

# **The Elite's K7s5a mainboard HOWTO**

# Table of Contents

<b><u>The Elite's K7s5a mainboard HOWTO</u></b> .....	<b>1</b>
<a href="mailto:sebfrance@ifrance.com">Sebfrance sebfrance@ifrance.com</a> , <a href="mailto:Andreas.Willenbrink@web.de">Andreas Willenbrink Andreas.Willenbrink@web.de</a> , <a href="mailto:Álvaro.Reguly_nospam-alvaro@reguly.net">Álvaro Reguly_nospam-alvaro@reguly.net</a> .....	1
<a href="#">1. Introduction</a> .....	1
<a href="#">2. The integrated sound board</a> .....	1
<a href="#">3. The mainboard's sensors</a> .....	1
<a href="#">4. The mainboard's LAN adapter</a> .....	1
<a href="#">5. Other</a> .....	1
<a href="#">1. Introduction</a> .....	1
<a href="#">2. The integrated sound board</a> .....	2
<a href="#">2.1 Getting the ALSA drivers</a> .....	2
<a href="#">2.2 Installing them</a> .....	2
<a href="#">2.3 Kernel Approach: The AC'97 Driver</a> .....	3
<a href="#">3. The mainboard's sensors</a> .....	3
<a href="#">3.1 Getting the drivers</a> .....	3
<a href="#">3.2 Installing them</a> .....	4
<a href="#">4. The mainboard's LAN adapter</a> .....	7
<a href="#">4.1 Making it work</a> .....	7
<a href="#">5. Other</a> .....	8
<a href="#">5.1 Links</a> .....	8
<a href="#">5.2 Some words</a> .....	8

# The Elite's K7s5a mainboard HOWTO

Sebfrance [sebfrance@ifrance.com](mailto:sebfrance@ifrance.com),  
Andreas Willenbrink [Andreas.Willenbrink@web.de](mailto:Andreas.Willenbrink@web.de),  
Álvaro Reguly [nospam-alvaro@reguly.net](mailto:nospam-alvaro@reguly.net)

v1.3, 2002-04-19

---

*How to use Elite's K7s5a mainboard under GNU/Linux.*

---

## 1. [Introduction](#)

## 2. [The integrated sound board](#)

- [2.1 Getting the ALSA drivers](#)
- [2.2 Installing them](#)
- [2.3 Kernel Approach: The AC'97 Driver](#)

## 3. [The mainboard's sensors](#)

- [3.1 Getting the drivers](#)
- [3.2 Installing them](#)

## 4. [The mainboard's LAN adapter](#)

- [4.1 Making it work...](#)

## 5. [Other](#)

- [5.1 Links](#)
  - [5.2 Some words](#)
- 

## 1. [Introduction](#)

I got the mainboard for christmas 2001, and after a little while, I understood that the new chipset sis735 was not handled by [Mandrake](#) 8.0, [OSS](#) had drivers for it that worked but I didn't really want to pay for it, so I looked on ALSA cvs and found that in their latest sources they had something for the sis 7012, and also find later the drivers for the sensors.

---

## 2. The integrated sound board

### 2.1 Getting the ALSA drivers

So we'll compiling the latest sources from ALSA, which should work for any other distrib as well... just the kernel sources will change because mandrake uses specific patches.

You may get your kernel sources from your distribution or from <http://www.kernel.org>

Here can be found Mandrake sources for the kernel used in 8.0:

<ftp://ftp.univ-savoie.fr/pub/Linux/Mandrake/8.0/i586/Mandrake/RPMS/kernel-source-2.4.3-20mdk.i586.rpm>

and grab the tarball from ALSA:

<ftp://ftp.alsa-project.org/pub/driver/alsa-driver-0.9.0beta10.tar.bz2>

### 2.2 Installing them

Install your kernel sources, in my case:

```
rpm -ivh kernel-source-2.4.3-20mdk.i586.rpm
```

Then decompress alsa drivers:

```
bzip2 -d alsa-driver-0.9.0beta10.tar.bz2 && tar -xvf alsa-driver-0.9.0beta10.tar
```

Make them and install them and create the devices files:

```
cd alsa-driver-0.9.0beta10 && make install && ./snddevices
```

Edit `/etc/modules.conf` to set everything, and add to it the following lines:

```
alias char-major-116 snd
alias char-major-14 soundcore
alias sound-slot-0 snd-card-0
alias sound-service-0-0 snd-mixer-oss
alias sound-service-0-1 snd-seq-oss
alias sound-service-0-3 snd-pcm-oss
alias sound-service-0-8 snd-seq-oss
alias sound-service-0-12 snd-pcm-oss
alias snd-card-0 snd-card-intel8x0
```

## 2.3 Kernel Approach: The AC'97 Driver

If you are using a 2.4 or later kernel you can use the [Audio Codec '97 \(AC'97\)](#) sound driver, as the integrated sound card is AC'97 compliant.

