UK Astronomy Technology Centre

Horst Meyerdierks
24 March 2003

Toshiba
Satellite Pro
4600

Summary

    * The laptop is dual-boot Windows 2000 Professional and Red Hat 7.1 [9] with the 2.4.9-34 kernel. Red Hat's 2.4.2 and 2.4.3-12 kernels work without problem.
    * X11 works, but accelerated only with XFree86 4.2.0 and a driver newer than that. There is a choice of three configuration files in the relevant section below.
    * Red Hat 6.2 (GNU/Linux 2.2.17) with XFree86 4.0.2 [14] worked moderately well, so long as you did not need the built-in Ethernet card or the sound card.
    * IrDA did not work under Red Hat 6.2, and apparently does not under 7.1 either.
    * The CD writer works fine, once you have figured out how ATAPI drives can be treated as SCSI drives.

Contents

Hardware

These are the features of the hardware:

    * Toshiba Satellite Pro 4600 PIII800/128/20/CD-RW+DVD/15"TFT/WiFi/NT4/W2000
    * Pentium III, 800 MHz
    * 256 MByte memory
    * 18.6 GByte disc (20000 million Byte)
    * combined CD re-writer (4x) and DVD reader (6x, CD at 24x)
    * internal 1.44 MByte floppy drive
    * 15 inch 1024x768 pixel TFT LCD
    * 16 MByte Trident CyberBlade XP graphics card
    * output to external monitor or analogue video (PAL/NTSC)
    * Yamaha YMF743 16-bit stereo sound card
    * internal V.90 modem: Toshiba Software Modem AMR
    * internal 100BaseTX Ethernet interface: Intel PRO/100 VE
    * WiFi wireless LAN: Toshiba Wireless LAN Mini PCI Card
    * IrDA interface: SMC IrCC FIR port
    * 2 PCMCIA card slots: Texas Instruments PCI-1410 CardBus Controller
    * ports: PS/2, serial, parallel (ECP), 2x USB

Also these peripherals:

* D-Link DFE-650TX PCMCIA 100BaseTX Ethernet card
* Adaptec SlimSCSI APA-1460D PCMCIA FastSCSI card
* Logitech mouse with 2 buttons and scroll wheel, USB with USB-to-PS/2 adapter
* Nokia 7110e mobile phone and modem, connected by DLR-3 serial cable [8]
* Nokia 6310i mobile phone and modem, same cable appears to work.

BIOS settings

You can access the BIOS (on all Toshiba laptops) by keeping the Escape key pressed while the red 'TOSHIBA' comes onto the screen at power-up or re-boot. Power-up mode is 'Boot' already. I change the PC Card Controller Mode from 'Auto-Selected' to 'PCIC Compatible'. And it may be useful to change the Pointing Devices to 'Simultaneous'.

I also like to use the set the LCD to semi-bright in the BIOS. However, each time Windows 2000 shuts down it resets this to super-bright. So I have de-installed some of the power management software in the Toshiba Windows 2000 system.

Dual-boot installation

When first switching on, you have to choose either Windows NT Workstation 4.0 or Windows 2000 Professional. Windows 2000 has a 16.5 GByte C: drive in partition /dev/hda2. The remaining 2 GByte would have contained the NT system in /dev/hda1, but this is now an extended partition containing the logical partition /dev/hda5 (drive D:). So I destroy this installation and restore from the Toshiba recovery CD. This ends up using the whole disk as FAT32 partition.

To get a dual-boot system, you need software that can reduce the size of the partition without destroying it. Or you have to back up the system, change the partitioning and restore the system into a smaller partition. I keep 2 GByte (/dev/hda1) for Windows 2000. As long as the file system has not been converted to NTFS, GNU parted [15] can be used to resize the Windows partition.

The next 2 GByte (/dev/hda2) are the Linux root file system followed by 512 MByte of swap (/dev/hda3). That leaves just over 14 GByte for an extended partition /dev/hda4, in which I have two equal-sized logical partitions for /home (/dev/hda5) and for data (/dev/hda6). The Windows 2000 partition (/dev/hda1) is also mounted by the Linux system.

