driver s'occupe de cela. Le programmeur pourra se concentrer sur l'applicatif avec des fonctions de haut niveau pour piloter les GPIO.

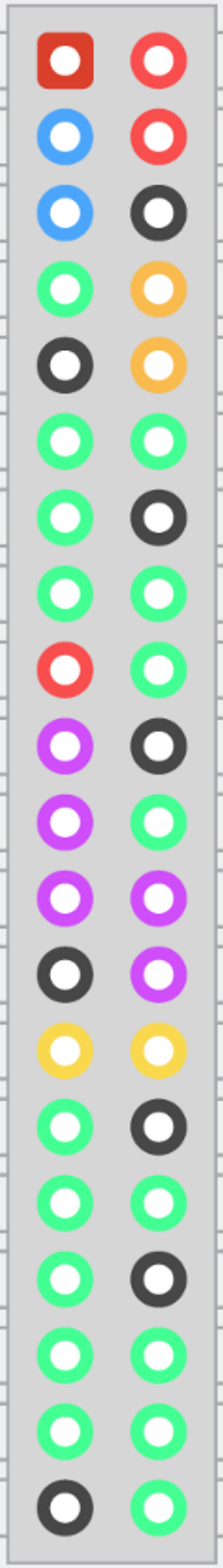


Attention

Les GPIO du Raspberry Pi sont compatibles avec des niveaux de tension 0-3.3V. Appliquer du 5V sur ces GPIO va les détruire. N'oubliez pas que le prix des Raspberry ont augmenté de manière importante depuis 2021 et que les délais de livraison sont de 6 mois.

Ci-dessous, une image plus détaillée des fonctions des broches du Raspberry Pi4

PIN	NAME		NAME	PIN
01	3.3V DC Power		5V DC Power	02
03	GPIO02 (SDA1,I²C)		5V DC Power	04
05	GPIO03 (SDL1,I²C)		Ground	06
07	GPIO04 (GPCLK0)		GPIO14 (TXD0, UART)	08
09	Ground		GPIO15 (RXD0, UART)	10
11	GPIO17		GPIO18(PWM0)	12
13	GPIO27		Ground	14
15	GPIO22		GPIO23	16
17	3.3V DC Power		GPIO24	18
19	GPIO10 (SP10_MOSI)		Ground	20
21	GPIO09 (SP10_MISO)		GPIO25	22
23	GPIO11 (SP10_CLK)		GPIO08 (SPI0_CE0_N)	24
25	Ground		GPIO07 (SPI0_CE1_N)	26
27	GPIO00 (SDA0, I²C)		GPIO07 (SCL0, I²C)	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12 (PWM0)	32
33	GPIO13 (PWM1)		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40



Dans un premier temps, nous allons configurer une des GPIO en Output pour faire clignoter une Led. Pour brancher la Led, il nous faut le signal de la GPIO:

