



# Bibliothèques - Sparkfun TMP102, TMP117

Mise à jour le [18/8/2023]



## Ressources

- **Description** en français sur Github [lien](#)
- **Guide** sur le site Sparkfun [lien](#).

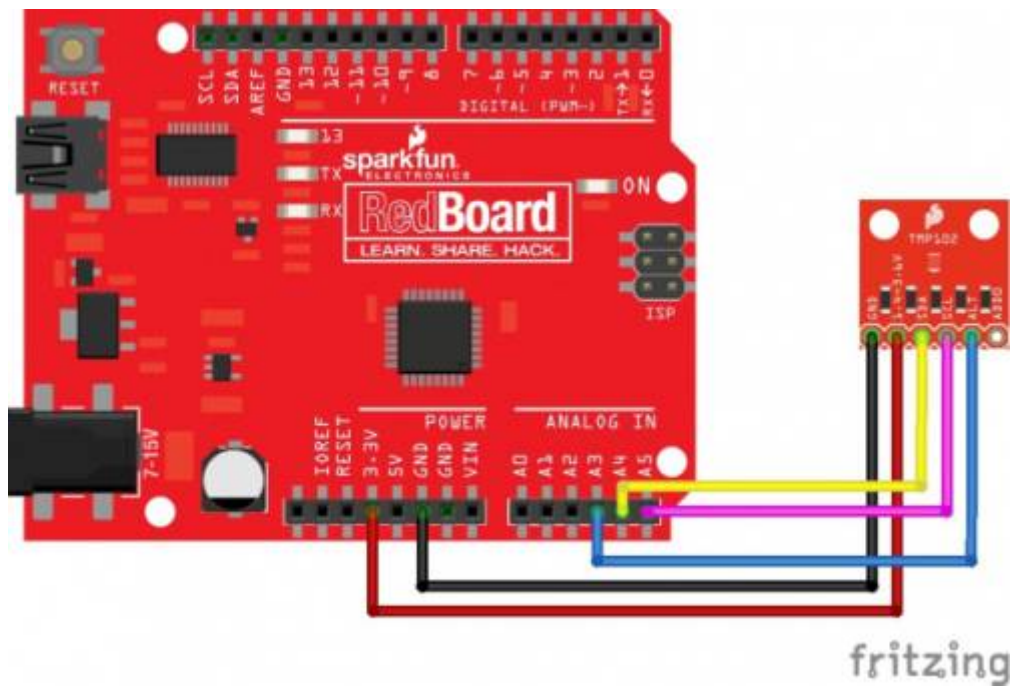
## Description

Le **TMP102** de Texas Instruments est un **capteur de température numérique**. Il transmet la température mesurée par un bus **I2C**. Sa **résolution** est égale à **0,0625°C** et sa **précision** **0,5°C**. Le module intègre des résistances de tirage de 4,7 kΩ et fonctionne entre 1,4 V et 3,6 V.

## Connexions

- VCC → 3.3V
- GND → GND
- SDA → SDA/A4
- SCL → SCL/A5
- ALT → A3


Exemple avec une ancien modèle de carte







- ADD0 est utilisée pour changer l'adresse du TMP102.
  - VCC → 0x49
  - SDA → 0x4A
  - SCL → 0x4B














La classe TMP102



Constructeur

| Syntaxe  | Description   |
|--|---|
| <div> TMP102(<b>byte</b> adresse);</div> <div>// Exemple<br/>TMP102 sensorTemp(0x48);</div> | Initialise un objet TMP102 avec l'adresse passée en paramètre |

Méthodes

| Syntaxe  | Description  |
|--|--|
| Initialisation   |  |
| <div> <b>void</b> begin(<b>void</b>);</div>         | Connecte le capteur au bus I2C.<br>// Exemple<br>sensorTemp.begin();   |
| Lecture de la température  |  |
| <div> <b>float</b> readTempC(<b>void</b>);</div>    | Retourne la température en °C.<br>// Exemple<br>float temperature = sensorTemp.readTempC();                  |
| <div> <b>float</b> readTempF(<b>void</b>);</div>    | Retourne la température en °F.<br>// Exemple<br>float temperature = sensorTemp.readTempF();                  |
| <div> <b>float</b> readLowTempC(<b>void</b>);</div> | Retourne le contenu du registre T_LOW en °C.<br>// Exemple<br>float temperature = sensorTemp.readLowTempC(); |

