

Running Linux on a Fujitsu-Siemens C1020

Wouter Verhelst

January 9, 2004

Contents

1	Revision History	2
2	Introduction	2
3	the X Window System	2
4	Network	3
5	Hard disk	3
6	USB	3
7	CardBus/PCMCIA	4
8	FireWire	4
9	Sound	4
10	Power Management	5
11	Built-in 802.11 Wireless	6
12	IrDA	6
13	Other hardware	8
	13.1 Modem	8
	13.2 Red optical connection at right-hand side	8
14	Legal stuff	9

1 Revision History

Description	Date
Initial version	September 3, 2003
FireWire update	Oktober 4, 2003
Modem and S/PDIF update; some grammar fixes and general cleanups	December 4, 2003
IrDA update	January 5, 2004
Recommend cpufreq instead of using ACPI P-states	January 9, 2004

2 Introduction

Hi there!

In the document about my previous laptop, at <http://people.debian.org/~wouter/laptop/>, I say that employers rule, especially if they buy you laptops. That's even more true if you're self-employed, and have to buy a (tax-deductible, as it's a professionally required tool) laptop.

Which is what happened to me. I'm a self-employed IT professional now. And since I like laptops, I bought myself one. Not only is it a nice tool, it also comes in handy in case you're at a customer, trying to figure out what's going wrong.

Of course, a laptop is much more interesting if you're running the operating system on it which you know best, which, in my case, is Debian GNU/Linux. This text tries to document how to install and configure Debian GNU/Linux on a Fujitsu-Siemens C1020, although other distributions should work as well.

I'm running Linux 2.4.21, 2.4.22, and some kernels of the 2.6.0-test series at the moment. For the most part, there are no differences due to that fact; when there are, you'll know.

3 the X Window System

The laptop contains a "S3 ProSavageDDR K4M266", which is perfectly supported in XFree86 4.2 and above. If you're running Woody, that means you have to install a back-port; have a look at <http://www.apt-get.org> to find one. Alternatively, you could download the driver's source from the author's website, which is linked to from the documentation section at <http://www.xfree86.org/>, but I do not recommend that.

You have to pick the "savage" driver module.

One minor gotcha: by enabling `CONFIG_AGP` and `CONFIG_AGP_VIA`, you can use the AGP Graphical Aperture, but you'll need to supply the `'agp_try_unsupported'` option to make it load. It works here (as in: it didn't fry my system yet), but the option is probably there for a reason; if you want to make sure nothing happens by doing that, ask around first.

There's a driver with 3D acceleration available from VIA/S3. The source of this driver is at <http://www.linux.org.uk/~alan/S3.zip>, or in binary form from <http://www.viaarena.com/>, with some persistent searching.

However, that driver only supports the kernel shipped with RedHat 8; as this kernel is quite

heavily patched (and as the driver requires some extensive kernel support), this driver doesn't currently compile on systems running a different kernel, unfortunately.

The laptop comes with a Synaptics touch pad. Although that works perfectly well with the usual mouse driver (at least under 2.4), there's a special driver module for X, which you can download at <http://w1.894.telia.com/~u89404340/touchpad/index.html>¹. It gives you some extra stuff, such as allowing you to scroll (button 4/5 and 6/7 events) by sliding your finger near the edge of the touch pad, and makes the hardware scroll button (the one between your hardware left and right mouse buttons) work. There's a bit of documentation in the tar-ball you can download, which includes pretty straightforward installation instructions.

If you're using 2.6, note that you need to load the 'evdev' module before your touch pad will *ever* work. Also, if you're using both 2.6 and 2.4, I recommend to make sure the evdev module loads at boot, whilst using the "auto-dev" protocol and "/dev/psaux" device; if you do that, you don't have to reconfigure your X server after every reboot.

4 Network

```
CONFIG_NET_ETHERNET
CONFIG_NET_PCI
CONFIG_8139T00
```

Well, the options say it all. There's a RealTek 8139 in this thing. Those work, but it's the worst low-end network card one could imagine. That doesn't mean the driver doesn't work, only that the RealTek 8139 chip set is broken by design; it eats a lot of CPU cycles.