The main features of the Linux system are:

  * Red Hat 7.1
  * Linux kernel 2.4.9-34 from Red Hat
  * XFree86 4.2.0 from XFree86

I have experimented with VMware [13]. This is installed as an application on the 'host' operating system and emulates PC hardware for one or more 'guest' operating systems. The host can be Linux, Windows NT or Windows 2000. The guest can be any of these, Windows 95 or Windows 98, probably also others. I used Windows 2000 as guest on Red Hat 6.2 as host.

The advantage is that you can run both systems at the same time, or you can run the host system only. There are disadvantages, too. You should have enough memory for both. The guest hardware is slow since it does not use the hardware directly, but through the host system. E.g. the guest 'disc partition' is an ordinary file on the host system, quite big, with a size limit of 2 GByte, possibly fragmented.

It was also unfortunate that the host system (Linux) was less suited to the real hardware than the guest system (Windows 2000) would have been. So in terms of hardware working I got the worst of both worlds.

In the end I went for traditional dual-boot, because my reason for having Windows 2000 is to run a Logitech QuickCam VC, which was a lot less efficient from a guest operating system through the parallel port emulation.

Thomas Smets [11] found his Windows 2000 system un-bootable after installing Linux; he had to use Windows repair procedures to fix the boot sector. While the master boot record is outwith all disc

partitions, any boot sector is inside the partition that the operating system is installed in and that the system is booting from.

Working and non-working hardware
Keyboard

On some previous Toshiba models there was the problem that the keyboard would not work if the gpm daemon was running. This is not a problem on this model. Nonetheless, I don't run the gpm daemon.

X11 graphics

When I got the machine (March 2001) this did not work. It took me a few days to find a configuration. Now I know of three working configuration.

1. VESA driver with XFree86 4.0.2.

My original solution was to run the XFree86 server 4.0.2, but to use neither the trident driver, nor the vga driver. I used the vesa driver, which works up to 1280x960 and up to 16 bpp. The same configuration works for later versions of XFree86 (4.0.3, 4.1.0, 4.2.0).

This configuration works for a number of other unsupported cards, too. Trident or otherwise.

You can obtain the X server from XFree86 [14]. By now Linux distributions will include these as well. Red Hat 7.1 ships with 4.0.3 and there is an update RPM for 4.1.0. Red Hat's Rawhide contains RPMs for 4.2.0 [9]. You can download my /etc/X11/XF86Config-4 file.

2. Trident driver with XFree86 4.2.0

In January 2002 XFree86 4.2.0 was released. This has un-accelerated support for this graphics card in the trident driver. This will also do 24 bpp. 4.1.0 was said to support the card, but did not.

You can obtain the X server from XFree86 [14]. Red Hat's Rawhide contains RPMs for 4.2.0 [9]. You can download my /etc/X11/XF86Config-4 file.

Martin Gräfe [16] sent me a new version of the trident driver. Place this in /usr/X11R6/lib/modules/drivers and you have accelerated support.

3. SVGA server from XFree86 3.3.6

Jean-Marie Teuler [12] sent me a configuration that makes the card work with the SVGA server in XFree86 3.3.6. Only up to 16 bpp, but this is still an interesting option: Version 3.3.6 ships even with older Linux distributions and I could imagine that this might run more efficiently than the un-accelerated or CPU-based trident and VESA drivers in 4.2.0. I have only briefly tried this to confirm that it works at all. You can download the /etc/X11/XF86Config file. This is Jean-Marie's file, except I britishised the keyboard and replaced the use of the font server with a list of directories to search for fonts.

There seem to be some possible hick ups with the configuration file. One is that an XF86Config* file in /root (home directory of the superuser) may be used in preference to one in /etc/X11. Another is that downloading under Windows may change the double quotes in the file to something else that is invalid when the X server reads it. Also, downloading under Windows may introduce additional Return characters at the end of each line, which the X server may not tolerate in its configuration file.

In my experience the file for version 3.3.6 has to be /etc/X11/XF86Config, while for 4.x is should be /etc/X11/XF86Config-4. I managed to switch the server by changing the symbolic link /etc/X11/X. For version 4.x it points to ../../usr/X11R6/bin/XFree86. For version 3.3.6 there can be several different servers and the one you want may be missing. The most common server is SVGA, and to use it /etc/X11/X needs to point to ../../usr/X11R6/bin/XF86_SVGA. If you run Red Hat and need to add a server, there is an RPM for each of them, e.g. XFree86-SVGA-3.3.6-38.i386.rpm.

