



Fiche Tutoriel 9

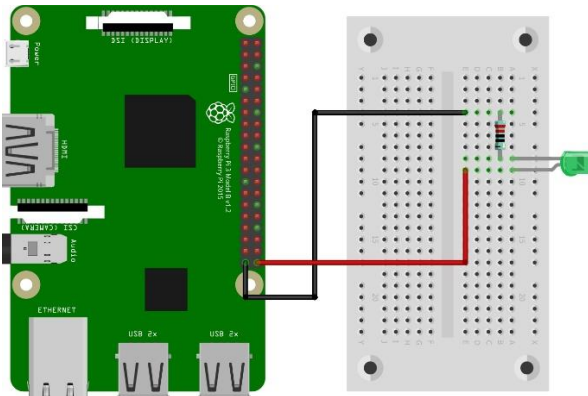
Raspberry – python

Exemple : allumer une LED externe

Dans cet Exemple nous allons faire exactement la même chose que dans la fiche pratique 8, mais nous allons utiliser la raspberry et un code python pour faire fonctionner la LED. Nous utiliserons également tout ce que nous avons vu précédemment à savoir Une Breadboard pour le circuit, une résistance de 220 Ω pour éviter de griller la LED, et utiliser la bibliothèque GPIO pour faire fonctionner le tout

Le montage du circuit.

Nous allons branché la cathode de la diode à la broche 21 de la rapsberry (fil rouge) et l’anode à l’une des masse (fil noir)



Le circuit est en place passons au code. Sous Raspbian cliquez sur menu framboise puis programmation et enfin Thonny python.

Le programme et ses étapes

Comme nous l’avons vu dans la fiche pratique 7 les étapes sont :

- Importer le bibliothèque GPIO pour utiliser les broches de la rapsberry
- Définir le mode de numérotation des broches BOARD ou BCM
- Définir la broche branchée sur annode de la LED comme une entrée
- Allumer la LED par lafonction HIGH

Taper le code suivant :

```
#!/usr/bin/python3
# chaque programme débutera par cette ligne elle permet de connaitre sous quel environnement
l'on doit lancer le programme

import RPi.GPIO as GPIO # on importe la bibliothèque GPIO et l'on créer un objet GPIO
GPIO.setmode(GPIO.BCM) #on définit la numérotation employée pour les broches
GPIO.setup(21,GPIO.OUT, initial=GPIO.LOW) # on déclare la pin 21 comme une sortie et à l'état bas
GPIO.output(21.GPIO.HIGH) #on met la broche 21 à haut pour allumer la LED
```

Sauvegarder ensuite votre programme dans le dossier home/py.



Pour le lancer deux solutions s'offrent à vous, soit par Thonny python en cliquant sur le bouton Run en haut de l'IDE.

Soit vous ouvrez la console de ligne de commande (cf fiche pratique 3), placer vous dans le dossier où est situé le programme par la commande

```
cd /home/pi, puis taper ensuite
```

```
sudo python3 <nom de votre programme avec l'extension py>
```

Voilà le programme fonctionne et la LED est allumée, mais il ne fait rien de plus. Nous allons donc la faire « blinker » pour mieux comprendre et voir le programme fonctionner comme dans la fiche 8 pour l'arduino. Nous allons allumer la LED puis l'éteindre pendant en certain temps et répéter ce code à l'infini par une boucle while.

Le code devient donc :

```
#!/usr/bin/python3
# chaque programme débutera par cette ligne elle permet de connaitre sous quel environnement
l'on doit lancer le programme

import RPi.GPIO as GPIO # on importe la bibliothèque GPIO et l'on créer un objet GPIO
import time #on importe la bibliothèque gérant le temps

GPIO.setmode(GPIO.BCM) #on définit la numérotation employée pour les broches
GPIO.setup(21,GPIO.OUT, initial=GPIO.LOW) # on déclare la pin 21 comme une sortie et à l'état bas

while true : #on définit une boucle infinie puisque la condition est toujours vrais
    GPIO.output(21,GPIO.HIGH) #on met la broche 21 à haut pour allumer la LED
    time.sleep(1) #on attend 1 seconde
    GPIO.output(21,GPIO.LOW) #on met la broche 21 à bas pour éteindre la LED
    time.sleep(1) #on attend 1 seconde
```

La LED clignote maintenant toutes les secondes. On peut simplifier le code comme d écrit dans la fiche pratique 7 en inversant l'état de la broche de sortie ce qui simplifie énormément le code

```
#!/usr/bin/python3
# chaque programme débutera par cette ligne elle permet de connaitre sous quel environnement
l'on doit lancer le programme

import RPi.GPIO as GPIO # on importe la bibliothèque GPIO et l'on créer un objet GPIO
import time #on importe la bibliothèque gérant le temps

GPIO.setmode(GPIO.BCM) #on définit la numérotation employée pour les broches
GPIO.setup(21,GPIO.OUT, initial=GPIO.LOW) # on déclare la pin 21 comme une sortie et à l'état bas

while true : #on définit une boucle infinie puisque la condition est toujours vrais
    GPIO.output(21, not GPIO.input(21)) #on inverse l'état de la broche 21
    time.sleep(1) #on attend 1 seconde
```

Si vous avez compris vous pouvez maintenant tenter de faire l'exercice de la fiche travaux pratique 2