HPT370/372/372A ATA RAID Controller

Red Hat Linux

Installation Guide

Version 1.32

Copyright © 2001 HighPoint Technologies, Inc.

All rights reserved.

Last updated on Jun 28, 2002

Table of Contents

1 Overview 1
2 Installing Red Hat Linux on HPT370/372/372A Controller 1
Step 1 Prepare Your Hardware for Installation 1
Step 2 Check System BIOS Settings 1
Step 3 Prepare the Driver Diskette 1
Step 4 Prepare the Kernel Image 2
Step 5 Prepare the Boot Diskette (Red Hat Linux 7.1/7.2/7.3 Only) 2
Step 6 Install Red Hat Linux 3
3 Installing HPT370/372/372A Driver on an Existing System 4
Step 1 Obtain the Driver Module 5
Step 2 Update the Kernel (Red Hat Linux 7.1/7.2/7.3 Only) 5
Step 3 Test the Driver Module 9
Step 4 Configure System to Automatically Load the Driver 9
Step 5 Configure System to Mount Volumes when Startup 11
4 Monitoring the Driver 11
Checking Devices Status 12
Rebuilding a Critical Array 12
Rescanning Devices 12
5 Updating the Driver 13
6 Installing RAID Management Software 14
Checking System Requirements 14
Preparing the Installation Files 14
Installing the Software Package 14
Running the Management Software 15
7 Uninstalling 15
Uninstalling the Driver 15
Uninstalling the Management Software 15

1 Overview

The purpose of this document is to provide clear instructions on how to install and use HPT37x ATA RAID controllers on Red Hat Linux system.

The driver supports following the HighPoint ATA RAID controllers:

HPT370 UDMA/ATA100 RAID Controller
HPT372 UDMA/ATA133 RAID Controller
HPT372A UDMA/ATA133 RAID Controller

2 Installing Red Hat Linux on HPT370/372/372A Controller

If you would like to install Red Hat Linux onto drives attached to HPT37x controller, please perform the following operations:

Step 1 Prepare Your Hardware for Installation

After you attach your hard disks to HPT37x controller, you can use HPT37x BIOS Setting Utility to configure your hard disks as RAID 0, RAID 1, RAID 0/1 or JBOD arrays, or just use them as single disks.

Before installation, you must remove all the disk drives, which are not physically attached to HPT37x controller, from your system.

Note

If you have other SCSI adapters installed, you must make sure the HPT37x controller BIOS will be loaded firstly. If not, try to move it to another PCI slot. Otherwise you may be unable to boot up your system.

Step 2 Check System BIOS Settings

In your system BIOS SETUP menu, change **Boot Sequence** in such a way that the system will first boot from floppy or CDROM, and then from SCSI. Refer to your BIOS manual to see how to set boot sequence.

If your BIOS settings do not support such a boot sequence, you can first set it to boot from floppy or CDROM. After you finish installation, set SCSI as the first boot device to boot up the system.

Step 3 Prepare the Driver Diskette

If you are installing Red Hat Linux 7.0/7.1, just copy all the files under rhdd\dd directory to a dos formatted diskette, label it as driver diskette.

If you are installing Red Hat Linux 7.2, the driver is contained in a floppy diskette image file. Refer to the table below to select the correct diskette image file to use.

rh72dd-athlon.img	Athlon UP system driver diskette image
rh72dd-athlonsmp.img	Athlon SMP system driver diskette image
rh72dd-i686.img	i686 UP system driver diskette image
rh72dd-i686smp.img	i686 SMP system driver diskette image

On a DOS or Windows system, you can make the Red Hat 7.2 driver diskette using rawrite.exe. It can be found on the Red Hat Linux CD (under /dosutils). Just run it under a command window and follow its prompt.

On a Linux system, you can use the " dd" command to make the boot diskette. Insert a floppy disk into the floppy drive and type the command, eg:

for i686 UP system

dd if=rh72dd-i686.img of=/dev/fd0

If you are installing Red Hat Linux 7.3 with Athlon CPU, just copy all the files under rh73dd\athlon directory to a dos formatted diskette; otherwise, copy all the files under rh73dd\i686 directory to a dos formatted diskette.