Two- and three-button pointers

I don't particularly like the built-in pointer. So I always carry an additional mouse and use it when space allows. As it happens, I have

a few choices here. The mouse has a wheel that can serve as middle button, and it is a USB mouse with a PS/2 adapter, so can be plugged in as one or the other. Also, it is a Logitech mouse rather than a Microsoft mouse, which means the wheel works as middle button without changing to IMPS/2 protocol or setting a ZAxisMapping.

One choice I don't have: The mouse must be connected before the system is booted. If I plug it in later it will be ignored. If the mouse is on PS/2, un-plugging it and plugging it back in seems to upset the X server considerably. If the mouse is on USB, disconnect and re-connect is fine.

The configuration I use is based on one that Helmut Bluhm [1] sent me. There are sections in the X configuration for two distinct mice. One is on /dev/mouse, which is symbolically linked to /dev/input/mice and corresponds to an external mouse on the USB port. The other is the common /dev/psaux, which is either the built-in pointer or an external mouse on the PS/2 port.

The ServerLayout section in the X configuration makes use of both mouse sections. If I plug the external mouse into PS/2 it overrides the built-in pointer, but if I plug it into USB it works simultaneously with the built-in pointer.

To make this work, /dev/mouse must not be a symbolic link to /dev/psaux (as it often is), but to /dev/input/mice:

```
  $ ls -l /dev/mouse
    lrwxrwxrwx        10  /dev/mouse -> input/mice
```

This makes /dev/mouse the USB mouse leaving /dev/psaux for the built-in pointer and PS/2 mouse. In XF86Config-4 the relevant sections are [1]:

```
  Section "InputDevice"
    Identifier  "USBMouse"
    Driver      "mouse"
    Option "Protocol"    "IMPS/2"
    Option "Device"      "/dev/mouse"
```

```
  Option "Emulate3Buttons"
  Option "Emulate3Timeout" "150"
  Option "ZAxisMapping" "4 5"
EndSection

Section "InputDevice"
  Identifier  "Internal"
  Driver      "mouse"
  Option "Protocol"    "IMPS/2"
  Option "Device"      "/dev/psaux"
  Option "Emulate3Buttons"
  Option "Emulate3Timeout" "150"
  Option "ZAxisMapping" "4 5"
EndSection

...

Section "ServerLayout"
  Identifier  "Simple Layout"
  Screen      "Screen 1"
  InputDevice "Internal"  "CorePointer"
  InputDevice "Keyboard1" "CoreKeyboard"
  InputDevice "USBMouse"  "SendCoreEvents"
EndSection
```

Built-in Ethernet

The Red Hat 7.1 installer recognises the built-in Ethernet card and configures the system to use the e100 kernel module. This worked fine for me. Hermann Gottschalk uses 'alias eth0 eepro100' in /etc/modules.conf [3]. When I was running a home-built 2.4.6 kernel from kernel.org [7], it did not have the e100 module. So I switched to eepro100 and did not switch back when upgrading to Red Hat's own 2.4.9-6. But now I have switched back, because I found the 2.4.9-34 eepro100 module to be very unreliable on a 10 Mbit/s hub, but the e100 module working fine under those conditions. Both seem to work fine on my regular 100 Mbit/s switch.

Bart Friederichs [17] finds that the eepro100 driver does not work well: "The original driver by Donald Becker seems to have some

problems. After a while it starts complaining that it has a wait_for_cmd timeout. Re-cycling (ifconfig down/up) the eth0 then fixes it (for some time)." He has since installed Gentoo and found Intel's e100 driver in it, which works fine.

With Red Hat 6.2 and its 2.2.17 kernel, the card was not recognised. Ed Schaller says "I found that I could get the internal Ethernet card to work. The eepro100 kernel module for the 2.2.19 kernels works fine, or you can download the new drivers from Donald Becker and friends as a source rpm from ftp://ftp.scyld.com." [10]

Eduardo Kita uses the internal Ethernet card with an e100 Intel driver under the 2.2.16 kernel: "The way I found the driver is this, starting at the Intel support page: http://support.intel.com/sites/support/index.htm?iid=intelhome+support&. Choose 'Networking' and then 'Adapters for Mobile System'. Select the 'PRO/100 VE'. Click on 'Software & Drivers' on the left frame. Select the Operating System." [6]

Wireless LAN

I don't use the wireless LAN card myself. Jan Newmarch sent me the information that I've attached as a separate document.