|   | Syntaxe   | Description  |
|---|---|--|
|    | <b>float</b> readHighTempC( <b>void</b> );            | Retourne le contenu du registre T_HIGH en °C.<br>// Exemple<br><code>float temperature = sensorTemp.readHighTempC();</code>  |
|    | <b>float</b> readLowTempF( <b>void</b> );             | Retourne le contenu du registre T_LOW en °F.<br>// Exemple<br><code>float temperature = sensorTemp.readLowTempF();</code>  |
|    | <b>float</b> readHighTempF( <b>void</b> );            | Retourne le contenu du registre T_HIGH en °F.<br>// Exemple<br><code>float temperature = sensorTemp.readHighTempF();</code>  |
| <b>Consommation, seuils de température et alerte</b>                                |   |  |
|    | <b>void</b> sleep( <b>void</b> );                     | Passe le capteur en mode faible consommation (<0,5µA).<br>// Exemple<br><code>sensorTemp.sleep();</code>   |
|    | <b>void</b> wakeup( <b>void</b> );                    | Réveille le capteur et le ramène en mode consommation normale (~10µA).<br>// Exemple<br><code>sensorTemp.wakeup();</code>  |
|    | <b>bool</b> alert( <b>void</b> );                     | Retourne le contenu du registre d'alerte. La valeur est identique à celle la broche ALT.<br>// Exemple<br><code>bool state = sensorTemp.alert();</code>                                      |
|   | <b>void</b> setLowTempC( <b>float</b> temperature);   | Ecrit le seuil d'alerte bas (en °C) dans le registre T_LOW.<br>// Exemple<br><code>sensorTemp.setLowTempC(20.0);</code>  |
|  | <b>void</b> setHighTempC( <b>float</b> temperature);  | Ecrit le seuil d'alerte haut (en °C) dans le registre T_High.<br>// Exemple<br><code>sensorTemp.setHighTempC(22.0);</code>   |
|  | <b>void</b> setLowTempF( <b>float</b> temperature);   | Ecrit le seuil d'alerte bas (en °F) dans le registre T_LOW.<br>// Exemple<br><code>sensorTemp.setLowTempF(84.0);</code>  |
|  | <b>void</b> setHighTempF( <b>float</b> temperature);  | Ecrit le seuil d'alerte haut (en °F) dans le registre T_High.<br>// Exemple<br><code>sensorTemp.setHighTempC(85.0);</code>   |
| <b>Utilitaires</b>  |   |  |
|  | <b>void</b> setConversionRate( <b>byte</b> rate);     | Règle le taux de conversion (0-3).<br>0 → 0,25Hz<br>1 → 1Hz<br>2 → 4Hz (par défaut)<br>3 → 8Hz<br>// Exemple<br><code>sensorTemp.setConversionRate(3);</code>                                |
|  | <b>void</b> setExtendedMode( <b>bool</b> mode);       | Active ou désactive le mode étendu.<br>false → désactive (-55°C à +128°C)<br>true → active (-55°C à +150°C)<br>// Exemple<br><code>sensorTemp.setExtendedMode(false);</code>                 |
|  | <b>void</b> setAlertPolarity( <b>bool</b> plolarity); | Règle la polarité de l'alerte.<br>false → active à l'état BAS<br>true → active à l'état HAUT<br>// Exemple <code>&lt;code cpp&gt;</code><br><code>sensorTemp.setAlertPolarity(false);</code> |

|   | Syntaxe  | Description   |
|---|--|---|
|  | <b>void</b> setFault( <b>byte</b> faultSetting); | Règle le nombre d'erreurs consécutives.<br>0 → 1 erreur<br>1 → 2 erreurs<br>2 → 4 erreurs<br>3 → 6 erreurs<br>// Exemple<br>sensorTemp.setFault(1);   |
|  | <b>void</b> setAlertMode( <b>bool</b> mode);     | Règle le type d'alerte.<br>false → Mode comparateur: Actif si temp > T_HIGH ou temp < T_LOW<br>true → Mode thermostat: actif quand temp > T_HIGH jusqu'à ce qu'une opération de lecture se produise<br>// Exemple <code cpp><br>sensorTemp.setAlertMode(false); |

## Exemple à télécharger

[temp1.cpp](#)

```
// Réglage de l'intelliSense (pour la classe TMP102)
// "C_Cpp.intelliSenseEngine": "Default" ou
// "C_Cpp.intelliSenseEngine": "Tag Parser" ne doit pas se trouver dans
// setting.json
// Fermer puis rouvrir éventuellement VSCode après la modification
#include <Wire.h>
#include <SparkFunTMP102.h>

TMP102 cptTemp(0x48); // Construction d'un objet TMP102 avec l'adresse
I2C 0x48

void setup()
{
    Serial.begin(115200); // Débit binaire par défaut du moniteur série
    cptTemp.begin(); // Connexion du capteur au bus I2C
}

void loop() {
    float temperature;
    temperature = cptTemp.readTempC(); // Lecture de la température
    Serial.print("Temperature: "); // Affichage dans le moniteur série
    Serial.print(temperature);
    Serial.println();
    delay(1000);
}
```

From:

<http://webge.fr/dokuwiki/> - **WEBGE Wikis**

Permanent link:

<http://webge.fr/dokuwiki/doku.php?id=materiels:capteurs:temperature:libtmp102&rev=1692370096>

Last update: **2023/08/18 16:48**

