

Capteur de température : DS18B20

D.Moreaux

8 mars 2020

1 Le composant

LE DS18B20 est un capteur de température calibré qui retourne une température précise au 1/10 de degré. Plusieurs de ces capteurs peuvent être connectés en parallèle et interrogés individuellement.

Son utilisation se base sur la librairie *OneWire* utilisée également par d'autres composants du même fabricant. Chaque composant a une identité composée d'un nombre de 8 bits identifiant le composant et d'un numéro de série sur 48 bits. A ces deux valeurs on ajoute une somme de contrôle sur 8 bits pour former une adresse sur 64 bits (8 octets).

Lorsque l'on a l'inscription face à soi, les pattes sont la masse, la broche de données et l'alimentation positive. On le trouve sur des platines avec un brochage différent (GND, Vcc, data).

Ce composant nécessite un PullUp sur le signal data partagé par tous les composants de la chaîne mais le pullup interne à l'Arduino peut faire l'affaire.

Il est possible de n'utiliser que deux fils : la masse et le signal data. Le composant chargera un condensateur interne à partir du +5V arrivant au travers du pull-up. Lorsqu'il est utilisé ainsi, on veillera à faire des pauses suffisantes entre deux opérations longues pour que ce condensateur puisse se recharger.

La lecture de température se fait en deux étapes : on demande d'abord une conversion et puis on lit le résultat à partir du "bloc note" du composant. La conversion peut prendre de 94 à 750ms selon la précision demandée (de 9 à 12 bits).

On trouve d'autres composants semblables : DS18S20, DS1822, DS1820, MAX31820 qui possèdent des identifiants différents (le DS18B20 a un identifiant de 0x28).

2 La librairie OneWire

La librairie OneWire permet de contrôler différents composants du fabricant. Cette librairie ne s'occupe que des accès de bas niveau (envoi d'une commande et lecture de valeurs) et il faut consulter la documentation du composant pour déterminer les commandes à envoyer.

Dans la plupart des cas, on préférera utiliser des bibliothèques de haut niveau qui se baseront sur OneWire mais se chargeront de l'utilisation des commandes correctes et des calculs des valeurs finales.

3 La librairie DallasTemperature.h

Plutôt que d'adresser le composant directement, on peut utiliser la librairie DallasTemperature qui se chargera de pas mal de tâches. La librairie OneWire devra être activée.

```
#include <OneWire.h>
#include <DallasTemperature.h>

OneWire onewire(pin);
DallasTemperature sensor(&onewire);

void setup() {
  sensor.begin();
}
```

On pourra ensuite utiliser les commandes suivantes :

<p>sensor.requestTemperatures()</p> <p>sensor.requestTemperaturesByIndex(idx)</p> <p>sensor.getTempCByIndex(idx)</p> <p>sensor.getTempFByIndex(idx)</p> <p>sensor.setResolution(res)</p> <p>sensor.getDS18Count()</p>	<p>lance la lecture de température sur tous les composants attachés sur la connexion One Wire</p> <p>Lance la lecture de température sur le nième périphérique</p> <p>Lit la température Celsius du nième périphérique attaché (0 pour le premier)</p> <p>Lit la température Fahrenheit du nième périphérique attaché (0 pour le premier)</p> <p>Fixe la résolution de tous les capteurs. res est une valeur de 9 à 12</p> <p>Retourne le nombre de capteurs de température sur la connexion</p>
<p>sensor.getAddress(adr,idx)</p> <p>sensor.isConnected(adr)</p> <p>sensor.requesttemperaturesByAddress(adr)</p> <p>sensor.getTempC(adr)</p> <p>sensor.getTempF(adr)</p>	<p>Récupère l'adresse du nième capteur dans le tableau de 8 octets adr</p> <p>Retourne TRUE si le capteur avec l'adresse adr est détecté</p> <p>Lance la lecture sur le capteur dont l'adresse est adr</p> <p>Lit la température Celsius sur le capteur dont l'adresse est adr</p> <p>Lit la température Fahrenheit sur le capteur dont l'adresse est adr</p>

Il existe encore bien d'autres fonctions, certaines ne sont disponibles que si on les autorise par `#define` avant d'inclure la librairie.

En général, on utilisera juste `requestTemperatures()` et `getTempCByIndex(0)`.