

TP3 Raspberry PI

Utilisation des GPIO : Le bus I2C

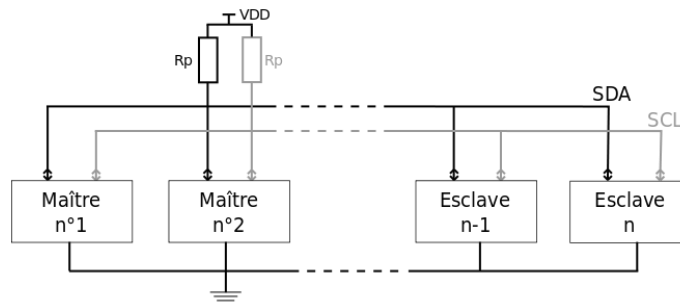
Objectif :

Le but du TP est de réaliser un thermomètre numérique à l'aide d'un capteur de température I2C, le TMP102, et le Raspberry

Rappel sur le bus I2C

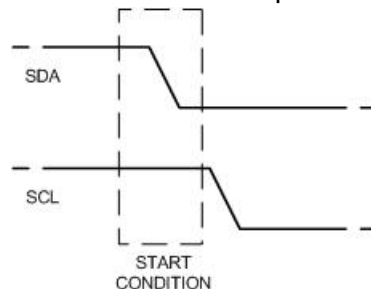
Le bus I2C est un bus local permettant les échanges séries à courte distance entre un microcontrôleur et des périphériques (ADC, DAC, afficheur, mémoire, capteurs etc ...) ou un autre microcontrôleur, les échanges sont gérés par protocole avec adressage.

Le bus I2C est synchrone et half-duplex, sa technologie d'interfaces drains-ouverts le rend très résistant aux courts-circuits et permet une gestion simple des erreurs.

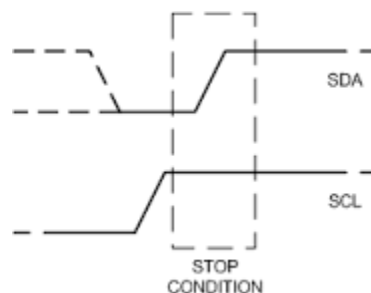


Un échange de données est toujours initié par le maître (ici Raspberry Pi).

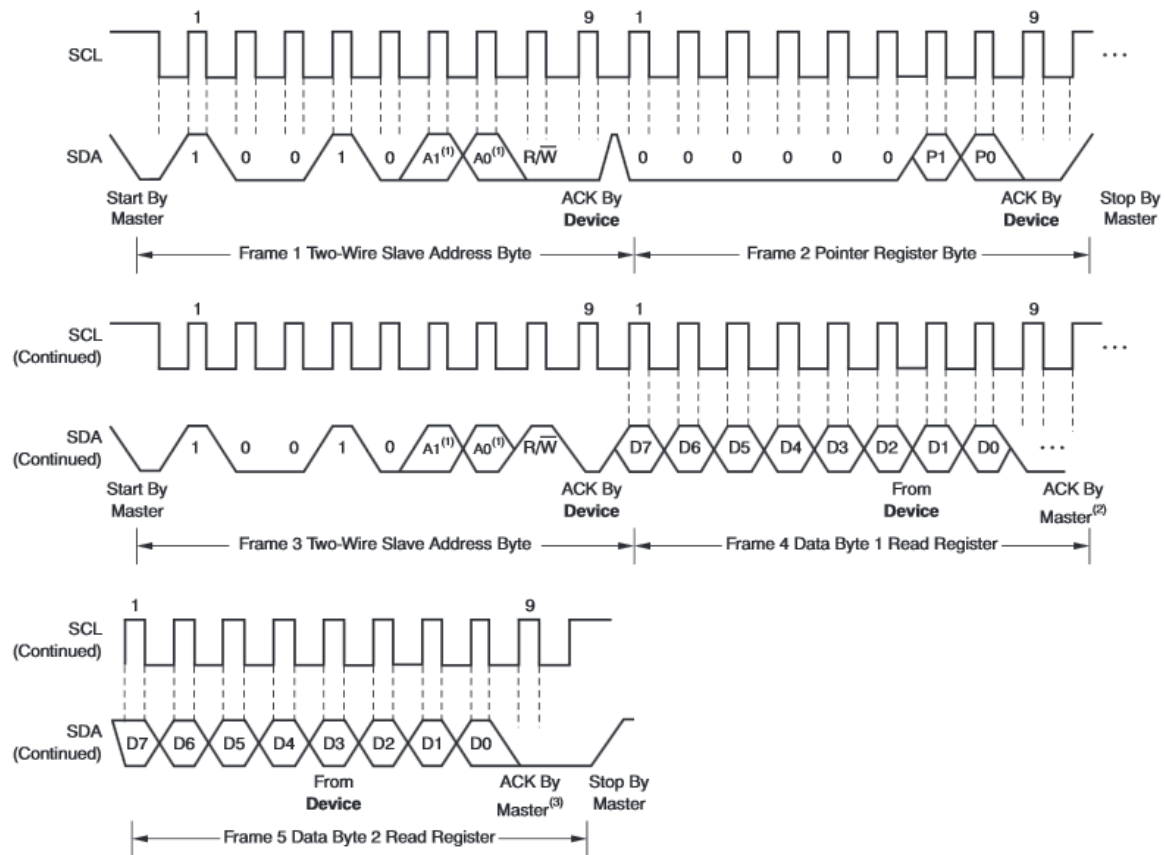
- a. Il commence par une Start Condition : SDA passe à 0 puis SCL passe à 0 :