You may enable your soundcard with a kernel module, as a matter of fact, you should be able to just use the module without even recompiling your kernel, because most GNU/Linux distributions have it already, just type:

```
modprobe i810_audio
```

and you should see something like this in your syslog:

```
Intel 810 + AC97 Audio, version 0.21, 21:31:04 Apr 15 2002
i810: SiS 7012 found at IO 0xd800 and 0xdc00, IRQ 11
i810_audio: Audio Controller supports 2 channels.
ac97_codec: AC97 Audio codec, id: 0x414c:0x4710 (ALC200/200P)
i810_audio: AC'97 codec 0 supports AMAP, total channels = 2
```

or type

*dmesg / less* and scroll to see the kernel messages.

If everything went fine, you may add `i810_audio` to `/etc/modules` so it will autoload everytime you boot:

```
echo "i810_audio" >> /etc/modules
```

or you like monolithic kernels (no modules), follow the step above to install the kernel sources and say Y to Sound card support and Y to Intel ICH (i8xx) audio support compile your kernel, install, reboot and now your integrated soundcard is working.

If you don't know how to compile a kernel, read the [Kernel HOWTO](#), it's easy and you'll get a optimized kernel for you system, also you'll learn a bit about that talk of using the source code (yes you are already taking advantage of it :-)

---

## 3. [The mainboard's sensors](#)

### 3.1 Getting the drivers

Get the `lm_sensors` 2.6.2 or more recent: [http://www.lm-sensors.nu/archive/lm\\_sensors-2.6.2.tar.gz](http://www.lm-sensors.nu/archive/lm_sensors-2.6.2.tar.gz) You'll need kernel sources too in order to compile that one.

## 3.2 Installing them

```
tar -xzf i2c-2.6.2.tar.gz
```

```
cd i2c-2.6.2
```

```
make && make install
```

Then there was a problem when making install for me, so if it doesn't work, just do the following:

```
cp kernel/i2c-core.o /lib/modules/`uname -r`/misc/i2c-core.o
```

```
cp kernel/i2c-proc.o /lib/modules/`uname -r`/misc/i2c-proc.o
```

Now let's copy the specific driver:

```
cp kernel/chips/it87.o /lib/modules/`uname -r`/misc/it87.o
```

ok now

```
cp prog/sensors/sensors /usr/bin/sensors
```

---

Edit `/etc/modules.conf` and add to it the following lines:

```
alias char-major-89 i2c-dev
```

```
options it87 temp_type=0x31
```

now:

```
modprobe i2c-proc
```

```
modprobe it87
```

```
modprobe i2c-isa
```

```
sensors
```

should output you some data

When it works, you can write it to a file, that will be loaded automatically on boot. Under [SuSE](#) the file is: `/etc/init.d/boot.local` (used to be `/sbin/init.d/`) and you should just add: `sensors -s` Under mandrake, the install should have created that file: `/etc/init.d/sensors`, which you can edit freely, `sensord` being just a daemon to watch over temperature and fan speed, I assume no one will need something like that for her/his own machine.

Now as root you should edit your `/etc/sensors.conf` search for that line: `chip "it87-*` and below you should be able to modify everything to get the right data, here is a copy of what i have in there:

## The Elite's K7s5a mainboard HOWTO

(please note that temp1 is trash and only temp2 and temp3 should be kept)

---

```
chip "it87-*"

# The values below have been tested on Asus CUSI, CUM motherboards.
# Voltage monitors as advised in the It8705 data sheet

label in0 "VCore"
label in1 "Vcc2.5v"
label in2 "Vcc5v"
label in3 "+5V"
label in4 "+12V"
label in5 "-12V"
label in6 "-5V"
label in7 "SB 5V"
label in8 "V battery"

set in0_min 1.5 * 0.95
set in0_max 1.5 * 1.05
set in1_min 2.4
set in1_max 2.6
set in2_min 3.3 * 0.95
set in2_max 3.3 * 1.05
set in3_min 5.0 * 0.95
set in3_max 5.0 * 1.05
set in4_min 12 * 0.95
set in4_max 12 * 1.05
set in5_min -12 * 0.95
set in5_max -12 * 1.05
set in6_min -5 * 0.95
set in6_max -5 * 1.05
```