Not that you'd notice that on a laptop powered by a Pentium4, but that's a different matter – at least for now ;-)

5 Hard disk

```
CONFIG_BLK_DEV_VIA82CXXX
```

The hard disk will work without the above option, but I *highly* recommend you use it, if you want a decent speed. Also, you might want to add a kernel boot option "ide0=ata66" to make it go even faster. See the man page of grub or lilo for information on how to do that.

6 USB

```
CONFIG_USB_UHCI
```

Plus, of course, the drivers for any USB hardware you might have. I usually compile drivers for all possible and impossible USB device as a module; that way, it doesn't waste any space

¹Yes, that's the same URL as the one mentioned in the help text for the PS/2 mouse driver in the 2.6 kernel

in my laptop's memory, nor do I have to recompile should I ever choose to buy or borrow some USB device.

When I first loaded the driver for the USB root hub, I noticed that there were *three* USB root hubs in my laptop. As it turned out, that's because the built-in wireless LAN has a USB interface, too. Counting them, we get:

1. The two USB slots at the back,
2. The built-in wireless LAN,
3. The port replicator

Which makes up for three USB root hubs. Getting the built-in wireless to work is a joke by itself; read on.

7 CardBus/PCMCIA

CONFIG_CARDBUS

The all-known CardBus/yenta interface. Not only are there no problems here, I think it's hardly possible to create *any* CardBus slot which is unsupported. Luckily.

8 FireWire

CONFIG_IEEE1394_OHCI1394

I have no FireWire devices, but a friend of mine has a digital camcorder with a FireWire-interface. When I was there once, I hooked the camera up to my laptop, and at least doing some video-capture using `dvgrab` worked flawlessly. I also tried to do something else using `'coriander'`, which did not work. However, I don't know whether the problem at hand was

- me not knowing enough about it (after all, I played with it for only about 10 minutes),
- The kernel not supporting some particular bit of my hardware
- `coriander` not supporting the particular camera my friend owns.

At least I can tell the interface works; you'll have to find out about the rest by yourself.

9 Sound

The sound card is integrated in the (VIA) South Bridge. There's a driver for another VIA South Bridge in the kernel (2.4), but it doesn't work on this one – at least not if you want to understand what the speakers do.

With ALSA, however, you can get it to work². The module you need is the “via82xx” one; it works alright, but the built-in speakers suck. If you want good quality, get yourselves headphones or external speakers :-)

10 Power Management

One important part of a laptop is the ability to use it “on the road”. In Linux 2.4, you have the choice between using APM or ACPI. Using APM should mean you’ll be able to suspend the thing, but then you’ll be running without SpeedStep, so for continued work, your battery will be empty fairly fast. Using ACPI means you won’t be able to suspend the machine (at least not in 2.4; 2.6 has support for that), but if you apply the patches from the ACPI4Linux Project at <http://acpi.sourceforge.net/>, you’ll be able to switch between P (SpeedStep) and T (throttling) modes; that means you’ll be able to run a lot longer on a single battery for continued work.

Of course, if you’re running 2.6, you won’t have to worry about this as much, since ACPI support in 2.6 is a lot better; plus, in 2.6, you don’t even need ACPI support for SpeedStep.

In any case, switching to a slower mode using ACPI is done by echoing two numbers to `/proc/acpi/processor/CPU0/limit`. I’m doing this for battery mode:

```
echo '1:5' > /proc/acpi/processor/CPU0/limit
```

and this for AC adapter mode:

```
echo '0:0' > /proc/acpi/processor/CPU0/limit
```

which means ‘run at full processor power when the AC adapter is connected, and at reduced SpeedStep mode, with 31% throttling when it is not’.

You’ll want to install `acpid` for complete ACPI support; however, it seems that the battery does not issue an event. I’ve fixed this by checking whether we’re on AC power at boot time, and switching the system accordingly. That Works For Me, but YMMV.

OTOH, if you’re running 2.6, there’s a neat new feature in the kernel, called the “CPU Frequency Scaling”, which will do the switching of processor speed fully automatic, in kernel space, with no need to configure anything. You want `CONFIG_CPU_FREQ`, `CONFIG_CPU_FREQ_GOV_PERFORMANCE`, `CONFIG_CPU_FREQ_GOV_POWERSAVE`, and `CONFIG_X86_P4_CLOCKMOD`. Additionally, you may be interested in setting frequencies manually, for which you need the userspace governor.