Step 4 Prepare the Kernel Image

When installing Red Hat Linux 7.1/7.2/7.3 onto hard disk attached to HPT370 or HPT372 controller, copy a kernel to a dos formatted diskette and re name to "**vmlinuz.hpt37x2**", then label it as UP kernel diskette. If installing to SMP system, copy the corresponding SMP kernel image to a dos formatted diskette and rename it to "**vmlinuz.hpt37x2smp**", then label it as SMP kernel diskette.

vmlinuz-2.4.18-3-athlon	RH7.3 Athlon UP system kernel image
vmlinuz-2.4.18-3-athlonsmp	RH7.3 Athlon SMP system kernel image
vmlinuz-2.4.18-3-i686	RH7.3 i686 UP system kernel image
vmlinuz-2.4.18-3-i686smp	RH7.3 i686 SMP system kernel image
vmlinuz-2.4.7-10-athlon	RH7.2 Athlon UP system kernel image
vmlinuz-2.4.7-10-athlonsmp	RH7.2 Athlon SMP system kernel image
vmlinuz-2.4.7-10-i686	RH7.2 i686 UP system kernel image
vmlinuz-2.4.7-10-i686smp	RH7.2 i686 SMP system kernel image
vmlinuz-2.4.2-2	RH7.1 UP system kernel image
vmlinuz-2.4.2-2smp	RH7.1 SMP system kernel image

Step 5 Prepare the Boot Diskette (Red Hat Linux 7.1/7.2/7.3 Only)

If you are installing Red Hat Linux 7.1/7.2/7.3 onto hard disk attached to HPT370 or HPT372 controller, you must boot from a customized boot diskette to start installation. There is no need of customized boot diskette for HPT372A controller.

First obtain the boot diskette image file, rh71boot.img for RedHat 7.1 or rh72boot.img for RedHat 7.2 or rh73boot.img for RedHat 7.3.

On a DOS or Windows system, you can make the boot diskette using rawrite.exe. It can be found on the Red Hat Linux CD (under /dosutils). Just run it under a command window and follow its prompt.

On a Linux system, you can use the " dd" command to make the boot diskette. Insert a floppy disk into the floppy drive and type the command:

RedHat 7.1

dd if=rh71boot.img of=/dev/fd0

RedHat 7.2

dd if=rh72boot.img of=/dev/fd0

RedHat 7.3

dd if=rh73boot.img of=/dev/fd0

Step 6 Install Red Hat Linux

1) Start installing Red Hat Linux by booting with the bootable disks or CDROM. **If you are installing Red Hat Linux 7.1/7.2/7.3 onto hard disk attached to HPT370/372 controller, you must boot from the boot diskette provided for HPT37x driver.**

2) On "**Welcome to Red Hat Linux**" installation screen, a prompted label "**boot:**" will appear at the bottom of the screen. If you are installing Red Hat Linux 7.0/7.1/7.3, type in "**expert text**" (without quotation mark) and then press **enter**. If you are installing Red Hat Linux 7.2, type in "**expert text updates**" (without quotation mark) and then press **enter**.

3) If you are installing Red Hat Linux 7.1/7.2/7.3, you will be asked "**Do you have a driver disk?**". Select "**Yes**".

4) When prompted "**Insert your driver disk and press OK to continue**", insert the driver diskette in the floppy drive and then select "**OK**".

5) If you are installing Red Hat Linux 7.1/7.2/7.3, please **go to step 10** since the system will load HPT37x driver automatically.

6) After the "**Devices**" dialog box appears, select "**Add Device**" option.

7) When asked "**What kind of device would you like to add?**", select "**SCSI**", and then select "**Ok**".

8) Scroll down to "**HPT370/372/372A ATA RAID Controller**", and then select "**Ok**".

9) The installation process will now display the "**HPT370/372/372A ATA RAID Controller**" as been found, select "**Done**".

10) If you are installing Red Hat Linux 7.2, when asked "Insert your updates disk and press ok to continue", just press <Enter> to continue.

11) Continue the installation as normal. You can refer to Red Hat Linux installation guide.

Note

The system device mapping order is the same as the order shown in HPT37x BIOS Setting Utility. If you have no other SCSI adapters installed, the device marked as "BOOT" or "HDD0" will be /dev/sda, "HDD1" will be /dev/sdb, "HDD2" will be /dev/sdc, etc. When creating mount points, you must mount /boot on /dev/sda.

12) When asked where to install lilo, you must select Master Boot Record (MBR) to make your system be able to boot from HPT37x controller. For Redhat 7.2/7.3, you may choose GRUB as system loader.