## The Elite's K7s5a mainboard HOWTO

```
set in7_min 5 * 0.95

set in7_max 5 * 1.05

# vid not monitored by IT8705F

ignore vid

# For this family of chips the negative voltage equation is different from
# the lm78. The chip uses two external resistor for scaling but one is
# tied to a positive reference voltage. See ITE8705/12 datasheet (SIS950
# data sheet is wrong)

#  $V_s = (1 + R_{in}/R_f) * V_{in} - (R_{in}/R_f) * V_{ref}$ .
#  $V_{ref} = 4.096$  volts,  $V_{in}$  is voltage measured,  $V_s$  is actual voltage.

compute in2 (1 + 1)*@ , @/(1 + 1)

compute in3 ((6.8/10)+1)*@ , @/((6.8/10)+1)

compute in4 ((30/10) +1)*@ , @/((30/10) +1)

# The next two are negative voltages (-12 and -5).
# The following formulas must be used. Unfortunately the datasheet
# does not give recommendations for  $R_{in}$ ,  $R_f$ , but we can back into
# them based on a nominal +2V input to the chip, together with a 4.096V  $V_{ref}$ .

# Formula:
# actual  $V = (V_{measured} * (1 + R_{in}/R_f)) - (V_{ref} * (R_{in}/R_f))$ 
# For -12V input use  $R_{in}/R_f = 6.68$ 
# For -5V input use  $R_{in}/R_f = 3.33$ 

# Then you can convert the formula to a standard form like:

compute in5 (7.67 * @) - 27.36 , (@ + 27.36) / 7.67

compute in6 (4.33 * @) - 13.64 , (@ + 13.64) / 4.33

compute in7 ((6.8/10)+1)*@ , @/((6.8/10)+1)
```

```
# Temperature

    label temp3      "Processor"
    set   temp3_over  40
    set   temp3_hyst  20

    label temp2      "Mainboard"
    set   temp2_over  45
    set   temp2_hyst  25

    ignore temp1

# Fans

    label fan1"Processor's Fan"
    set   fan1_min 2000

    ignore fan3

    ignore fan2
```

---

## [4. The mainboard's LAN adapter](#)

### 4.1 Making it work...

If your board came with the optional on-board LAN card, you have a Silicon Integrated Systems [SiS] SiS900 10/100 Fast Ethernet card, this card is supported by Linux kernels 2.4 and later (that I know of, as I only use 2.4 kernels).

To make it work you should type

```
modprobe sis900
```

and you should see something like this in your syslog:

```
sis900.c: v1.08.02 11/30/2001
eth0: SiS 900 PCI Fast Ethernet at 0xd400, IRQ 5, 00:d0:09:ea:7a:98.
```

or type

`dmesg / less` and scroll to see the kernel messages.

If everything went fine, you may add `sis900.o` to `/etc/modules` so it will autoload everytime you boot:

```
echo "sis900" >> /etc/modules
```

or compile it in your kernel, where you should say Y to SiS 900/7016 PCI Fast Ethernet Adapter support in your kernel configuration.

---

## 5. [Other](#)

### 5.1 Links

**Warning:** If your board is working fine, then don't fix it. Anyway here are the cool links:

- [Elite Group \(new BIOS\)](#)
- [Overclockers Work Bench](#)
- [ECS K7s5a FAQ](#)
- [french guide for the K7S5a](#)

### 5.2 Some words

Using a modified BIOS, I was able to successfully overclock my Athlon 1000Mhz to 1050Mhz flawlessly, but when I messed with `hdparm` (30Gig Western Digital hard drive) using `hdparm -c1 -X66 /dev/hda`, I managed to corrupt my filesystem, so beware.

If you are a real tweaker, you might want to change your BIOS Logo and stuff, some people tried that and it worked OK, some others not, so they had to flash their BIOS again (blindly with a bootdisk and a working bios in it) to make the board work again. It is really up to you. Search Google or OC Workbench and you'll be on your way.

Good Luck! :-)

---