PCMCIA

PCMCIA worked fine under Red Hat 6.2, kernel 2.2.17, using the i82365 module. Under Red Hat 7.1, kernel 2.4.2, there are problems. Instead of i82365 the yenta_socket module is used, and this seems to get hooked onto the WiFi card. The PCMCIA bus is completely ignored. This is a problem I have on my own system and below is how I fixed it. But you should initially ignore this, as at least one correspondent has no problem at all in this area. [5]

So we change the PCIC setting in /etc/sysconfig/pcmcia so that the file reads (This is the Red Hat place to control this, other distributions may use /etc/pcmcia.conf or the pcmcia boot script):

```
 PCMCIA=yes
 PCIC=i82365
```

```
PCIC_OPTS=
CORE_OPTS=
```

Alas, this is not sufficient. The Red Hat boot script insists on running kudzu and changing PCIC to yenta_socket. Doh. So you also have to edit the boot script (on Red Hat this is /etc/rc.d/init.d/pcicma). Search for the first line quoted here and un-comment the if/fi block that follows it:

```
    kernel=`uname -r | cut -d. -f1-2`
    if [ $PCIC = i82365 -a $kernel = 2.4 ]; then
       kudzu=$(kudzu -b pci -c socket -p | grep yenta_socket)
 #      if [ -n "$kudzu" ]; then
 #         PCIC=yenta_socket
 #      fi
    fi
```

If you intend to use the WiFi the above is of course a bad idea, but it seems to make the PCMCIA slots work.

IrDA

I wrote this section while using Red Hat 6.2, both with its 2.2.17 kernel and with a 2.4.1 kernel I built from source. Under Red hat 7.1 I have made only one brief attempt at using IrDA, without success. More recently, Paul Hampson [4] has posted software on the Linux IrDA mailing list to make IrDA work on the Tosh 4600.

I have finally given up on IrDA. On previous Toshiba models the IrDA port was a Toshiba Type-O and was handled by the toshoboe kernel module. Later, that module had to be patched for the changed device ID in the Type-DO ports.

According to the Windows 2000 system the chip is in fact an SMC IrCC Fast Infrared Port on an Intel 82801 BAM LPC interface controller 244C. Said controller is on PCI bus 0, device 31, function 0. Windows also lists these resources for the IrDA chip:

```
 I/O range 02F8-02FF
 IRQ     10
```

I/O range 0130-0137
DMA   01

Looking on Linux at /proc/pci we find:

  Bus 0, device 31, function 0:
    ISA bridge: Intel Unknown device (rev 3).
      Vendor id=8086. Device id=244c.
      Medium devsel. Fast back-to-back capable. Master Capable.
No bursts.

The smc-ircc module, however, does not load. Drebes [2] gives an
important hint: hack the drivers/net/irda/smc-ircc.c source code to
hard wire the I/O addresses. That alone does not help, I also fix the
IRQ and DMA:

```
  static unsigned int io[]  = { 0x130, ~0, ~0 };
  static unsigned int io2[] = { 0x2f8, 0, 0 };
  ...
  static int ircc_irq=10;
  static int ircc_dma=1;
```

The module will now load, logging in /var/log/messages:

  IrDA: Registered device irda0
  smc-ircc, Overriding IRQ - chip says 14, using 10
  smc-ircc, Overriding DMA - chip says 13, using 1

and there is an irda0 interface now:

  ifconfig irda0

    irda0 Link encap:IrLAP HWaddr 00:00:00:00
        NOARP MTU:2048 Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:8
        Interrupt:14 Base address:0x2f8

But all is not well: The IRQ may still be 14 and the HWaddr is invalid. When I bring the interface up and then run irdadump and irdaping the system just hangs.

CD writer

The combined CD reader, DVD reader and CD re-writer works without problem as an IDE ATAPI CD reader. CDs and DVDs alike can be mounted as ISO9660 file systems. Audio CDs play, but I don't have any audio or video DVDs.

In order to use the CD writer, the drive must be used as a quasi-SCSI drive. I.e. one must use the ide-scsi kernel module instead of the ide-cd module. The drive then is on device /dev/scd0 instead of /dev/hdc.

In Red Hat 6.2 I solved this by insmod'ing the ide-scsi module from /etc/rc.d/rc.local. This pre-empted any loading of ide-cd, e.g. when a CD was mounted. I also change the symbolic link /dev/cdrom to point to scd0.