13) *(HPT370/372 only)* If you are installing Red Hat Linux 7.1/7.2/7.3 onto hard disk(s) attached to HPT370 or HPT372, when "**Installation Complete**" dialog appears, an additional step must be taken to replace the installed kernel with our new kernel, which has HPT366 IDE support removed. On the driver disk, there is a setup script "**postinstall**" which will do this work for you. When the "**Installation Complete**" dialog appears, **DO NOT** press **ENTER**. Press **Alt-F2** to activate the command shell and type in the following commands:

for Red Hat Linux 7.1/7.2/7.3

 # chroot /mnt/sysimage
 # mkdir /mnt/floppy

insert UP kernel diskette

 # mount /dev/fd0 /mnt/floppy
 # cp /mnt/floppy/vmlinuz.hpt37x2 /boot
 # umount /mnt/floppy

insert SMP kernel diskette *smp system only*

 # mount /dev/fd0 /mnt/floppy *smp system only*
 # cp /mnt/floppy/vmlinuz.hpt37x2smp /boot *smp system only*
 # umount /mnt/floppy *smp system only*

insert driver diskette

 # mount /dev/fd0 /mnt/floppy
 # sh /mnt/floppy/ postinstall
 # umount /mnt/floppy

Then press **Alt-F1** to return to the setup screen and press **ENTER** to finish setup.

3 Installing HPT370/372/372A Driver on an Existing System

If you are currently running Linux and would like to access drives or arrays attached to the HPT37x controller, you can perform the following steps.

Note

1. If you use a SCSI adapter to boot your system, you must make sure the HPT37x controller BIOS will be loaded after that adapter's BIOS. If not, try to move it to another PCI slot. Otherwise you may be unable to boot up your system.
2. A defect in default Red Hat Linux 7.1/7.2/7.3 kernel will hang your system during boot phase with HPT372 chip. So, if you want to use HPT372 chip with Red Hat Linux 7.1/7.2/7.3 system, you should boot from the provided boot floppy and update the kernel first.

Step 1 Obtain the Driver Module

You can extract the module file from the file modules.cgz on the driver disk. Using the following commands:

```
# mount /dev/fd0
# cd /tmp
# gzip -dc /mnt/floppy/modules.cgz | cpio -idumv
```

Driver modules for different kernel version will be extracted:

/tmp/2.2.16-22/hpt37x2.o	Red Hat Linux 7.0 driver
/tmp/2.4.2-2/hpt37x2.o	Red Hat Linux 7.1 driver
/tmp/2.4.7-10/hpt37x2.o	Red Hat Linux 7.2 driver
/tmp/2.4.18-3/hpt37x2.o	Red Hat Linux 7.3 driver

Step 2 Update the Kernel (Red Hat Linux 7.1/7.2/7.3 Only)

If you are about to use Red Hat Linux 7.1/7.2/7.3 system with HPT370 or HPT372 controller, you must update the kernel to remove HPT366 IDE support from it first. There are two ways to do this.

1. Building a new kernel from kernel source

To build a new kernel, you must have installed kernel source and related packages. These packages are:

kernel-headers
kernel-source
make
binutils
glibc-devel
ncurses
ncurses-devel
cpp
gcc

You can find these packages (.rpm file) under directory RedHat/RPMS/ on Red Hat

Linux CD. You can use command "**rpm –i**" to install an .rpm file.

After all required packages are installed, follow these steps to build a new kernel:

1) Use the following command to add a link to the source directory:

for RedHat 7.1

 # ln –s /usr/src/linux-2.4.2 /usr/src/linux

for RedHat 7.2

 # ln –s /usr/src/linux-2.4.7-10 /usr/src/linux

for RedHat 7.3

 # ln –s /usr/src/linux-2.4.18-3 /usr/src/linux

2) Remove the following 2 lines from /usr/src/linux/drivers/ide/ide-pci.c:

```
{DEVID_HPT34X, "HPT34X", PCI_HPT34X, NULL, ...
{DEVID_HPT366, "HPT366", PCI_HPT366, ATA66_HPT366, ...
```

3) If you are using RedHat 7.2/7.3 on Athlon type CPU, you need to check kernel type setting every time before rebuilding new kernel. Using the commands below:

 # cp /boot/kernel.h /boot/kernel.h.old
 # sed 's/i686/athlon/g' /boot/kernel.h.old > /boot/kernel.h

4) Type in the following commands to configure the kernel:

 # cd /usr/src/linux
 # make clean
 # make menuconfig

Now the kernel configuration dialog will appear. You can adjust the options based on your running system's hardware/software configuration. For most options you can simply use the default setting except the following:

a. Seselecting the option "Processor type and features" according to your system configuration.

b. Enable kernel versioning option by selecting "Loadable module support->Set version information on all module symbols".

c. Configure SCSI support and SCSI disk support as module by setting "SCSI support->SCSI support" and "SCSI support->SCSI disk support" as module.

d. Enable initial RAM disk support by selecting "Block devices->RAM disk support" and "Block devices->Initial RAM disk (initrd) support".

e. Enable loopback device support by selecting "Block devices->Loopback device support".

f. For convenience, you can enable the option "File systems ->DOS FAT fs support" to easily access DOS floppies.

When you finish, select "Exit" and save the kernel configuration when prompted.

5) Type in the following command:

 # make dep

Note

1.For RedHat 7.0/7.1, the file /usr/src/linux/include/linux/modversions.h may be incorrect after "make dep" in some Red Hat Linux distributions. Please check if there is a line "#include <linux/rhconfig.h>" before " #include <linux/modsetver.h>" in that file. If not, you should add the line manually.
2.For RedHat 7.2/7.3, you have to check kernel version after running "make dep". The UTS_RELEASE define line in /usr/src/linux/include/linux/version.h should be
for RedHat 7.2
#define UTS_RELEASE "2.4.7-10"
for RedHat 7.3
#define UTS_RELEASE "2.4.18-3"
If not, correct it using an editor (eg. vi)

6) Type in the following command to build the kernel image:

 # make bzImage

If the kernel is built successfully, you can copy the new kernel file /usr/src/linux/arch/i386/boot/bzImage to /boot directory.

 # cp /usr/src/linux/arch/i386/boot/bzImage /boot

For more information on how to build and install a new kernel from kernel source, please refer to Linux kernel documents. The Kernel-HOWTO document gives a detailed guide to kernel configuration, compilation, upgrades, and troubleshooting.

2. Using the kernel on the driver diskette

If you do not want to build the kernel by yourself, you can simply use the kernel "vmlinuz.hpt37x2" on the driver diskette we provided.

After you get the new kernel, you can install it using Grub or Lilo. Follow the steps below.

Using Grub

For RedHat 7.2/7.3, you can choose using Grub to boot system kernel.

1) Copy the new kernel image to the boot directory. Here we name it to /boot/vmlinuz.hpt37x2. You can change it to whatever name you want.

2) Modify /boot/grub/menu.lst settings to install the new kernel configuration. You can use the command "**vi /boot/grub/menu.lst**" to open menu.lst with the vi editor and modify it. Generally you can add the following lines to the file (in this example, we name the new kernel file as "vmlinuz.hpt37x2".):

title RedHat Linux (hpt37x2)	
root (hd0,0)	*Grub's root device*
kernel /vmlinuz.hpt37x2 ro root=/dev/hda5	*The new kernel image*
initrd /initrd-2.4.2-2.img	*RAM disk file, see below*

Note

a) Your grub's root device may be not on (hd0,0). Check the correct location and modify the line "root (hd0,0)" to match your system configuration.

b) Your system's root file system may be not on /dev/hda5. Check the correct location and modify the line " root=/dev/hda5" to match your system configuration. You can use the "mount" command to see on which device you boot file system is mounted.

c) If your system needs an initial RAM disk image to boot up (especially when you use a SCSI drive to boot), you should check existing sections in your menu.lst file for the "initrd" line and add it to the new section. The default RAM disk file name is initrd-2.4.2-2.img for Redhat 7.1, initrd-2.4.7-10.img for Redhat 7.2, initrd-2.4.18-3.img for Redhat 7.3.

Using Lilo

1) Copy the new kernel image to the boot directory. Here we name it to /boot/vmlinuz.hpt37x2. You can change it to whatever name you want.

