

Raspberry : faire varier la puissance électrique avec PWM

Ressources

- Brochage du GPIO et des cartes HAT : <https://fr.pinout.xyz/>
- Article sur PWM : <https://mespotesgeek.fr/fr/variation-de-puissance-electrique-via-raspberry/>
- Document PDF sur PWM : [4346-5-rpi-pwm.pdf](#)

Présentation

Pour **contrôler l'intensité** lumineuse d'une LED ou la **vitesse** d'un moteur en utilisant les sorties GPIO du Raspberry, il faudrait pouvoir **gérer la puissance électrique** disponible en faisant varier le courant ou l'intensité.

Cela n'est pas possible car la tension de sortie est de **3,3 V** à l'état haut et de **0 V** (la masse) à l'état bas. La solution est de **d'alterner ces deux états de façon répétée** : cette technique se nomme le **PWM** pour Pulse Width Modulation (modulation de largeur d'impulsion).

Principe de fonctionnement

Le principe général consiste à appliquer une **succession d'impulsion** (passage de l'état 0 à 1) pendant des durées bien choisies afin d'obtenir, en **moyenne** sur une certaine durée, **plusieurs valeurs intermédiaires** au lieu d'avoir uniquement deux valeurs, tout ou rien.

Pour, il faut gérer deux paramètres:

- La fréquence (période),
- et le rapport cyclique

Fréquence

La fréquence (ou période) est le nombre d'impulsion générées par seconde et est mesurée en Hertz (Hz). Par exemple 50 Hz correspond à 50 impulsions en une seconde soit une impulsion toute les 0.02 secondes.

Le rapport cyclique

Le rapport cyclique est le pourcentage de temps ou le signal est en position 1 pendant le cycle. Par exemple pour une fréquence de 50 Hz et un rapport cyclique de 80% cela signifie que toutes les 0.02 secondes le signal sera à 1 (high) pendant 80% du temps du cycle de 0.02 seconde et à 0 (LOW) pendant les autres 20% du temps.

Gérer l'éclairage d'une LED

En t'aidant des documents donnés en ressource, réalise :

- le **câblage** d'une LED sur un des ports du GPIO du Raspberry,
- un programme qui donne **l'impression** de faire **varier** plusieurs fois l'intensité lumineuse de la LED.

Éléments de réponse

```
import RPi.GPIO as GPIO
import time

# Utiliser la numérotation électronique du GPIO
GPIO.setmode(GPIO.BCM)

# choisir le numéro de broche du GPIO à utiliser en sortie
GPIO.setup(4, GPIO.OUT)
```

```
# creation d'un objet PWM appelé LED en précisant le numero de broche (4) et la frequence (50Hz)
LED = GPIO.PWM(4, 50)

# demarrage du PWM avec un cycle a 0 (LED off) : LED éteinte
LED.start(0)

# Simuler un éclairage de la LED à la moitié de sa puissance lumineuse = définir le rapport cyclique à 50
LED.ChangeDutyCycle(50)

# attendre 2 secondes
time.sleep(0.01)

# éteindre la LED
LED.ChangeDutyCycle(0)

# Arrêter le PWM
LED.stop()

# libérer le port du GPIO utilisé
GPIO.cleanup()
```

Les activités ...

[Je reviens à la liste des activités.](#)

From:

/ - **Les cours du BTS SIO**

Permanent link:

[/doku.php/isn/gpio_varpuissance](#)

Last update: **2018/04/24 15:16**

