

Sommaire

Configuration des Senseurs de la carte mère.....	1
Ce qu'il nous faut.....	1
Installation.....	1
Compilation des modules et du noyau.....	2
Compilation des bibliothèques et des utilitaires.....	2
Détection des senseurs.....	3
Utilisation.....	3
Copyright.....	6

Configuration des Senseurs de la carte mère.

Configuration des Senseurs de la carte mère.

par Fred

Votre carte mère dispose peut-être de senseurs permettant de savoir, à tout moment, la température des différents éléments de celle-ci, ainsi que diverses informations intéressantes pour la bonne marche de votre PC, en particulier si vous comptez l'overclocker.

Ce qu'il nous faut

Pour pouvoir connaître l'état des différents senseurs de votre carte mère, il faut que nous installions deux (au moins) paquetages : `i2c-2.9.2.tar.gz` et `lm_sensors-2.9.2.tar.gz` que l'on peut trouver sur le site de l'auteur de ces paquetages.

Évidemment, lecteur quand tu liras ces lignes, les numéros des paquetages auront peut-être changé : prend la dernière version !

Installation

La documentation de ces deux paquetages préconise trois types d'installation (au moment où j'écris ces lignes), mais je ne vais m'intéresser qu'à une seule méthode : celle qui consiste à patcher (je ne vois de mot français, y a-t-il un académicien dans la salle ?) le noyau de Linux. L'opération se déroule sans douleur (évidemment, si vous n'avez jamais recompilé votre noyau, [../kernel/kernel.php3 allez voir la documentation sur ce site]).

Remarque : chaque commande qui suit est sur une seule ligne, bien sûr !

La première chose à faire est de décompresser ces deux paquetages :

```
[user@localhost user]$ tar xzf i2c-2.9.2.tar.gz
[user@localhost user]$ tar xzf lm_sensors-2.9.2.tar.gz
```

Ensuite, il faut créer les rustines (ou patch en anglais). Pour que cela fonctionne, il faut, au préalable, avoir installé les sources du noyau dans `/usr/src/linux` (c'est l'emplacement par défaut de la plupart des distributions ; les paquetages nécessaires sont disponibles sur le CD de votre distribution).

Nous créons les rustines par :

```
[user@localhost user]$ cd /home/user/i2c-2.9.2
[user@localhost i2c-2.9.2]$ mkpatch/mkpatch.pl ./usr/src/linux i2c-patch
Puis, on patche le noyau :
```

```
[user@localhost user]$ cd /usr/src/linux
[user@localhost linux]$ su
Password : *****
[root@localhost linux]# patch -p1 -E </home/user/i2c-2.9.2/i2c-patch
```

...
...

(Attention: ces commandes sont sur une seule ligne !) Voilà pour le patch `i2c`. Puis le patch `lm_sensors` :

```
[user@localhost user]$ cd /home/user/lm_sensors-2.9.2
[user@localhost lm_sensors-2.9.2]$ mkpatch/mkpatch.pl . /usr/src/linux > lm-patch
Enfin, il faut patcher le noyau :
```

```
[root@localhost linux]# patch -p1 -E < /home/user/lm_sensors-2.9.2/lm-patch
```

...

...

Normalement, tout ce passe bien et nos sources du noyau sont bien patchées !

Compilation des modules et du noyau

La compilation se déroule de manière standard pour les sources du noyau : on lance un `make xconfig` pour configurer les sources.

En plus des supports habituels pour votre configuration ([./kernel/kernel.php3 voir la documentation du site]), il faut que vous activiez (en modules c'est mieux...) tout dans : `Character Devices/I2C support/` (en fait, seuls `I2C support`, `I2C bit-banging interfaces`, `I2C PCF 8584 interfaces` et `Pseudo ISA adapter` nous seront utiles). Il faut aussi que vous activiez tout dans : `Character Devices/Hardware sensors support/`.

Puis vous compilez votre noyau comme d'habitude :

```
make dep ; make bzImage ; make modules ; make modules_install
```

Vous l'installez, vous relancez lilo et vous rebootez !

Compilation des bibliothèques et des utilitaires

Cette phase est essentielle : elle permet la détection automatique de vos senseurs...

Pour cela vous devez compiler les bibliothèques (library en anglais) `libsensors.*`. Placez vous dans le répertoire contenant les sources de `lm_sensors` :

```
$ cd /home/usr/lm_sensors-2.9.2
```

Et éditez le fichier nommé `Makefile` (c'est lui qui gère la compilation de `lm_sensors`), vous devez trouver les lignes noires et les remplacer par les lignes rouges :

```
COMPILE_KERNEL := 1
```

```
COMPILE_KERNEL := 0
```

```
I2C_HEADERS=/usr/local/include
```

```
I2C_HEADERS=$(LINUX_HEADERS)
```

```
PREFIX := /usr/local
```

```
PREFIX := /où/vous/voulez
```

```
# attention : les binaires seront installés dans $PREFIX/bin etc...
```

En fait, il suffit souvent de commenter la ligne noire (en mettant un `#` devant) et de décommenter la ligne rouge (en enlevant le `#` devant).

Maintenant on peut compiler le tout par `make`. Un certain nombre de fichiers ne sont pas trouvés : ce n'est pas grave. On installe le tout par : `make install`.