With Red Hat 7.1, the installer puts into the Linux section of the lilo configuration the line

  <Tab>append="hdc=ide-scsi"

and it makes the symbolic link to /dev/scd0 as well.

The line in /etc/fstab remains as it was:

 /dev/cdrom /mnt/cdrom iso9660 noauto,user,exec,ro 0 0

When it comes to writing, the cdrecord application does not use the /dev/scd* devices, but the generic SCSI devices /dev/sg*. Best scan for devices to find the device specification you need:

 cdrecord -scanbus

   Cdrecord 1.8 (i686-pc-linux-gnu) Copyright (C) 1995-2000 Jörg Schilling
   Using libscg version 'schily-0.1'

```
scsibus0:
    0,0,0    0) 'SEAGATE ' 'ST11200N SUN1.05' '9500' Disk
    0,1,0    1) 'SEAGATE ' 'ST11200N SUN1.05' '9500' Disk
    0,2,0    2) 'SEAGATE ' 'ST11200N SUN1.05' '9500' Disk
    0,3,0    3) 'SEAGATE ' 'ST11200N SUN1.05' '9500' Disk
    0,4,0    4) *
    0,5,0    5) *
    0,6,0    6) *
    0,7,0    7) *
  scsibus1:
    1,0,0   100) 'TOSHIBA ' 'DVD-ROM SD-R2002' '1324'
Removable CD-ROM
    1,1,0   101) *
    1,2,0   102) *
    1,3,0   103) *
    1,4,0   104) *
    1,5,0   105) *
    1,6,0   106) *
    1,7,0   107) *

  cdrecord speed=4 dev=01,00,00 -data /tmp/mycd.iso
```

Sound

With a Red Hat 7.1 system (hence kernel 2.4.2), sndconfig recognises the card as i810_audio. Hermann Gottschalk [3] gave me this for /etc/modules.conf:

```
  alias char-major-14 off
  alias sound off
  alias midi of
  #
  # YaST2: sound cards support
  #
  alias char-major-116 snd
  options snd snd_cards_limit=1 snd_major=116
  alias snd-card-0 snd-card-intel8x0
  options snd-card-intel8x0 snd_id=card1 snd_index=0
  #
  # YaST2: sound system dependent part
```

```
  #
  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-0-11 snd-mixer-oss
  alias sound-service-0-12 snd-pcm-oss
```

I don't care too much about the sound card, so all I can say is that it plays an audio CD.

Disclaimer

This information is provided in good faith, but no warranty can be made for its accuracy. These pages may be freely pointed to from other Web pages, but you should not copy pages to your own Web site. Myself or the UKATC are not responsible for any information contained on the pages linked to from my pages.

References

  1. Helmut Bluhm, e-mails, 23 May 2001 and 25 May 2001
  2. Roberto Jung Drebes, e-mail to linux-irda@pasta.cs.uit.no, 5 Jan 2001, then archived at http://www.pasta.cs.uit.no/pipermail/linux-irda/2001-January/002229.html
  3. Hermann Gottschalk, e-mail, 19 Apr 2001
  4. Paul Hampson, e-mail to linux-irda@pasta.cs.uit.no, 18 Oct 2001, then archived at http://www.pasta.cs.uit.no/pipermail/linux-irda/2001-October/003262.html
  5. Mathias Herberts, e-mail, 1 June 2001
  6. Eduardo Kita, e-mails, 17 Apr 2001 and 28 May 2001
  7. Linux Kernel Archive, http://www.kernel.org
  8. Nokia, http://www.nokia.com/
  9. Red Hat, http://www.redhat.com/
 10. Ed Schaller, e-mail, 29 Mar 2001
 11. Thomas Smets, e-mail, 3 May 2001
 12. Jean-Marie Teuler, e-mail, 31 January 2002
 13. VMware, http://www.vmware.com/
 14. XFree86, http://www.xfree86.org/

15. GNU, ftp://ftp.gnu.org/gnu/parted/
16. Martin Gräfe, e-mail, 4 September 2002
17. Bart Friederichs, e-mail, 16 March 2003

See also:

   * Werner Heuser, 2002, Linux on a Toshiba Satellite Pro 4600, http://www.tuxmobil.org/toshiba_sp4600.html

Valid HTML 3.2!   Linux on Laptops