- b. Puis l'adresse du destinataire : impaire pour une écriture (LSB R/W=1), paire pour une lecture (LSB R/W=0).
- c. Acquiescement par le destinataire qui place un niveau bas sur la 9^{ième} impulsion sur SCL.
Le sens de l'échange des données dépend de l'application, lecture pour un capteur.
- d. L'échange se termine par une condition stop, sda passe à 1 puis SCL passe à 1

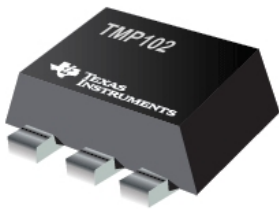


Exemple de trame I2C, lecture d'un capteur TMP102



I. Réalisation du montage

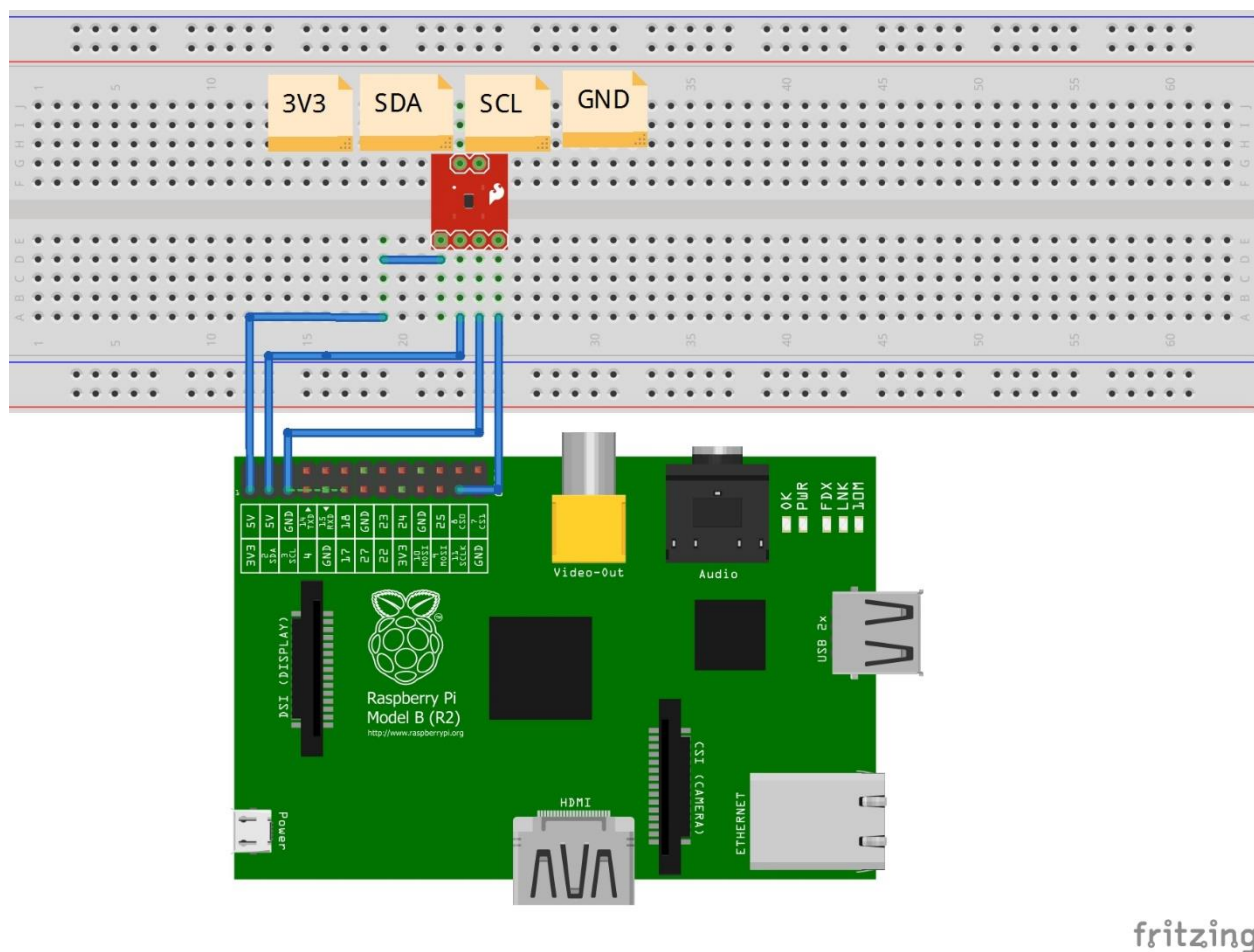
Le TP met en œuvre le capteur Texas Instrument TMP102 : <http://www.ti.com/product/tmp102>



