

## Surveillance de température avec Linux



par Stefan Blechschmidt  
<sb/at/sbsbavaria.de>

### *L'auteur:*

Electricien confirmé, je me suis retrouvé, en 1990, devant une station de travail à faire du développement d'une station de contrôle et de commutation. J'ai, bien évidemment été infecté par un « virus » inconnu mais c'était parfait.

### *Traduit en Français par:*

Iznogood  
<iznogood/at/iznogood-factory.org>



### *Résumé:*

Les serveurs Linux effectuent leur travail d'une manière fiable et en sécurité. Ils n'ont pas besoin d'attentions spéciales, ils travaillent, travaillent et travaillent... Mais que se passe-t-il si le pingoin devient trop chaud ? L'été passé vient de nous le montrer : nous devrions équiper notre petit ami avec un système de surveillance de la température. Cet article vous montrera comment en installer un facilement pour 10 Euros.

---

## Les Composants

Brian C.Lane a écrit un programme pour le capteur de température DS1820 de DALLAS Semiconductor. Il n'est pas difficile d'assembler un système de contrôle de température avec ces deux composants et quelques autres.

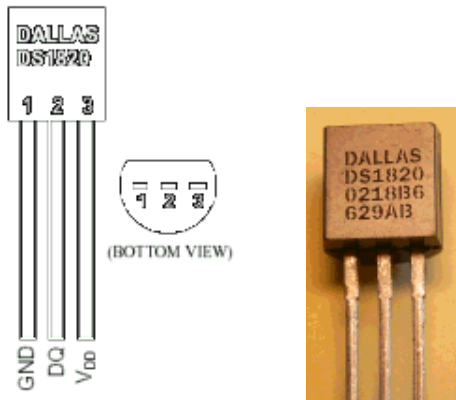
### **Note :**

Brian C.Lane a modifié sa page et adapté la version que j'utilise. Vous pouvez donc le télécharger sur : [digitemp-1.3.tar.gz](#)

Ceux qui souhaitent utiliser la version actuelle [digitemp-3.2.0.tar.gz](#) peuvent la télécharger sur

<http://www.digitemp.com/software.shtml> .

## Le Capteur



Le capteur avait été initialement produit par DALLAS Semiconductor qui a fusionné en Maxim/Dallas Semiconductor. Selon les spécifications techniques, le capteur peut mesurer les températures de  $-55^{\circ}\text{C}$  à  $125^{\circ}\text{C}$ . Les données de mesure sont générées comme un signal numérique avec une largeur de 9 bits. De plus, chaque capteur possède un identifiant sur 64 bits, permettant une connexion sur bus de plusieurs capteurs. Il est possible d'utiliser 100 capteurs sur un bus ayant une longueur de 300m .

Le circuit que nous présentons ici ne devrait fonctionner que pour 10 capteurs sur un bus de 60m. J'utilise actuellement 4 capteurs sur un bus d'environ 12m.

Vous pouvez trouver plus d'informations à propos de ce capteur sur cette Fiche Technique.

Je voudrais simplement ajouter que le circuit présenté ici ne peut faire de mesures que jusqu'à  $75^{\circ}\text{C}$ , ce qui devrait être suffisant pour notre application.