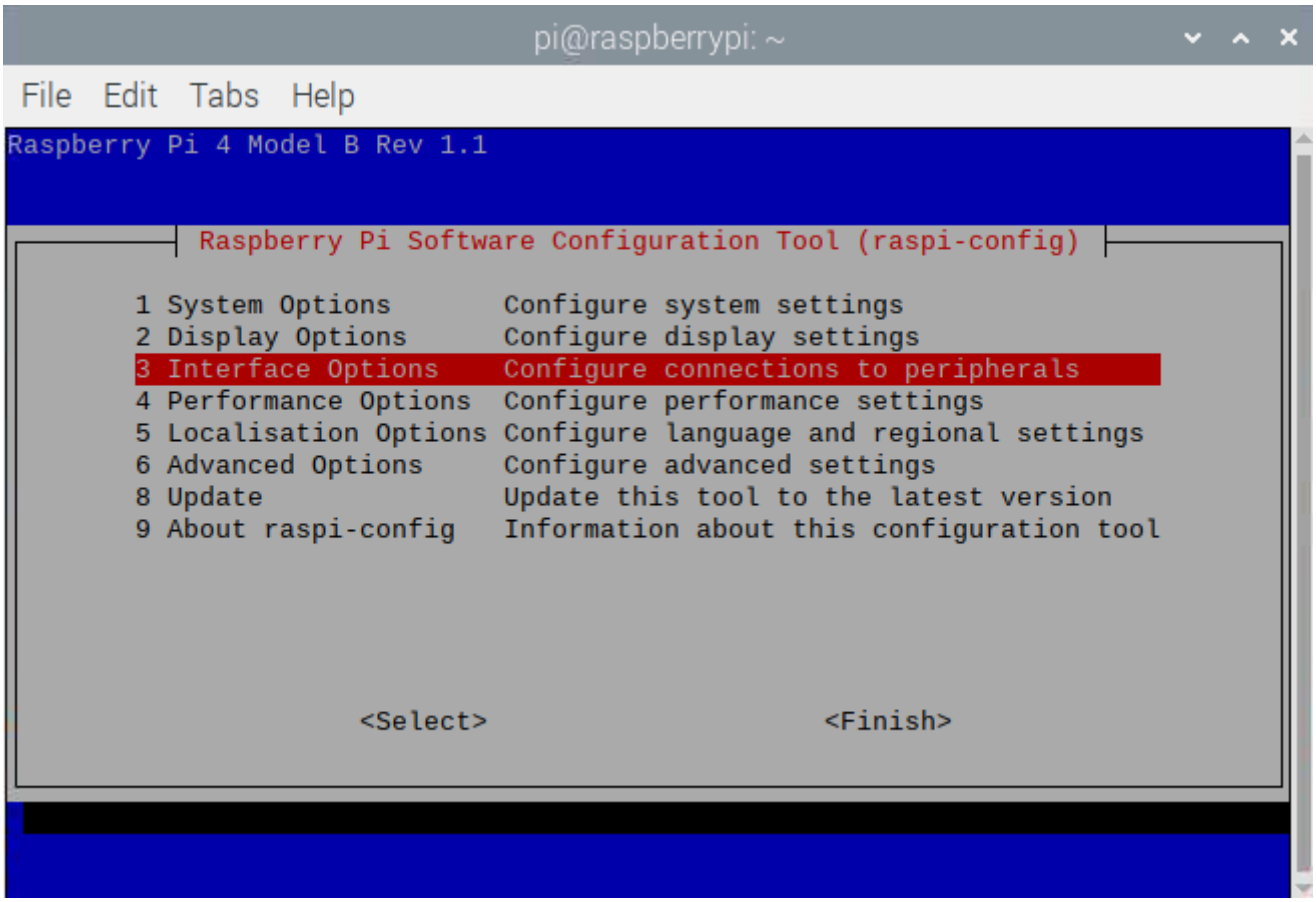
- GPIO 5 par exemple (broche 29)
- Le GND 0V (broche 6)