Ce composant disponible en boîtier SOT563 est difficilement implantable sur circuit imprimé sans l'aide de machine de positionnement. Sparkfun propose une petite carte équipée d'un TMP102, des résistances de pull-up du bus I2C et d'un condensateur de découplage : <https://www.sparkfun.com/products/11931>

Le Raspberry Pi dispose d'une interface I2C et d'une bibliothèque. broche 3 : SDA, broche 5 : SCL (les résistances de pull-up sont déjà sur le module Sparkfun)

Réaliser le montage suivant



II. Programmation d'un thermomètre

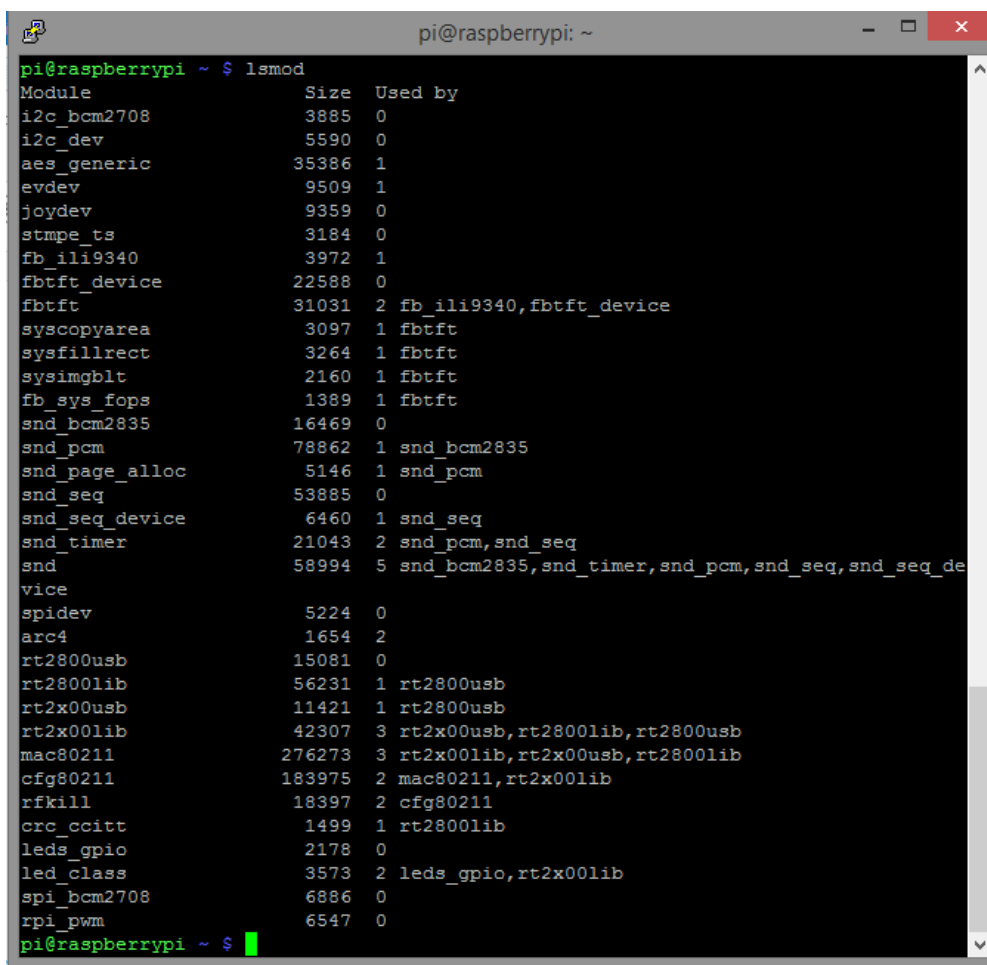
1. Afin d'installer le paquet **i2c-tools** et la bibliothèque **python-smbus**, taper les commandes suivantes :

```
apt-get update
sudo apt-get install python-smbus
sudo apt-get install i2c-tools
```

