

connecté via 4k7Ω, Vgs sera toujours au-delà de 3,0V avec le bouton enfoncé si la sortie GPIO est à l'état haut.

Les 2N7000 sont disponibles pour 10cts. D'autres superbes composants sont maintenant disponibles. Certains comme le 2SK4043LS peuvent commuter des courants de 80A avec un petit Vgs de 2,5V. Un simple transistor commandé par le PI ne pourrait jamais faire ça. Le 2SK3018 est un CMS (composant monté en surface) conçu pour des conditions de Vgs de faible valeur comme ici avec le PI.

```
#test d'une entrée avec indication visuelle
import RPi.GPIO as GPIO
import time

# utilisation de la numérotation BCM du GPIO
GPIO.setmode(GPIO.BCM)

# (pull_up_down peut être PUD_OFF, PUD_UP ou
# PUD_DOWN, PUD_OFF par défaut)
GPIO.setup(4, GPIO.IN,
pull_up_down=GPIO.PUD_UP)

# teste si la broche peut atteindre
# l'état haut
if GPIO.input(4):
    print ('Entrée à l'état haut - Correct')
    time.sleep(0.2)
    GPIO.setup(4, GPIO.IN,
        pull_up_down=GPIO.PUD_DOWN)
else:
    print ('Entrée en erreur - broche 4')
    time.sleep(1)
    quit()

# teste si l'entrée peut atteindre l'état bas
if GPIO.input(4):
    print ('Entrée en erreur - broche 4')
    time.sleep(1)
    quit()
else:
    print ('Entrée à l'état bas - Correct')
    time.sleep(1)

# si on arrive ici, les états haut et bas
# des entrées sont atteignables
print ('Test des entrées - Correct')

# début de la démo avec le bouton
print ('Appuyez sur le bouton')

while True:
    # positionner l'entrée à l'état haut et
    # attendre un appui sur le bouton
    GPIO.setup(4, GPIO.IN,
        pull_up_down=GPIO.PUD_UP)
```

```
# bouton enfoncé
if not GPIO.input(4):
    print ('bouton enfoncé')
    time.sleep(1)

# bouton relâché
if GPIO.input(4):
    print ('bouton relâché')
    flash = 20
    GPIO.setup(4, GPIO.OUT)

# clignoter jusqu'à
while flash > 0:
    GPIO.output(4, True)
    time.sleep(0.1)
    GPIO.output(4, False)
    time.sleep(0.1)
    flash -= 1

print ('appuyez sur le bouton')
```

Il y a beaucoup à dire sur le FET dans cette application.

Clive Tombs

Note de l'éditeur

Nous aimons recevoir le courrier de nos lecteurs. Si vous avez un commentaire sur un article, ou une astuce en rapport avec le Raspberry Pi à partager, s'il vous plaît envoyez-les nous, et nous essaierons de les prendre en compte dans un prochain numéro.

LE SAVIEZ VOUS ?

La série d'articles "Sous contrôle" des numéros 2, 3 et 4 est excellente pour débiter et apprendre à utiliser les GPIO. Si vous n'avez pas encore commencé, mais que vous voulez vous lancer, il y a eu quelques mises à jour de la bibliothèque Python RasPi GPIO requise avant de commencer.

1) La bibliothèque RasPi GPIO peut maintenant être facilement installée avec :

```
$sudo apt-get install python-rpi.gpio
ou
$sudo apt-get install python3-rpi.gpio
```

2) Ajoutez les lignes suivantes à chaque programme :

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
```