2) Modify /etc/lilo.conf settings to install the new kernel. You can use the command "**vi /etc/lilo.conf**" to open lilo.conf with the vi editor and modify it. Generally you can add the following lines to the file (in this example, we name the new kernel file as "vmlinuz.hpt37x2".):

image=/boot/vmlinuz.hpt37x2	*The new kernel image*
label=linux.hpt37x2	
initrd=/boot/initrd-2.4.2-2.img	*RAM disk file, see below*
read-only	
root=/dev/hda5	*Root file system, see below*

Note

a) Your root file system may be not on /dev/hda5. Check the correct location and modify the line " root=/dev/hda5" to match your system configuration. You can use the "mount" command to see on which device you root file system is mounted.

b) If your system needs an initial RAM disk image to boot up (especially when you use a SCSI drive to boot), you should check existing sections in your lilo.conf file for the "initrd=" line and add it to the new section. The default RAM disk file name is initrd-2.4.2-2.img for Redhat 7.1, initrd-2.4.7-10.img

for Redhat 7.2.

3) Tell lilo to boot the new kernel by default by modifying the "**default=**" line to "**default=linux.hpt37x2**".

4) After you finish the modification, save the file and exit the editor, then use the command "**lilo**" to install the new kernel.

5) If "**lilo**" finishes without error, reboot from the new kernel to go to the next step, "**Test the Driver Module**".

Step 3 Test the Driver Module

You can test out the module to ensure that it works for your system by changing working directory to the location where hpt37x2.o resides and typing in the command "**insmod hpt37x2.o**".

Sometimes insmod will report "**unresolved symbols**" when you attempt to load the module. This can be caused by two ways:

1) If your system is using a kernel which has not built-in SCSI support, you must load the SCSI module before load hpt37x2.o. Try to load SCSI modules first.

E.g. # **insmod scsi_mod**
insmod sd_mod
insmod hpt37x2.o

2) If you recompile the kernel with SCSI support and still receive the "**unresolved symbols**" error, it may be caused that you have not configured symbol versioning correctly. To correct it, recompile the kernel with symbol versioning configured. Please refer to the kernel documents for more information.

To ensure the module has been loaded successfully, you can check the driver status by typing in the command "**cat /proc/scsi/hpt37x2/X**", where **X** is the filename you found under /proc/scsi/hpt37x2/. You should see the driver banner and a list of attached drives. You can now access the drives as a SCSI device (the first device is /dev/sda, then /dev/sdb, etc.).

Example

You have configured a RAID 0/1 array using 4 disks. It will be registered to system as device **/dev/sda**. You can use "**fdisk /dev/sda**" to create a partition on it, which will be **/dev/sda1**, and use "**mkfs /dev/sda1**" to setup a file system on the partition. Then you can mount **/dev/sda1** to somewhere to access it.

Step 4 Configure System to Automatically Load the Driver

Most likely, you will not want to type in "**insmod hpt37x2.o**" each time you boot up the system. Therefore you must install the module and tell the system about it. To install the module, type in the following commands (first change directory to where the proper

hpt37x2.o can be located):

On Red Hat Linux 7.0, use

```
# install -d /lib/modules/2.2.16-22/scsi
# install -c hpt37x2.o /lib/modules/2.2.16-22/scsi
```

On Red Hat Linux 7.1, use

```
# install -d /lib/modules/2.4.2-2/kernel/drivers/scsi
# install -c hpt37x2.o /lib/modules/2.4.2-2/kernel/drivers/scsi
```

On Red Hat Linux 7.2, use

```
# install -d /lib/modules/2.4.7-10/kernel/drivers/scsi
# install -c hpt37x2.o /lib/modules/2.4.7-10/kernel/drivers/scsi
```

On Red Hat Linux 7.3, use

```
# install -d /lib/modules/2.4.18-3/kernel/drivers/scsi
# install -c hpt37x2.o /lib/modules/2.4.18-3/kernel/drivers/scsi
```

Then you should inform the system when to load the module.

1. If you have no other SCSI adapters installed, you can edit the file "**/etc/modules.conf**" and add the following lines:

   ```
   probeall block-major-8 scsi_mod sd_mod hpt37x2
   options –k hpt37x2
   ```

   This tells the kernel to try loading the SCSI and hpt37x2 modules whenever it tries to access a SCSI device /dev/sd[a-z]. If you have SCSI support compiled in kernel, you may remove the "scsi_mod" and "sd_mod" from that line.

   The line “options –k hpt37x2” ensures the module will not be autocleaned. Otherwise the RAID management software will not run properly if system loads the driver with “autoclean” option.

   Notice

   Upon your system configuration the modules configuration file may be another file, possibly deprecated "conf.modules" file. You may have to check which configuration file you use and modify the correct one.

   Now, reboot the system and try to type in the command "fdisk /dev/sda". The kernel should automatically load the HPT37x2 driver.