`modprobe` permet d'ajouter des modules dans le Noyau Linux, comme le module I2C.

2. Taper les commandes suivantes :

```
sudo modprobe i2c-dev
sudo modprobe i2c-bcm2708
modinfo nom_du_module → retourne les informations du module nom_du_module
```



```
pi@raspberrypi ~ $ lsmod
Module                  Size      Used by
i2c_bcm2708             3885      0
i2c_dev                 5590      0
aes_generic             35386     1
evdev                   9509      1
joydev                  9359      0
stmpe_ts                3184      0
fb_ili9340              3972      1
fbtft_device            22588      0
fbtft                   31031     2 fb_ili9340,fbtft_device
syscopyarea             3097      1 fbtft
sysfillrect             3264      1 fbtft
sysimgblt               2160      1 fbtft
fb_sys_fops             1389      1 fbtft
snd_bcm2835             16469      0
snd_pcm                 78862     1 snd_bcm2835
snd_page_alloc          5146      1 snd_pcm
snd_seq                 53885      0
snd_seq_device          6460      1 snd_seq
snd_timer               21043     2 snd_pcm,snd_seq
snd                     58994     5 snd_bcm2835,snd_timer,snd_pcm,snd_seq,snd_seq_de
vice
spidev                  5224      0
arc4                    1654      2
rt2800usb               15081      0
rt2800lib               56231     1 rt2800usb
rt2x00usb               11421     1 rt2800usb
rt2x00lib               42307     3 rt2x00usb,rt2800lib,rt2800usb
mac80211                276273     3 rt2x00lib,rt2x00usb,rt2800lib
cfg80211               183975     2 mac80211,rt2x00lib
rfkill                  18397     2 cfg80211
crc_ccitt               1499      1 rt2800lib
leds_gpio               2178      0
led_class               3573     2 leds_gpio,rt2x00lib
spi_bcm2708             6886      0
rpi_pwm                 6547      0
pi@raspberrypi ~ $
```

3. Editer le fichier **raspi-blacklist.conf** en tapant la commande :

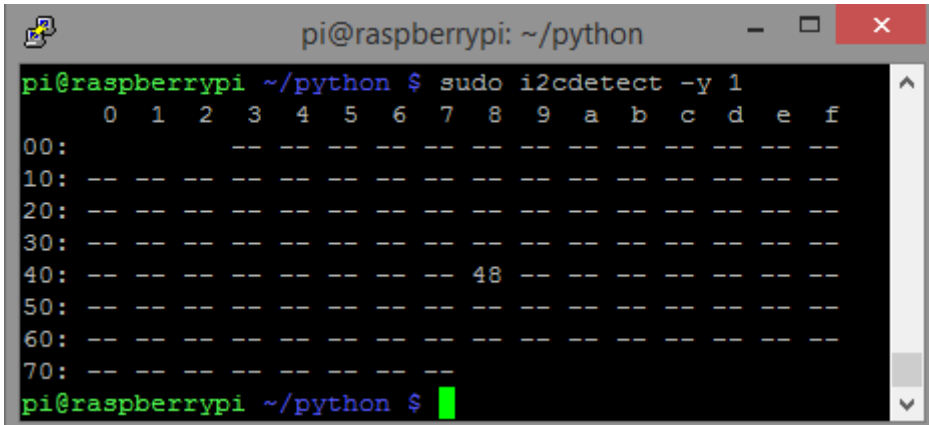
```
sudo nano /etc/modprobe.d/raspi-blacklist.conf
```