Une dernière étape avant la détection : la création des périphériques. Celle-ci se fait très simplement par le script : `mkdev.sh`. Il faut être root pour pouvoir le lancer :

```
$ su
Password : *****
# /home/user/lm_sensors-2.6.3/prog/mkdev/mkdev.sh
Voilà, il ne reste plus qu'à détecter vos senseurs.
```

Détection des senseurs

Là, c'est très simple : il suffit de lancer le script `sensors-detect` qui se trouve dans `/home/user/lm_sensors-2.9.2/prog/detect`, lancez-le et validez toutes les questions (les choix par défaut sont tous bons, mais vous pouvez lire ce qu'on vous demande pour affiner votre configuration). À la fin du script vous devez voir apparaître quelque chose comme :

```
WARNING! If you have some things built into your kernel, the
below list will contain too many modules. Skip the appropriate ones!
To load everything that is needed, add this to some /etc/rc* file:
```

```
#----cut here----
# I2C adapter drivers
modprobe i2c-piix4
modprobe i2c-isa
# I2C chip drivers
modprobe eeprom
modprobe w83781d # w83781d est le nom du chip !
#----cut here----
```

To make the sensors modules behave correctly, add these lines to either `/etc/modules.conf` or **`/etc/conf.modules`**:

```
#----cut here----
# I2C module options
alias char-major-89 i2c-dev
#----cut here----
```

La première série de lignes en rouge (entre les `cut here`) doit être insérée dans votre fichier `/etc/rc.d/rc.local` : ce sont ces lignes qui vont charger au démarrage les modules permettant la lecture des senseurs (Attention : prenez les lignes correspondant à votre configuration, celles qui sont présentées ici sont les miennes !).

La seconde série doit être insérée dans le fichier `/etc/conf.modules` : elles servent au chargement automatique des modules qui peuvent être chargé automatiquement.

Voilà c'est tout.

Utilisation

L'installation de `lm_sensors-2.9.2` crée un petit script de lecture des senseurs : `sensors`, lorsqu'on le lance on obtient :

Hardware-hard_plus-sensors

```
[user@localhost user]# sensors
w83781d-i2c-1-2d
Adapter: SMBus PIIX4 adapter at 5000
Algorithm: Non-I2C SMBus adapter
VCore 1: +1.98 V (min = +1.80 V, max = +2.20 V)
VCore 2: +2.00 V (min = +1.80 V, max = +2.20 V)
+3.3V: +3.36 V (min = +2.97 V, max = +3.63 V)
+5V: +5.07 V (min = +4.50 V, max = +5.48 V)
+12V: +11.86 V (min = +10.79 V, max = +13.11 V)
-12V: -12.38 V (min = -10.78 V, max = -13.18 V)
-5V: -5.12 V (min = -4.50 V, max = -5.48 V)
CPU Fan: 4891 RPM (min = 3000 RPM, div = 2)
CPU temp: +39.0 C (limit = +60.0 C, hysteresis = +50.0 C)
vid: +2.00 V
alarms: Chassis intrusion detection ALARM
beep_enable:
  Sound alarm disabled
```

Ou en tout cas quelque chose d'approchant. Certaines informations peuvent vous sembler absurdes (par exemple des températures négatives, etc.), elles correspondent en général à des senseurs non branchés ou non existants. Il peut être intéressant de désactiver ces senseurs (du moins leur lecture), cela se fait en éditant le fichier `/etc/sensors.conf`. Mais avant de pouvoir l'identifier il faut connaître le nom de la puce qui gère les senseurs : le `[#chip_name chip]`. Pour l'identifier, c'est simple, vous faites :

```
[user@gros home]# cat /proc/sys/dev/sensors/chips
256 w83781d-isa-0290
257 w83781d-i2c-1-2d
258 eeprom-i2c-1-50
```

Les noms des puces présentes sur ma carte mère sont en **gras**, en **rouge** c'est le protocole utilisé pour lire le contenu de ces puces.

Dans le fichiers `/etc/sensors.conf`, vous devez vous placer après une ligne commençant par :

```
chip "lm78-*" "lm78-j-*" "lm79-*" w83781d-*" "sis5595-*
```

(Comme vous pouvez le constater, plusieurs puces ont la même configuration : ici lm78, lm79, w83781d, sis5595, mais ce qui doit vous intéresser dans ce fichier c'est votre puce !, le reste vous pouvez l'effacer). Vous avez donc repéré votre puce dans `/etc/sensors.conf`. Vous devez trouver à la suite de cette ligne (chip quelque chose):

```
label in0 "VCore 1"
label in1 "VCore 2"
label in2 "+3.3V"
label in3 "+5V"
label in4 "+12V"
label in5 "-12V"
label in6 "-5V"

label fan1 "CPU Fan"
ignore fan2
ignore fan3
```

```
ignore temp1  
label temp2 "CPU temp"  
ignore temp3
```

Les lignes commençant par **label** donnent un nom (celui que vous verrez dans la sortie de `sensors`) à un capteur : mettez ce qui vous plaît. Les lignes commençant par **ignore** signalent à `sensors` que ce capteur ne doit pas être lu : les capteurs qui retournent des informations erronées ou qui ne vous intéressent pas peuvent être ignorés avec profit.

Voilà, maintenant vous pouvez surveiller votre PC dans les moindres recoins !

Cette page est issue de la documentation 'pré-wiki' de Léa a été convertie avec HTML::WikiConverter. Elle fut créée par Frédéric Bonnaud.

Copyright

Copyright © Frédéric Bonnaud



*Ce document est publié sous licence Creative Commons
Attribution, Partage à l'identique, Contexte non commercial 2.0 :*
<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>