2. If you use a SCSI adapter to boot the system, you cannot do as above since this may conflict with other SCSI devices. However, you can add the driver to the existing RAM disk image. First check which image file you are using by checking the “initrd=” line in file /etc/lilo.conf, then use the following commands (we assume the file is /boot/initrd-2.4.2-2.img. For Redhat 7.2 system, just need to substitute “initrd-2.4.2-2.img” with “initrd-2.4.7-10.img” to get the default RAM disk file

name):

 # gzip –dc /boot/initrd-2.4.2-2.img > /tmp/initrd.ext2
 # mkdir /mnt/initrd
 # mount –o loop /tmp/initrd.ext2 /mnt/initrd
 # cp hpt37x2.o /mnt/initrd/lib/ (specify the correct location of hpt37x2.o here)

Now, add a line "insmod /lib/hpt37x2.o" to the file /mnt/initrd/linuxrc, just below the line of insmoding SCSI adapter's kernel module. Example of linuxrc:

```
……
echo "Loading scsi_mod module"
insmod /lib/scsi_mod.o
echo "Loading sd_mod module"
insmod /lib/sd_mod.o
echo "Loading aic7xxx module"
insmod /lib/aic7xxx.o          ← SCSI adapter's kernel module
insmod /lib/hpt37x2.o          ← new inserted line
echo "Loading jbd module"
……
```

 # umount /mnt/initrd
 # gzip –c /tmp/initrd.ext2 > /boot/initrd-2.4.2-2.img

If you are using Lilo to boot your system, you also need to run lilo:

 # lilo

Then reboot your system and the driver will be loaded.

Step 5 Configure System to Mount Volumes when Startup

Now you can inform the system to automatically mount the array by modifying the file /etc/fstab. E.g. You can add the following line to tell the system to mount /dev/sda1 to location /mnt/raid after startup:

/dev/sda1 /mnt/raid ext2 defaults 0 0

4 Monitoring the Driver

Once the driver is running, you can monitor it through the Linux proc file system support. There is a special file under /proc/scsi/hpt37x2/. Through this file you can view driver status and send control commands to the driver.

Note

The file name is the SCSI host number allocated by OS. If you have no other SCSI cards installed, it will be 0. In the following sections, we will use X to represent this number.

Checking Devices Status

Using the following command to show driver status:

 # cat /proc/scsi/hpt37x2/X

This command will show the driver version number, physical device list and logical device list.

Rebuilding a Critical Array

A RAID 1 array or a RAID 0/1 array may become critical after a disk member fails. When an array is in critical status, it will loss the ability of fault tolerance until you finish rebuilding.

Generally rebuilding will automatically start if you have a spare disk for the RAID 1 array or you have replaced the failed disk. In these cases, the array needs to be synchronized to ensure data consistency. If the array is broken, you must first add a disk to the array. To add a disk to an array and start rebuilding, you can use the following command:

 # echo "hpt rebuild a b,c,d" > /proc/scsi/hpt37x2/X

In the command, "**a**" is array number shown in the logical device list. "**b**" is controller number (always 0 if you have one HPT37x controller installed), "**c**" is bus number (0 for primary channel, 1 for secondary channel), "**d**" is device number (0 for master device, 1 for slave device). E.g.

 # echo "hpt rebuild 1 0,1,0" > /proc/scsi/hpt37x2/X

will rebuild the array with logical device number 1 using the secondary master disk on the controller.

If rebuilding cannot be automatically started, you can use command

 # echo "hpt rebuild start" > /proc/scsi/hpt37x2/X

to start rebuilding. To stop the rebuilding process, use command

 # echo "hpt rebuild stop" > /proc/scsi/hpt37x2/X

Rescanning Devices

If you attach a disk after the system boots up, the driver will not detect the disk automatically. In this case, you can tell the driver to rescan the devices attached to it by typing in the following command:

 # echo "hpt rescan all" > /proc/scsi/hpt37x2/X

This command will rescan all devices and refresh their states. If you want to rescan only

a single device, you can use

 # echo "hpt rescan a,b,c" > /proc/scsi/hpt37x2/X

In the command, "**a,b,c**" specifies the controller, bus and device number for the disk. E.g. 0,1,0 specifies the secondary master disk on the first HPT37x controller.

Note

If the driver detects out a new disk plugged by rescanning the command and there is a broken RAID 1 array, the disk will be automatically used to rebuild the RAID 1 array.