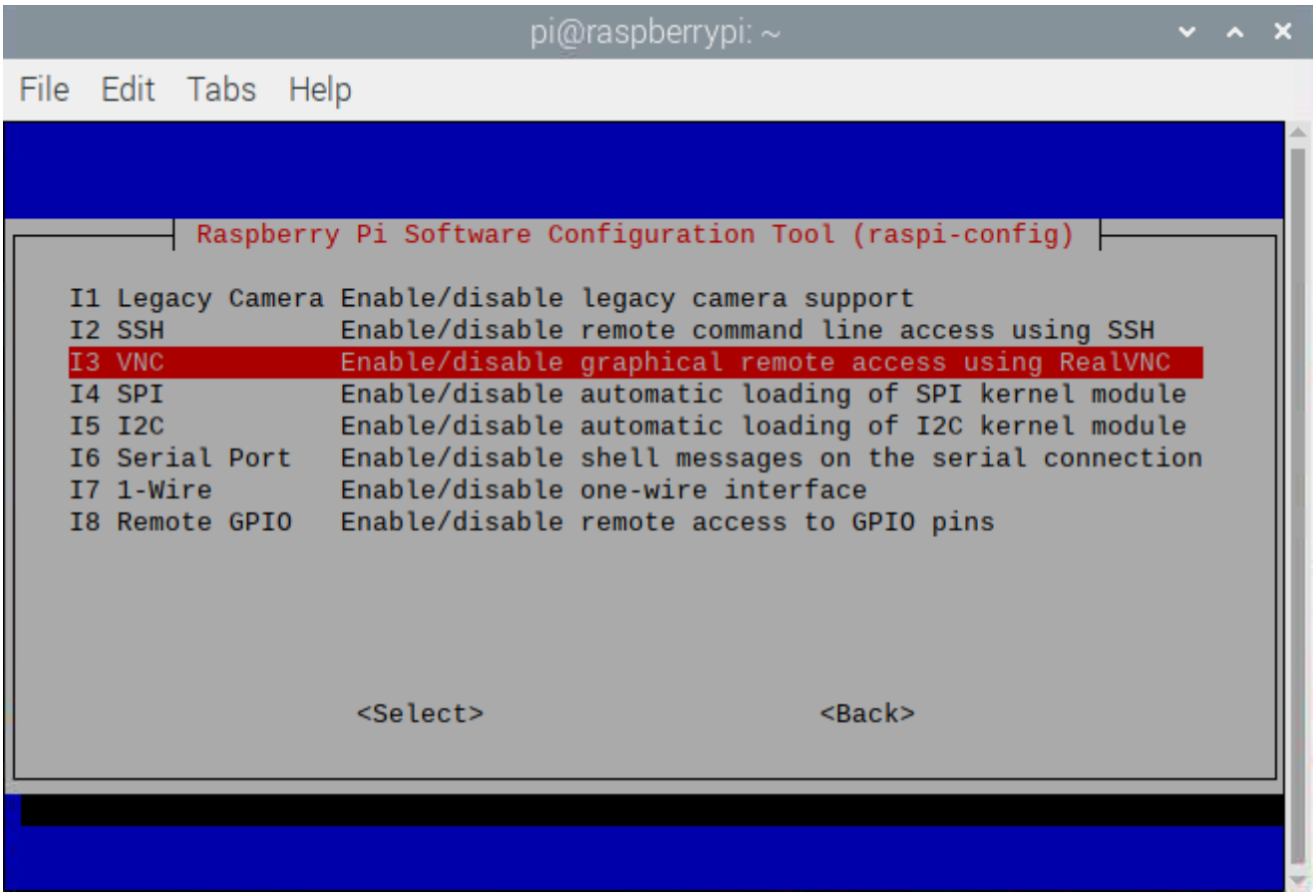
Validation GPIO avec Raspi-Config

Il est nécessaire que les sorties GPIO soit activées au niveau de la configuration du Raspberry Pi. Depuis un terminal ou une liaison SSH, lancer la commande :

```
sudo raspi-config
```

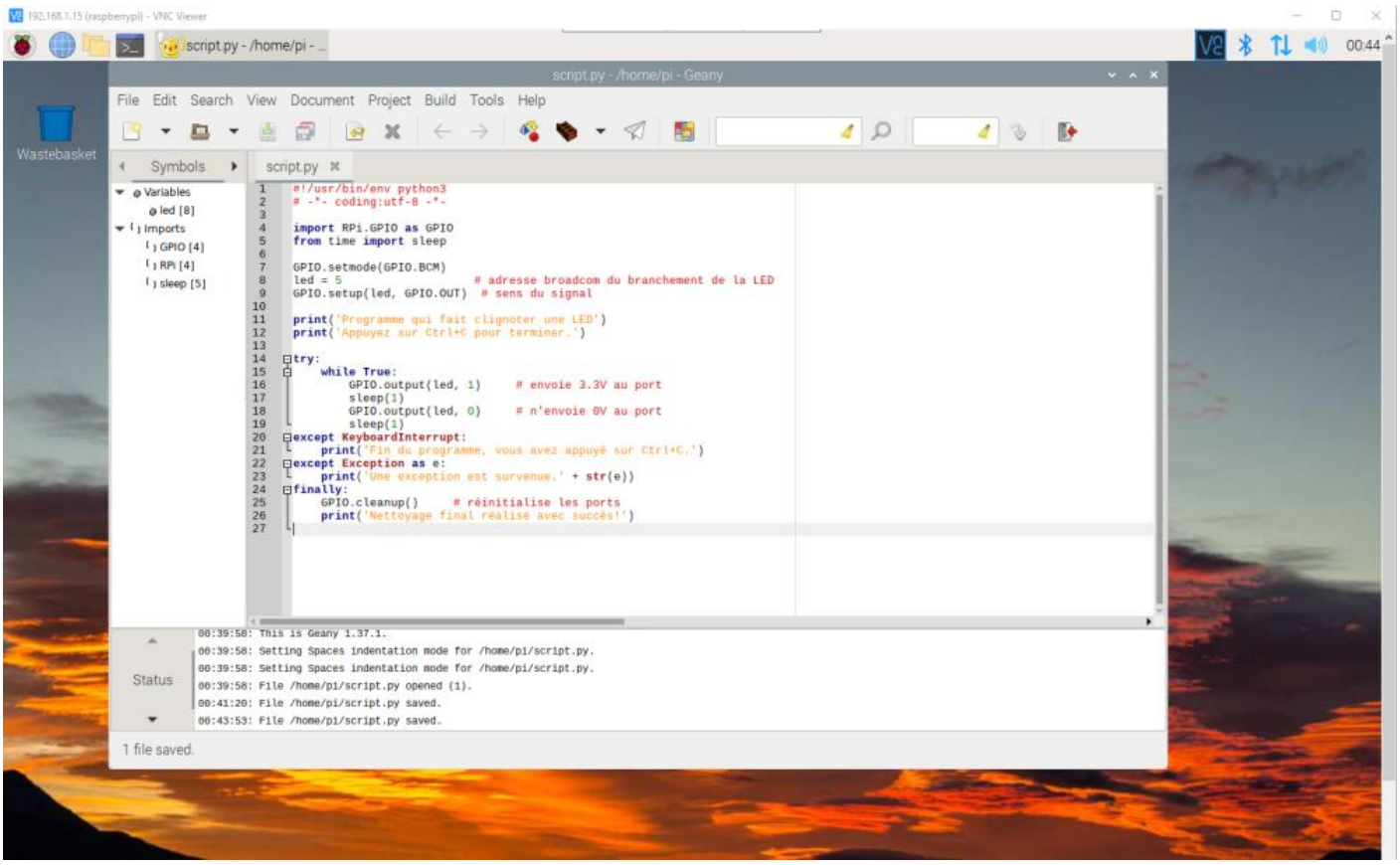
Dans Interface Options -> Activer **I8 Remote GPIO**





