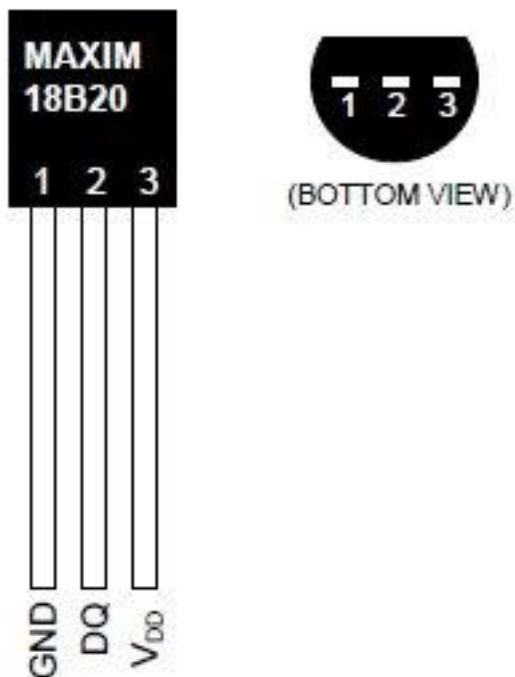


[Lire la température d'un DS18B20 via le bus 1-WIRE en C](#)Catégorie : [Commun à toute la gamme](#)

Publié par dinastar le 13/06/2016

Je vous propose dans cet article, de mettre en œuvre le bus 1-WIRE afin de récupérer la température d'un ou de plusieurs capteur(s) numérique(s) DS18B20.





Les principales choses à savoir

- Le bus n'a besoin que d'un seul « fil » donc seulement une seule broche de prise sur votre μC pour communiquer en bidirectionnel (Vcc,DQ,Gnd soit 3 fils utilisés).
- Résistance de tirage (4,7K) de DQ vers Vcc, donc quand DQ = Vcc, bus au repos.
- Mode « Parasite Power » possible et permettant d'alimenter le capteur par le bus (DQ,Gnd soit 2 fils utilisés). En gros, dans le capteur, une capa entre Vcc et DG permet de filtrer les trames binaires afin de créer Vcc.
- 64bits de code unique (comme une adresse) en ROM. Donc obligation de lire dans la ROM interne du DS18B20 afin de connaître son code et de communiquer avec lui, sauf si qu'un seul DS est présent physiquement sur le bus (dommage qu'il ne soit pas gravé visuellement sur le boîtier ...).
- 55°C à 125°C (précision 0.5% de -10 à 85°C)
- Résolution de 9 à 12bits => précision de 0,5 à 0,0625°C => temps de conversion de 93,75 à 750ms.
- 3 octets d'EEPROM pour sauver quelques paramètres (résolution, alarme max, alarme min)
- Contrôle d'erreur par CRC8
- Vitesse du bus standard : ~16Kbits/s
- Il existe un mode de vitesse « overdrive » mais n'est possible qu'avec

certaines familles voire certains composants au sein même d'une famille. C'est le programme qui devra les tester un par un.

Sources de lecture

Je ne vais pas détailler ici le fonctionnement du bus en profondeur, mais forcément, pour aller plus loin, je vous conseille vivement de lire ces 2 documents :

« Le Bus 1-WIRE » de Daniel MENESPLIER en français ...

<http://daniel.menesplier.free.fr/Doc/BUS%201%20WIRE.pdf>

... vous donnera déjà une bonne approche sur le fonctionnement de ce bus, et de quelques périphériques très connus aujourd'hui comme le DS18B20, la clé iButton DS1990 etc ...

Après cela, il sera plus confortable, surtout pour ceux qui ont des difficultés en anglais, de lire la datasheet du capteur.

« DS18B20 Programmable Resolution 1-Wire Digital Thermometer » en anglais donc

<http://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>

Une autre réalisation d'un membre, et sur lequel je me suis également inspiré

<http://www.68hc08.net/articles.item.1...escale.html?keywords=wire>

Les fichiers de code en C

- OWIRE.c => toutes les fonctions nécessaires pour créer le bus. Et donc compatibles avec tous les composants 1-Wire existants à ce jour.
- DS18B20.c => fonctions spécifiques pour communiquer avec ce capteur.

Voici un exemple d'utilisation de cette lib pour 3 capteurs DS18B20 :

Attention ! Dans mon application, les codes ROM de chaque capteur, uniques, ont été sauvegardés dans une mémoire EEPROM externe et dédiée à cela. En effet, il existe une méthode (lire le PDF de D.MENESPLIER) afin d'effectuer une recherche de tous les DS présents sur le bus (non codée dans ma lib).

Cela est spécifique dans le cas où plus d'un capteur est connecté sur le même bus. Là est la difficulté : gérer les code ROM et savoir où sont placés les capteurs !

Dans mon système, à la configuration initiale, je connecte les capteurs un à un afin d'affecter chaque code ROM à un emplacement physique (pour sonder 3 pièces de la maison par ex.), et donc un emplacement en mémoire EEPROM. Comme cela, le système reconnaît tous les DS18B20 connectés sur le bus et ce, à chaque marche/arrêt de la carte.

ï»¿

İ»ı