Now, if you run on battery power, the kernel will automatically select the “powersave” governor, and when you connect the system to a power source of some kind, the “performance” governor will be selected. If you compile governors as modules (I don’t recommend that), make sure to load the additional module before you (dis)connect your system to/from a power outlet.

You can view the current processor speed in `/proc/cpuinfo`, and configure the driver through files in `/sys/devices/system/cpu/cpu0/cpufreq`

The buttons (lid and power) do work, OTOH. Create a file in `/etc/acpi/events` with a contents such as

²Which means I had to “mess” with ALSA this time. Have a look at my other laptop-document if you don’t know what I’m talking about

```
event=button[ /]lid
action=/etc/acpi/powerbtn.sh
```

and a file called `/etc/acpi/powerbtn.sh` with a contents like

```
echo '1' > /proc/acpi/sleep
```

to make your system suspend to RAM when you close the lid. Or, if you're using 2.6, make the contents of the latter file look like this:

```
echo '4' > /proc/acpi/sleep
```

to have it hibernate. Which, by the way, is pretty cool :-)

11 Built-in 802.11 Wireless

There are some incarnations of this laptop that feature an integrated IEEE 802.11 Wireless Ethernet connection. It's connected to the system through a USB interface, and is supposed to have two antennas integrated in the LCD case, which should give it quite a range.

These are supported by the "Linux-wlan" project, at <http://www.linux-wlan.org/>; you'll want the "linux-wlan-ng" flavour.

Unfortunately, the drivers aren't very stable yet, and only work with a vanilla kernel. If you apply the ACPI patches, you'll have two possibilities: when you try to load the modules under X `modprobe` will segfault(!), and the relevant module gets stuck in an "initializing" state. When trying from console, the result is a kernel panic.

Without the ACPI patches, however, it seems to work. I think that's only a minor inconvenience, since the extra features the ACPI patches give you are most interesting when you're on the road, where it's likely you won't have any wireless network in the neighbourhood anyway. That doesn't mean it isn't inconvenient, but it makes it a bit less of a headache.

What's more of a problem, is that the drivers can't seem to handle a sustained high data transfer rate very well. If I try to throw a lot of data through it, my keyboard-interrupt gets lost somewhere, and the wireless link is dropped as well. Which is bad, because it doesn't come up again.

Also, the drivers don't work with 2.6 (yet, I presume), so going that road won't help either.

12 IrDA

```
CONFIG_ISA
CONFIG_NSC_FIR
```

I *had* tried to get this to work, since I have a Nokia 6310i GSM, but didn't reach anything useful. There's a FIR controller in the laptop, which, according to Fujitsu-Siemens' phone

support, is a “HP 2300 3600” controller. I’m not sure whether that’s spelled right, but in any case, I couldn’t find any reference to this on the ’net, or in the IrDA-HOWTO.

I finally gave up trying to get the IrDA controller to work. Fortunately, Michael Mráka <mailto:michael.mraka@linux.cz> was more persistent than me, and did in fact reach success. The key to the puzzle was that I didn’t expect the nsc-ircc driver to require ISA to function correctly, but apparently it does³. Oh well.

It’s not entirely straightforward, though. The procedure is as follows:

- Unless you modified any serial settings in your BIOS, the IrDA port is at /dev/ttyS1, IO port 0x2f8, IRQ 3. Check that⁴; if this assumption is wrong, both your serial port and IrDA interface will not work.
- You’ll have to perform some black magic with the module arguments to make sure the “plain” serial driver will not run away with the resources the nsc-ircc driver needs. Doing that is slightly different on 2.4 vs 2.6 kernels; so, you’ll have to choose now:
 - For 2.6, edit or create the file /etc/modprobe.d/local (on a Debian system; if you’re unfortunate enough to use something else⁵, go for /etc/modprobe.conf). The file should contain the following information:

```
alias irda0 nsc-ircc
alias char-major-4 8250
install nsc-ircc /sbin/modprobe 8250; /sbin/modprobe --ignore-install nsc-ircc
remove nsc-ircc /sbin/modprobe -r --ignore-remove nsc-ircc; /sbin/modprobe -r 8250
install 8250 /sbin/modprobe --ignore-install 8250; /bin/setserial /dev/ttyS1 ua
```

After that, on Debian, run “update-modules” to update the master file (other distributions should work right away).