4. Si ce n'est pas le cas, ajouter des dièse(#) devant chacune des deux lignes suivantes :

```
spi-bcm2708
i2c-bcm2708
```

5. Afin de détecter les périphériques I2C connectés au Raspberry, taper la commande suivante :

```
sudo i2cdetect -y 1
```



```
pi@raspberrypi: ~/python
pi@raspberrypi ~/python $ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  48  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi ~/python $
```

6. Redémarrer le Raspberry pi.

Programme de gestion du capteur I2C tmp102 en python

```
#!/usr/bin/env python
import smbus
import time

bus_pi = smbus.SMBus(1)

# adresse I2C du capteur TMP102
addr = 0x48

while True:
    try:
        # x contient les 2 premiers octets transmis par le TMP102

        x = bus_pi.read_word_data(addr,0)
        msb=x>>8
        lsb=x&0x00FF
        wtemp=((lsb<<8) | msb)>>4

        # affichage de addr en hexa sur 2 chiffres et wtemp en hexa sur 4 chiffres

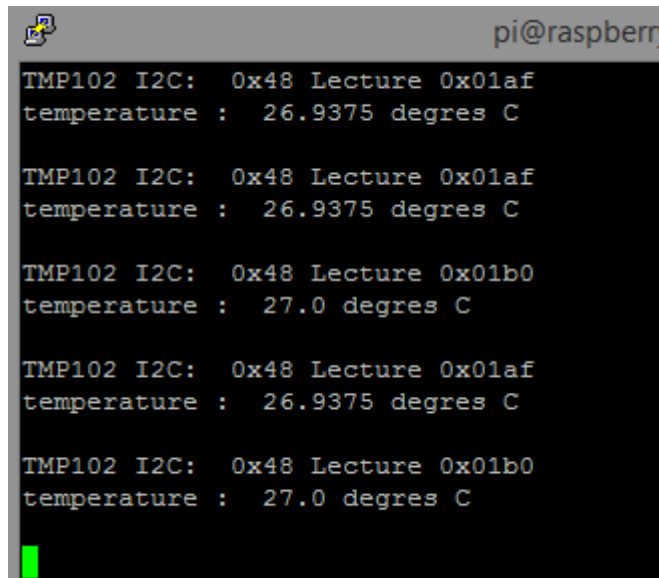
        print 'TMP102 I2C: ',hex(addr), 'Lecture ', hex(wtemp),'\n\r'
        print 'temperature : ',wtemp*0.0625 , 'degres C\n\r'

        # tempo de 0.5s avant une nouvelle lecture

        time.sleep(0.5)
    except:
        print '...erreur...'
        break
```

1. Lancer l'éditeur « nano » en tapant la commande : `sudo nano`
2. Saisir le programme ci-dessus
3. Pour enregistrer **ctrl+x** et entrer le nom de fichier **TMP102.py**. Ensuite valider en tapant sur **Y**.
4. Pour tester votre programme, taper la commande `sudo python TMP102.py`

Résultat attendu :



```
pi@raspberr
TMP102 I2C: 0x48 Lecture 0x01af
temperature : 26.9375 degrees C

TMP102 I2C: 0x48 Lecture 0x01af
temperature : 26.9375 degrees C

TMP102 I2C: 0x48 Lecture 0x01b0
temperature : 27.0 degrees C

TMP102 I2C: 0x48 Lecture 0x01af
temperature : 26.9375 degrees C

TMP102 I2C: 0x48 Lecture 0x01b0
temperature : 27.0 degrees C
```

5. Pour sortir du programme, taper sur ctrl+c.