5 Updating the Driver

If you are not booting from disks attached to HPT37x controller, you can update the driver just by reinstalling it following the previous section, "**Install HPT370/372/372A Driver on an Existing System**".

If you are using a system installed to HPT37x controller, you can update the driver by the following steps.

1) First obtain the new driver module file hpt37x2.o. Refer to the previous section "**Obtain the Driver Module**". In the following steps, we assume you have copied it to /tmp/hpt37x2.o.

2) Replace hpt37x2.o in the boot RAM disk image, /boot/initrd-xxx.img, where xxx is the kernel version. (2.2.16-22 for Red Hat Linux 7.0, 2.4.2-2 for Red Hat Linux 7.1, 2.4.7-10 for Red Hat Linux 7.2)

```
# gzip -dc /boot/initrd-xxx.img > /tmp/initrd.ext2
# mkdir /mnt/initrd
# mount -o loop /tmp/initrd.ext2 /mnt/initrd
# cp /tmp/hpt37x2.o /mnt/initrd/lib/hpt37x2.o
# umount /mnt/initrd
# gzip -c /tmp/initrd.ext2 > /boot/initrd-xxx.img
```

3) If you are using lilo to boot kernel, do remember using "**lilo**" to reinstall the RAM disk:

```
# lilo
```

4) Update hpt37x2.o in /lib/modules:

Red Hat Linux 7.0:

```
# cp /tmp/hpt37x2.o /lib/modules/2.2.16-22/scsi/hpt37x2.o
```

Red Hat Linux 7.1:

```
# cp /tmp/hpt37x2.o /lib/modules/2.4.2-2/kernel/drivers/scsi/hpt37x2.o
```

Red Hat Linux 7.2:

```
# cp /tmp/hpt37x2.o /lib/modules/2.4.7-10/kernel/drivers/scsi/hpt37x2.o
```

Red Hat Linux 7.3:

```
# cp /tmp/hpt37x2.o /lib/modules/2.4.18-3/kernel/drivers/scsi/hpt37x2.o
```

5) Reboot your system to make the new driver take effect.

6 Installing RAID Management Software

HighPoint RAID Management Software is used to configure and keep track of your hard disks and RAID arrays attached to HPT37x controller. Installation of the management software is optional but recommended.

Checking System Requirements

To run the RAID Management GUI, you must have the following software packages installed on your system:

1) X-Window system
2) gtk library v1.2 or later.

If you are using KDE or GNOME workstation, they are already installed. Otherwise you may check your system and refer to your Linux system manual for how to install these packages.

Preparing the Installation Files

You should have two files to finish the installation.

hptinstall.sh	Installation script file
hptraid.tar.gz	Package of software components

Installing the Software Package

Before installation, you must log on as root and change the directory to the location where your installation files are. Then you can use the command "**sh hptinstall.sh –i**" to install the software.

The following is an example.

```
[root@tmp]# ls
hptinstall.sh  hptraid.tar.gz
[root@tmp]# sh hptinstall.sh –i
Starting hptsvr daemon:                                    [ OK ]
HighPoint ATA RAID Management Software has been installed successfully!
[root@tmp]#
```

Note

If an old version is installed on your system you will be prompted to choose whether to overwrite existing files or not. To continue the installation, type in "**Y**".

Running the Management Software

You must log on as root to run the management software.

To run the software from a console window, you can just type in "**hptraid**" to start it. If you do not want to block the console, type in "**hptraid&** ".

If you are using GNOME or KDE, you can also run it from the menu bar:

On KDE, you can start it by choosing "**Menus (menu-bar)->Applications->HighPoint ATA RAID Management Software**".

On GNOME, you can start it by choosing "**KDE menus->GNOME->Applications-> HighPoint ATA RAID Management Software**".

7 Uninstalling

Uninstalling the Driver

You can only uninstall the driver when your system is not booting from devices attached to HPT37x controller. Just remove the lines you added to /etc/modules.conf and /etc/fstab.

Uninstalling the Management Software

Before you uninstall the software, you must log on as root. Then you can use the command "**hptinstall.sh –u**" to uninstall the software.

```
[root@tmp]# hptinstall.sh -u
Are you sure to uninstall HighPoint ATA RAID Management Software?(Y/N)y
Stopping hptsvr daemon:                                    [ OK ]
Uninstall finished!
[root@tmp]#
```