Programmation depuis VNC

On peut utiliser Geany comme éditeur de texte pour écrire le script Python.



Ci-dessous, un exemple de code pour faire clignoter une Led branchée sur GPIO5:

```
#! /usr/bin/env python3
# -*- coding: utf-8 -*-

import RPi.GPIO as GPIO
from time import sleep

GPIO.setmode(GPIO.BCM)
led = 5
GPIO.setup(led,GPIO.OUT)

print('It BLINKs')
print('Ctrl+C to quit')

try:
    while True:
        GPIO.output(led,1)   # envoie 3.3V au port
        sleep(1)
        GPIO.output(led,0)   # envoie 0V au port
        sleep(1)
except KeyboardInterrupt:
```

```
    print('End of')
except Exception as e:
    print('Exception occured' + str(e))
finally:
    GPIO.cleanup()           # init des ports GPIO
```

Pour rendre le script python exécutable, on remarque sur les deux premières lignes

- le SHA-BANG (SHE-BANG): #!/usr/bin/env python3
- l'encodage texte : *# -- coding:utf-8 --*

Un script Python n'est pas un fichier compilé, le SHA-BANG précise l'interpréteur (python3) qui va permettre d'exécuter le script.

Pour rendre le script exécutable, on utilise la commande:

```
chmod + x script.py
```

Ensuite, on peut exécuter le script en faisant ./script.py

```
./script.py
```

Limites de la lib RPI.GPIO

- Note that this module is unsuitable for real-time or timing critical applications. This is because you can not predict when Python will be busy garbage collecting. It also runs under the Linux kernel which is not suitable for real time applications - it is multitasking O/S and another process may be given priority over the CPU, causing jitter in your program. If you are after true real-time performance and predictability, buy yourself an Arduino http://www.arduino.cc !
- Note that the current release does not support SPI, I2C, hardware PWM or serial functionality on the RPi yet. This is planned for the near future - watch this space! One-wire functionality is also planned.
- Although hardware PWM is not available yet, software PWM is available to use on all channels.

La librairie RPI.GPIO est simple à utiliser et permet rapidement de piloter une sortie TOR ou de lire l'état d'une entrée TOR, par contre, les contraintes temporelles ne sont pas maîtrisées, càd par exemple, lorsque le code met la sortie à 1, l'action réelle de mise à 1 peut prendre un temps variable non maîtrisé.

- Il existe de nombreuses librairies ou méthodes pour piloter les GPIO, des plus simples à celles réservées aux experts. Tout va dépendre de l'application. Ces aspects temporels pourront être développés dans un prochain article.

Revision #1
Created 5 July 2023 14:09:33 by Philippe Celka
Updated 5 July 2023 14:12:12 by Philippe Celka