- For 2.4, edit or create the file /etc/modutils/local (on a Debian system; if you’re using something else, go for /etc/modules.conf). The file should contain the following information:

```
alias irda0 nsc-ircc
alias char-major-4 serial
pre-install nsc-ircc /sbin/modprobe serial
post-remove nsc-ircc /sbin/modprobe -r serial
post-install serial /bin/setserial /dev/ttyS1 uart none port 0x0
```

- That was the hardest part. Next, you’ll need to install the “irda-common” package⁶.
- Once all that is done, you’re all set. If you want to use the IrDA stack, you can start it using “irattach irda0 -s”. Another option would be to have this starting at every boot

³Can you blame me? PCI exists since at least as long as 1994; after ten years, one would assume that hardware manufacturers could create hardware which does not require ISA to function correctly, so I did not switch ISA on in my custom-compiled kernel. Apparently that was an incorrect assumption... unless this is Linux-specific, in which case some kernel people really need to rewrite some parts of the kernel :-)

⁴by running “dmesg” after modprobe’ing your serial driver (either “serial” if you’re running 2.4, or “8250” for 2.6), and checking the output

⁵just kidding, of course

⁶on a Debian “unstable” system; other distributions might call that package differently; “irda-tools” is a good guess – you’re looking for the package containing “irattach”

(most distributions support this), but I recommend against doing so; loading the IrDA stack and running IrDA discovery all the time will deplete your battery rather fast. It's probably better to have it disabled until you're actually going to use it...

After doing all of the above, I could see my mobile phone in the discovery log, and I got ping-replies with "irdaping", so I guess it works. That doesn't mean I actually understand the ins and outs of the IrDA subsystem now, but at least I can be reasonably sure the hardware works now :-)

13 Other hardware

So far for the good news. What follows is hardware which is available, but which I either tried to get to work, but could not, or did not try at all. If any of this is outdated, incorrect, or documented somewhere, please let me know. Thanks.

13.1 Modem

There's a software modem in this machine. I haven't tried to get it to work yet; it might work, but it might just as well not work. Feel free to let me know how to do it in case you find out.

I tried using the drivers from linuxant, and they don't explicitly tell me there's no modem inside, but when I try to send AT-commands to the modem, I get no response either. I didn't try to actually dial out, but if I don't even get any response to AT commands, well...

Yes, I know that 'the other one' out there with a document such as this one did succeed in getting the thing to work. But his information is, uh, like, incomplete. To say the least. And since he doesn't provide an email address either, contacting him is pretty hard.

I've had mails from people claiming that with the Linuxant driver, version 6.03.00lnxt03091800full, the modem would work straight away. This would not be the case for the V5 driver, as it always responds "NO DIALTONE" when trying to dial.

I must add that I haven't tried this myself (didn't have the time yet), but I have no reason to assume it would be incorrect; for that reason, I did not yet move this section to somewhere else.

13.2 Red optical connection at right-hand side

This document used to say "At the right-hand side of the laptop, behind the CD-RW/DVD drive, there's a red led in the case. I don't know what it's called, but it is an optical connection of some sort."

The nice bit about writing documents such as this one is that people tend to send you mails explaining what stuff is, if you don't know by yourself..

I've had two people mail me that the optical connection at the right-hand side is an S/PDIF optical audio output. You'd need an amplifier with optical inputs (I, myself, never heard of

that), while apparently some digital recorders (such as a minidisc) have a connector that can do stuff with such output.

Well, now I know what it is; if someone could tell me whether it works, I would be a happy man.

14 Legal stuff

This document is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

If you downloaded the tarball of this document, you should have found the text of the GPL in there. If not, it's at <http://www.gnu.org/licenses/gpl.txt>

You can contact me at <mailto:wouter@grep.be>