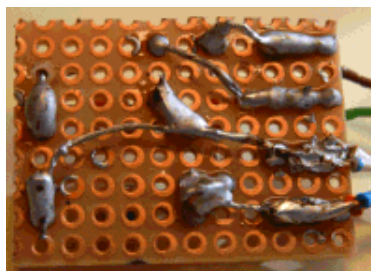
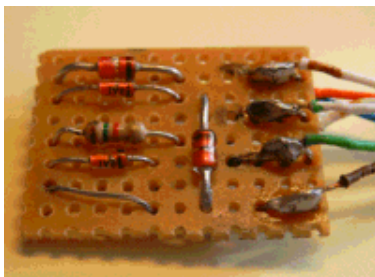
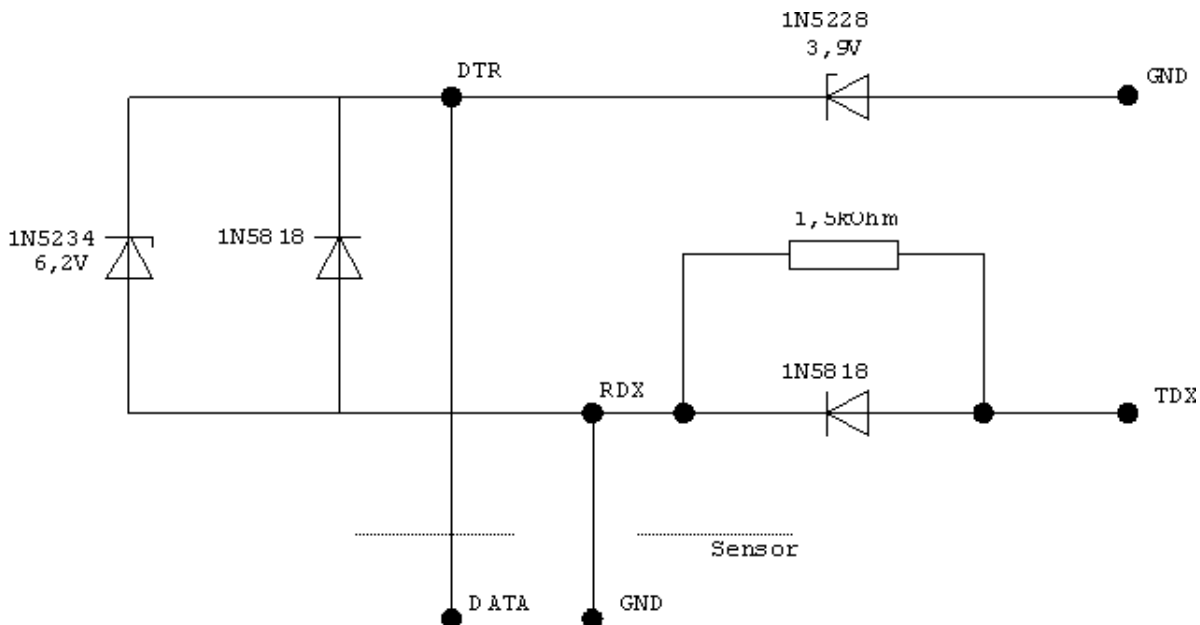
## Les Autres Composants

Pour pouvoir connecter le capteur à une interface série, nous avons besoin de composants supplémentaires. Vous les trouverez dans la plupart des boutiques vendant les composants au détail, sur internet ou mieux dans la boutique électronique du coin.

Quantité	Identification	Type	alter.Type
1	Résistance	1,5 k Ohm	-----
2	Diode Schottky	1N5818	BAT 43
1	Diode Zener	1N5228	ZPY 3,9V
1	Diode Zener	1N5234	ZPY 6,2V
1	Capteur	DS18S20	-----
1	Connecteur, Série	RS232C/9	-----
1	Protection de Connecteur	SUB-D/9	-----
1	Circuit	Prépercé	-----

## Design of the Interface

Comme le circuit d'interface est très simple, j'ai choisi une plaque d'essai. Désolé pour la qualité des soudures :-).





-s/dev/ttyS0	Set serial port
-l/var/log/temperature	Send output to logfile
-f5	Fail delay in S
-r500	Read delay in mS
-v	Verbose output
-t0	Read Sensor #
-a	Read all Sensors
-d5	Delay between samples (in sec.)
-n50	Number of times to repeat
-o2	Output format for logfile
-o"output format string"	See description below

Logfile formats: 1 = One line per sensor, time, C, F (default)  
 2 = One line per sample, elapsed time, temperature in C  
 3 = Same as #2, except temperature is in F  
 #2 and #3 have the data seperated by tabs, suitable for import  
 into a spreadsheet or other graphing software.

The format string uses strftime tokens plus 3 special ones for  
 digitemp - %s for sensor #, %C for centigrade, %F for fahrenheit.  
 The case of the token is important! The default format string is:  
 "%b %d %H:%M:%S Sensor %s C: %.2C F: %.2F" which gives you an  
 output of: May 24 21:25:43 Sensor 0 C: 23.66 F: 74.59

Comme « l'aide » nous l'indique, les capteurs installés doivent être initialisés. Pour ce faire, nous devons définir l'interface à laquelle est connecté le circuit, ainsi que les paramètres de l'initialisation.

