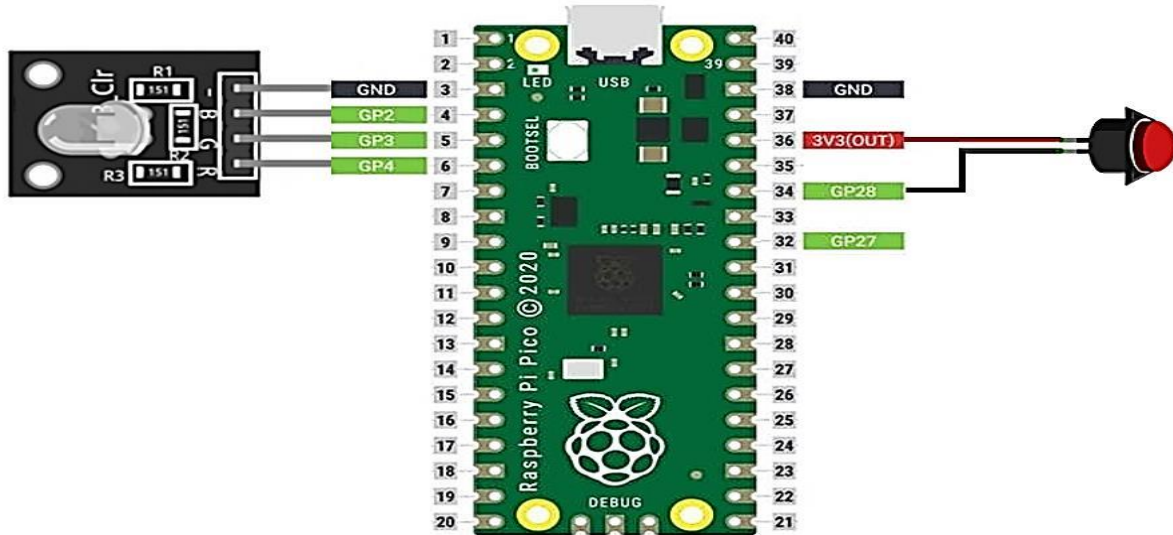


## Série de TD N°5: Application - Les boucles et les compteurs

### Exercice

Soit le circuit suivant composé d'une carte Raspberry Pi Pico, d'un module RGB et d'un bouton-poussoir.



### Partie 1 :

Le code suivant est un compteur de 0 à 9.

1. Remplacez la boucle "for" par une boucle "while".
2. Modifiez le code de manière à :
  - Allumer la LED en ROUGE (GP4) lorsque le compteur s'arrête.
  - Allumer la LED en VERT (GP3) pendant le comptage.
3. Modifiez le code pour démarrer le compteur avec le bouton-poussoir (GP28).
4. Donnez l'algorithme du dernier code

```
import time
x = 0

for y in range(0, 10):
    print(x)
    time.sleep(0.5)
    x += 1
```

### Partie 2 :

1. Expliquez le code suivant et donnez son algorithme.
2. Proposez un algorithme pour changer l'état de la LED RGB (3 cas possibles) à chaque fois que le bouton est pressé :
  - Couleur ROUGE si i=0
  - Couleur VERTE si i=1
  - Couleur BLEUE si i=2
3. Traduire l'algorithme en code MicroPython

```
import machine, time
from machine import Pin
led1 = Pin(4, Pin.OUT)
led2 = Pin(3, Pin.OUT)
button = Pin(28, Pin.IN, Pin.PULL_DOWN)
i=0

while True:
    if button.value():
        i=i+1

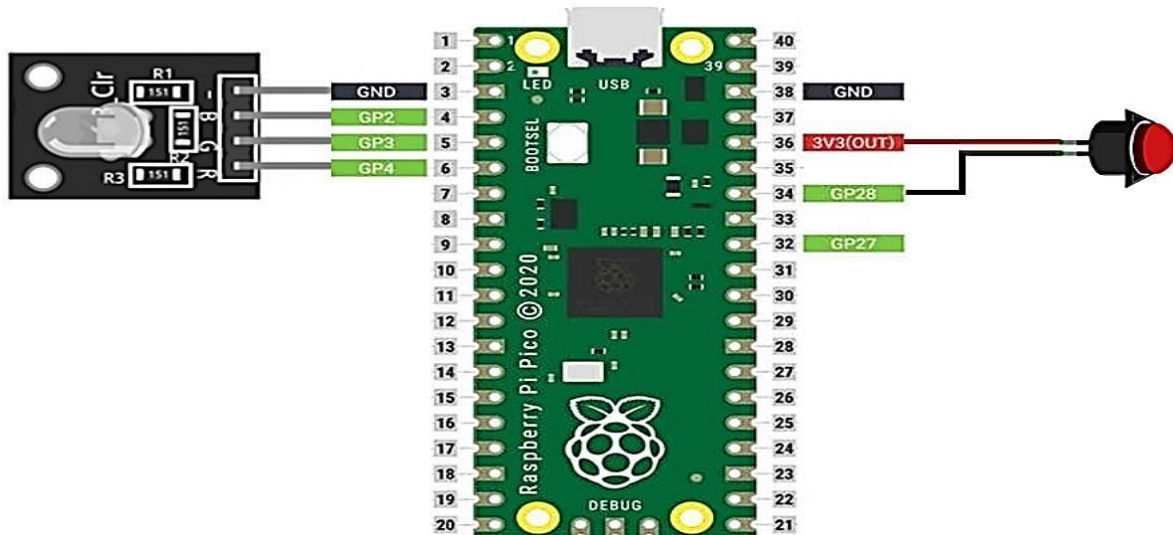
        if i==1:
            led1.on()
            led2.off()
        else:
            i=0
            led1.off()
            led2.on()

    print( "i:",i)
    time.sleep(0.5)
```

## Application - Loops and Counters

### Activity :

The following circuit is composed of a Raspberry Pi Pico board, an RGB module and a push button.



### Part 1 :

The next code is a 0 to 9 counter.

1. Replace for loop with while loop.
2. Modify the code in order to :
  - Turn on the LED in RED (GP4) when the counter stops
  - Turn on the LED in GREEN (GP3) while counting
3. Modify the code to start the counter with the push button (GP28)
4. Give the algorithm of the last code

```
import time
x = 0

for y in range(0, 10):
    print(x)
    time.sleep(0.5)
    x += 1
```

### Part 2 :

1. Explain the next code and give its algorithm
2. Suggest an algorithm to change the state of the RGB LED (3 possible cases) every time the button is pressed :
  - RED colour if i=0
  - GREEN colour if i=1
  - BLUE colour if i=2
3. Convert the algorithm into a MicroPython code

```
import machine, time
from machine import Pin
led1 = Pin(4, Pin.OUT)
led2 = Pin(3, Pin.OUT)
button = Pin(28, Pin.IN, Pin.PULL_DOWN)
i=0

while True:
    if button.value():
        i=i+1

        if i==1:
            led1.on()
            led2.off()
        else:
            i=0
            led1.off()
            led2.on()

    print("i:",i)
    time.sleep(0.5)
```