

# Capteur Niveau D'eau Dans Reservoir (NUMERIQUE)

## Code:

```
import spidev
import time

# Configuration SPI
spi = spidev.SpiDev()
spi.open(0, 0)      # (bus=0, device=0) pour CEO
spi.max_speed_hz = 1000000 # Débit en Hz (1 MHz)

def lire_canal(channel):
    # Envoi de la commande SPI pour le MCP3008
    adc = spi.xfer2([1, (8 + channel) << 4, 0])
    data = ((adc[1] & 3) << 8) + adc[2]
    return data

try:
    print("Lecture du capteur de niveau d'eau (Ctrl+C pour arrêter)")
    print("-----")

    # Calibration (à ajuster)
    VALEUR_MIN = 100 # Valeur quand vide
    VALEUR_MAX = 950 # Valeur quand plein

    while True:
        valeur_brute = lire_canal(0) # Lecture du CH0
        pourcentage = (valeur_brute - VALEUR_MIN) / (VALEUR_MAX - VALEUR_MIN) * 100
        pourcentage = max(0, min(100, round(pourcentage, 1)))
```

```
print(f"Niveau: {pourcentage}% | Valeur brute: {valeur_brute}")

time.sleep(1)

except KeyboardInterrupt:

    print("\nArrêt du programme")

    spi.close()
```

### **Les Bibliothèques :**

```
sudo apt update
```

```
sudo apt install git python3-dev python3-setuptools

git clone https://github.com/doceme/py-spidev

cd py-spidev

sudo python3 setup.py install

python3 -c "import spidev; print(spidev.__version__)"

sudo nano tr. py

sudo nano tr. py - l

python3 tr. Py
```

## Câblage :

### Branchement entre le MCP3008 et la Raspberry Pi 4

| MCP3008          | RASPBERRY PI 4 (GPIO) |
|------------------|-----------------------|
| VDD (PIN 16)     | 3.3V (Alimentation)   |
| VREF (PIN 15)    | 3.3V (Référence)      |
| AGND (PIN 14)    | GND (Masse)           |
| CLK (PIN 13)     | GPIO 11 (SCLK)        |
| DOUT (PIN 12)    | GPIO 9 (MISO)         |
| DIN (PIN 11)     | GPIO 10 (MOSI)        |
| CS/SHDN (PIN 10) | GPIO 8 (CEO)          |
| DGND (PIN 9)     | GND (Masse)           |

### Branchement du capteur de niveau d'eau

| CAPTEUR                 | MCP3008                |
|-------------------------|------------------------|
| VCC (3.3V OU 5V)        | 3.3V (Broche 1 du Pi)  |
| GND                     | GND (Broche 6 du Pi)   |
| SIG (SIGNAL ANALOGIQUE) | CH0 (Pin 1 du MCP3008) |

## Avec Affichage Sur Ecran LCD

**Code :**

```
import sys
import time
import spidev

# Ajouter le chemin vers la bibliothèque LCD
sys.path.append("/chemin/absolu/vers/cd_lcd") # Remplacez par le vrai chemin
import drivers

# Initialisation du LCD
display = drivers.Lcd()

# Configuration SPI (votre code existant)
spi = spidev.SpiDev()
spi.open(0, 0)
spi.max_speed_hz = 1000000

def lire_canal(channel):
    adc = spi.xfer2([1, (8 + channel) << 4, 0])
    return ((adc[1] & 3) << 8) + adc[2]

try:
    # Calibration
    VALEUR_MIN = 100
    VALEUR_MAX = 950

    while True:
        valeur_brute = lire_canal(0)
        pourcentage = (valeur_brute - VALEUR_MIN) / (VALEUR_MAX - VALEUR_MIN) * 100
```

```
pourcentage = max(0, min(100, round(pourcentage, 1)))  
  
# Affichage terminal  
print(f"Niveau: {pourcentage}% | Valeur brute: {valeur_brute}")  
  
# Affichage LCD  
display.lcd_display_string(f"Niveau: {pourcentage}%", 1)  
display.lcd_display_string(f"Raw: {valeur_brute}", 2)  
  
time.sleep(1)  
  
except KeyboardInterrupt:  
    spi.close()  
    display.lcd_clear()
```

câblage :

