



Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

## Lm-sensors: Controla la temperatura de tu procesador (58033 lecturas)

Per Miguel Ángel Calderón, [MAC](#) ()

Creado el 04/11/2001 16:59 modificado el 04/11/2001 16:59

*[Lm\\_sensors](#)<sup>(1)</sup> es un proyecto que pretende dar soporte a todos los dispositivos sensores modernos dentro del kernel. Quieres instalarlo en tu linux? sigue leyendo...*

## Como monitorizar la temperatura de tu ordenador en Linux

Hoy en día tenemos procesadores que alcanzan velocidades de reloj superiores al GHz, con lo que la temperatura que puede alcanzar dicho componente está empezando a resultar un dato muy importante que conviene tener controlado.

Muchas de las placas base diseñadas a partir de 1997 incluyen dispositivos para monitorizar la temperatura a la que se encuentran los componentes más importantes del ordenador, la CPU, la memoria, etc. o por ejemplo también el número de revoluciones por minuto de los ventiladores que forman parte del sistema de refrigeración.

El proyecto que se encarga del soporte de dichos dispositivos sensores dentro del kernel de linux se llama [lm\\_sensors](#)<sup>(1)</sup>. Por ejemplo el chip LM78, pese a estar considerado obsoleto hoy en día por su fabricante, *National Semiconductors*, es uno de los precursores en este género.

Una manera sencilla para ver si nuestra placa base viene equipada con alguno de estos sensores es mirar en la BIOS a ver si aparece, en la sección *Gestión de energía*, algún indicativo de la temperatura del procesador, placa o memoria, así como un apartado para la monitorización del número de revoluciones por minuto de los ventiladores que lleva el ordenador.

Para más información sobre el proyecto pasaos por la sección [info](#)<sup>(2)</sup> de la web oficial.

## Consiguiendo la última versión de lm\_sensors

En la sección [download](#)<sup>(3)</sup> de la web oficial puedes encontrar versiones de lm-sensors para los kernel 2.0.x, 2.2.x y 2.4.x. En el momento en que se escribió este artículo la última versión estable era la 2.6.1.

Los archivos que necesitaremos son los siguientes:

- El paquete con el código fuente del lm\_sensors: [lm\\_sensors-2.6.1.tar.gz](#)<sup>(4)</sup>
- El paquete con el código fuente de la última versión del driver del bus i2c: [i2c-2.6.1.tar.gz](#)<sup>(5)</sup>

Nota: En Debian disponemos de esos paquetes, tan solo hay que ejecutar `apt-get install lm-sensors-source i2c-source` y nos instalará tanto los binarios para monitorizar el estado de los sensores como los fuentes para el kernel. Una vez instalados los paquetes de los fuentes Debian deja los ficheros tar.gz en /usr/src .

## Compilando los fuentes

El primer paso para conseguir compilar lm-sensors es obtener una versión de los módulos del kernel para el bus i2c que sea lo suficiente moderna, y disponer de las cabeceras necesarias para la compilación de lm-sensors. La versión que se indica en el apartado anterior es adecuada. Desconozco si existe algún kernel 2.4.x que disponga de una versión de i2c



lo suficientemente moderna. La versión 2.4.9, que es la que yo dispongo en este momento no lo es.

Una vez disponemos de la versión de i2c adecuada y la hemos descomprimido su compilación no tiene mayor secreto, *make* nos generará los archivos binarios.

Para adaptar la compilación en nuestro sistema podemos modificar el Makefile, donde encontraremos variables de entorno que mantiene la información de, por ejemplo, donde se encuentran las fuentes del kernel, donde están las cabeceras del propio kernel o donde queremos que nos instale los módulos el proceso de intalación.

Una vez hemos compilado según nuestras necesidades podemos instalar los módulos del i2c con *make install*. Este proceso nos copia todos los módulos que contiene el proyecto i2c por lo que posiblemente se incluyan muchos más de los que necesitamos para dar soporte a nuestro hardware. Los módulos no usados podemos borrarlos. Un último detalle importante, debemos agregar la siguiente línea al fichero `/etc/modules.conf`:

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

Una vez tenemos listo el i2c ya podemos compilar el lm-sensors. Una vez descomprimido en un directorio, lo siguiente y más importante es modificar el fichero Makefile. Debemos modificar la línea que hace referencia a la variable `I2C_HEADERS`. Debemos poner en ella la localización de los fuentes del i2c, en mi caso como uso debian es la siguiente:

```
I2C_HEADERS=/usr/src/modules/i2c
```

A continuación ejecutamos *make*.

Ahora ya disponemos de los binarios del lm-sensors también. De la misma forma que en el i2c tenemos varias opciones configurables en el Makefile. Por ejemplo podemos modificar el proceso de instalación y modificar el kernel en lugar de generar módulos a parte. Por defecto lm-sensors viene preparado para generar módulos y este es el sistema que yo acabo de explicar. Una vez que tenemos los módulos si realizamos un *make instal* el Makefile nos copiará los módulos donde la variable `MODDIR` indica. Por defecto se trata de `/lib/modules/[nuestro kernel]/misc`. Por último necesitamos ejecutar *depmod -a* para establecer las dependencias de los nuevos módulos.