### La commande

```
digitemp -i -s/dev/ttyS0
```

le réalise dans le cas où le circuit d'interface est connecté à la première interface série.

Le logiciel détecte les capteurs et un message ressemblant à ceci apparaît :

```
DigiTemp v1.3 Copyright 1997-99 by Nexus Computing
```

```
ROM #0 : 1032724700080086
```

```
ROM #1 : 1092214400080089
```

Nous remarquons que le logiciel a détecté 2 capteurs. De plus, le fichier **.digitemprc** sera créé dans le répertoire courant ; il contiendra les données du capteur, l'interface et le format de sortie.

Avec la commande **./digitemp -a**, nous sommes alors en mesure de lire la sortie des capteurs. Veuillez noter le caractère **./** car nous sommes encore dans un répertoire qui ne fait pas partie du chemin de recherche courant.

### Sortie de la Mesure

```
DigiTemp v1.3 Copyright 1997-99 by Nexus Computing
```

```
Sep 24 21:53:35 Sensor 0 C: 37.94 F: 100.29
```

```
Sep 24 21:53:38 Sensor 1 C: 10.62 F: 51.129
```

## Adapter le Système

Pour intégrer notre programme dans le système, nous devons procéder à quelques réglages.

Nous devons d'abord copier les exécutable, i.e. `digitemp`, dans un répertoire qui permet l'exécution de programmes sans en spécifier le chemin. J'ai sélectionné `/usr/local/bin/` pour ce faire. Ceux qui n'en sont pas sûr peuvent trouver les chemins avec `echo $PATH`. De plus, le fichier `.digitemprc`, contenant les données d'initialisation, doit être copié dans le répertoire de l'utilisateur qui effectue les mesures. Pour sauvegarder les données dans un fichier et pas sur la console, le switch `-l[PATH/FILE NAME]` doit être appliqué.

## Mesures Automatiques

Nous avons besoin d'automatiser nos mesures, ce que `cron` fera très bien pour nous. Avec `crontab -u [USER] -e`, nous pouvons le mettre en place pour tous les utilisateurs. L'entrée

```
0-59/15 * * * * /usr/local/bin/digitemp -a -l/var/log/digitemp.log
```

initialise `digitemp` pour faire des mesures toutes les 15 minutes et de les écrire dans `/var/log/digitemp.log`.

Avec la commande `tail /var/log/digitemp.log`, nous pouvons lire les dernières lignes des mesures. Si le fichier `/var/log/digitemp.log` n'a pas été généré, veuillez vérifier les droits d'accès des fichiers.

## Commentaires Finaux

L'archive contient quelques scripts Perl pour l'interprétation graphique. Je ne vais pas les décrire ici. Je réfléchis à un autre article sur la manière de collecter les données des capteurs dans une base de données MySQL et sur la manière de les interpréter via une interface web..

Cela sera principalement réalisé en Perl par l'utilisation des fonctions MySQL, CGI et graphiques.

## Téléchargements

- [digitemp-1.3.tar.gz](#)

## Liens / Références

- <http://www.fli4l.de/> - Page d'accueil fli4l
- <http://www.perl.org/> - Page d'accueil Perl
- Survol général de DS1820
- Fiche de Spécification DS18S20
- Page Digitemp de Brian C.Lane
- Page d'accueil Brian C.Lane
- Page d'accueil de Maxim/Dallas Semiconductor
- LinuxNetMag Article et Forum

---

Site Web maintenu par l'équipe d'édition LinuxFocus © Stefan Blechschmidt "some rights reserved" see <a href="http://linuxfocus.org/license/">linuxfocus.org/license/</a> <a href="http://www.LinuxFocus.org">http://www.LinuxFocus.org</a>	Translation information: de --> -- : Stefan Blechschmidt <sb/at/sbsbavaria.de> de --> en: Jürgen Pohl <sept.sapins/at/verizon.net> en --> fr: Iznogood <iznogood/at/iznogood-factory.org>
--	--