## Detectando nuestros sensores

Una vez tenemos los drivers instalados disponemos de la herramienta *sensors-detect*. Una vez ejecutada nos pedirá información sobre los buses que queremos rastrear y los sensores de que dispone dentro de cada bus. El resultado en mi máquina ha sido el siguiente:

To load everything that is needed, edit `/etc/modules` and add the modules listed here to it:

```
#----cut here----
# I2C adapter drivers
i2c-isa
# I2C chip drivers
w83781d
#----cut here----
```

Then, run `/etc/init.d/modutils`

To make the sensors modules behave correctly, add these lines to `/etc/modutils/local` and run `update-modules`:

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



Como podemos observar me ha detectado que necesito el módulo i2c-isa y el módulo del sensor w83781d. A continuación podemos probar con la herramienta *sensors* a ver el estado de dicho sensor:

```
pegaso:/etc/modutils# sensors w83781d-isa-*
```

```
w83781d-isa-0290
Adapter: ISA adapter
Algorithm: ISA algorithm
+2.46 V (min = +2.52 V, max = +3.08 V) ALARM
+2.46 V (min = +2.52 V, max = +3.08 V) ALARM
+3.3V: +3.58 V (min = +2.97 V, max = +3.63 V)
+5V: +4.99 V (min = +4.50 V, max = +5.48 V)
+12V: +11.97 V (min = +10.79 V, max = +13.11 V)
-12V: -11.61 V (min = -10.78 V, max = -13.18 V)
-5V: -5.10 V (min = -4.50 V, max = -5.48 V)
fan1: 0 RPM (min = 3000 RPM, div = 2) ALARM
fan2: 4963 RPM (min = 3000 RPM, div = 2)
fan3: 0 RPM (min = 3000 RPM, div = 2) ALARM
Temp/MB: +35.0°C (limit = +60°C, hysteresis = +50°C)
Temp/CPU: +46°C (limit = +70°C, hysteresis = +60°C)
Temp3: +208°C (limit = +60°C, hysteresis = +50°C)
vid: +3.50 V
alarms: Chassis intrusion detection
beep_enable:
Sound alarm disabled
```

Como podeis apreciar mi placa base me informa de la temperatura del procesador, de la placa base y de otro dispositivo que desconozco. Los valores que se aprecian por el programa pueden no ser acordes con los valores que nos muestra el monitor de la BIOS, para eso el programa *sensors* tiene un fichero de configuración donde podemos añadir offsets o variar la fórmula que utiliza dicho programa. EL fichero se llama */etc/sensors.conf*

## Como monitorizar los dispositivos desde Gkrellm

Gkrellm es fácilmente configurable para mostrar información de los sensores instalados en el sistema. Si los drivers están correctamente instalados el kernel publicará su información a través de la estructura de archivos del directorio */proc*. Gkrellm tan solo se encarga de ir extrayendo esta información, si está presente, e ir mostrándola.

Si Gkrellm no encuentra ningún problema nos mostrará una entrada más dentro del menu *Integrados* de su sección de configuración. Una vez en ella podemos ver como ha detectado todos los campos de información que el kernel es capaz de recoger del dispositivo sensor.

En la lista nos apareceran varias líneas que indican valores de temperaturas (*temp*) y varias que indican número de revoluciones de los ventiladores (*fan*). Para indicar a Gkrellm que queremos que muestre uno de esos valores de temperatura hemos de poner el nombre *cpu* en el campo *Etiqueta* del indicador de temperatura que nos interesa. Para el caso de las revoluciones de los ventiladores hemos de escribir la cadena *mb*. Gkrellm también nos permite agregar *offsets* y factores multiplicativos tal como hacíamos en el fichero */etc/sensors.conf*

Eso es todo. Espero que este artículo sirva de algo. Se que me dejo muchos detalles, sin embargo no se que más contaros al respecto. Si teneis sugerencias por favor hacer uso de los comentarios y haremos que este artículo sea un buen punto de partida para instalar *lm-sensors* en linux.

P.D: [Aqui](#)<sup>(6)</sup> podeis ver un screenshot como ejemplo del uso de Gkrellm con *lm-sensors*.

- MAC.



---

**Lista de enlaces de este artículo:**

1. <http://www2.lm-sensors.nu/~lm78/>
2. <http://www2.lm-sensors.nu/~lm78/info.html>
3. <http://www2.lm-sensors.nu/~lm78/download.html>
4. [http://www2.lm-sensors.nu/~lm78/archive/lm\\_sensors-2.6.1.tar.gz](http://www2.lm-sensors.nu/~lm78/archive/lm_sensors-2.6.1.tar.gz)
5. <http://www2.lm-sensors.nu/~lm78/archive/i2c-2.6.1.tar.gz>
6. <http://web.wt.net/~billw/gkrellm/gkrellm.png>

---

E-mail del autor: macal\_ARROBA\_ono.com

**Podrás encontrar este artículo e información adicional en:** <http://bulma.net/body.phtml?nIdNoticia=964>