# The Linux Busmouse Howto

Mike Battersby, mike@starbug.apana.org.au

1.2, 1994/08/02

## Contents

# 1 Introduction.

This document is a guide to getting your busmouse working with Linux. I've written this in the hope that the ten people a week who post "how do I get my busmouse to work" questions in the comp.os.linux.* newsgroups will read it first, though I won't be holding my breath.

Busmouse support has been in the kernel for as long as I can remember, and hasn't changed in a long time, so this document should be relevant to any version of Linux you're likely to have.

## 1.1 Disclaimer.

The information in this document is correct to the best of my knowledge, but there's a always a chance I've made some mistakes, so don't follow everything too blindly, especially if it seems wrong. Nothing here should have a detrimental effect on your computer, but just in case I take no responsibility for any damages incurred from the use of the information contained herein.

`Microsoft(R) is a Trademark of Microsoft Corporation.`

[ trademark notices for other mice, anyone? — Mike ]

## 1.2 Feedback.

If you find any mistakes in this document, have any comments about its contents or an update or addition, send them to me at the address listed at the top of this howto.

## 1.3 Acknowledgements.

This howto has been, in the spirit of Linux, a community effort. Many thanks go to Johan Myreen for the sections on the PS/2 mice, Robert T. Harris for help on the ATI-XL sections and Reuben Sumner for miscellaneous info and constructive criticism.

Thanks also to the multitudes of people who have sent me mouse information, fixes or words of encouragement.

# 2 Determining your mouse type.

There are two separate but important characteristics you will need to know about your mouse before you go on: what interface it uses and what protocol it uses. The interface is the hardware aspect of the mouse, taking into account things like which i/o ports it uses and how to check if it is installed. This is the part which the kernel is concerned with, so that it knows how to read data from the mouse. The protocol is the software aspect of the mouse. Applications need to know the protocol to interpret the raw mouse data they receive from the kernel.

## 2.1 Mouse interfaces.

The Linux kernel currently supports four different kinds of bus mouse interface : Inport (Microsoft), Logitech, PS/2 and ATI-XL. The majority of bus mice have Inport interfaces, although lately PS/2 interfaces are becoming more popular. There is no surefire way of determining your mouse interface — mouse developers generally do their own thing when it comes to standards. The following sections may help, otherwise you'll just have to make it up.

## 2.2 Inport mice.

This includes most of the old style Microsoft mice which are shaped like a bar of dove soap. U.S. users who have purchased Gateway computers should note that the mice that come with them are not Inport mice but PS/2 mice (see below). Inport mice generally connect to an interface card which plugs into the bus on your motherboard. If the plug which connects your mouse cord to the interface card is round, has 9 pins, and a notch in one side you likely have an Inport mouse.

As far as I can tell, apart from the ATI-XL, all ATI mice (such as those on the Graphics Ultra cards) are plain Inport mice.

## 2.3 Logitech mice.

Logitech mice in general appear almost exactly the same as Inport mice. They too connect to an interface card via a 9 pin mini-din connector. Hopefully, it will have come in a Logitech box or have "Logitech" printed on the connector card so that you can tell it actually is a Logitech mouse.

There are also some truly ancient Microsoft mice (ones with ball bearings on the bottom as well as the mouse ball and a DB9 connector) which also use the Logitech protocol.

## 2.4 PS/2 mice.

PS/2 mice aren't really bus mice at all. The PS/2 mouse interface is not on an expansion card, the mouse is connected to the PS/2 Auxiliary Device port on the keyboard controller. A PS/2 mouse port uses a 6-pin mini DIN connector, similar to the keyboard connector. Many laptops also use this kind of interface to their trackballs — except for the connector, of course.

### 2.5 ATI-XL mice.

ATI-XL mice are a variant of Inport mice, with some slight differences. They come on the ATI-XL combined video adaptor/mouse card. Unless you know you have an ATI-XL card (and thus an ATI-XL mouse), you probably don't have one of these. It is possible for ATI-XL mice to use either the ATI-XL or Inport kernel drivers, although the ATI-XL driver should give better results.

### 2.6 Mouse protocols.

The PC world is full of different and conflicting mouse protocols. Fortunately, the choice for bus mice is considerable smaller than that for serial mice. Most Inport, Logitech and ATI-XL mice use the "BusMouse" protocol, although there are some ancient Logitech mice which use the "MouseSystems" protocol, and some even older Microsoft mice which use the Logitech protocol. PS/2 mice use the "PS/2" protocol.

## 3 Getting your mouse working.

Once you have figured out your mouse interface and protocol types, you're ready to proceed.

### 3.1 Setting the mouse interrupt.

Now, you'll need to know which interrupt number your mouse is using, and make sure it doesn't conflict with any other peripherals you have installed.

You should make sure that your mouse is not trying to use the same interrupt as any of your other devices — it is not possible for the mouse to share an interrupt under Linux, even though it may work fine under other operating systems. Check the documentation for all your peripherals to see which interrupt they use. In most cases IRQ4 is used for the first serial port (`/dev/ttyS0`), IRQ3 for the second (`/dev/ttyS1`) (these are assuming you actually have such devices — if you don't you can happily use their IRQ's), and IRQ5 for some SCSI adaptors.

Note that for ATI-XL, Inport and Logitech mice the kernel default is to use IRQ5, so if you are stuck with a pre-compiled kernel (eg, CD-ROM users) you will have to use that.

### 3.2 Inport and Logitech mice.

If you open up your computer's case and look at the card which your mouse plugs into, you should notice a block of jumpers on the card (hopefully labeled "INTERRUPT") with positions for interrupt (otherwise known as IRQ) numbers 2,3,4 and 5. To change the interrupt simply move the jumper from its current position onto the correct pair of pins.

```
**************************************************
***     MAKE SURE YOUR COMPUTER IS TURNED OFF   ***
***     BEFORE CHANGING THE JUMPERS AROUND.     ***
**************************************************
```

## 3.3 ATI-XL mice.

ATI-XL busmice have a software selectable IRQ - you should have received with your mouse a MS-DOS program (`VSETUP.EXE`) to set the IRQ. In order to do so you must (temporarily) boot MS-DOS and run this program. Note that the VSETUP program takes an optional parameter "/70" to increase the vertical refresh rate (which results in less flicker). The VSETUP program also allows you to select either the primary or secondary mouse address - you should set this to the primary address or the kernel will not be able to detect your mouse.

Once VSETUP has been run you must perform a hard reset for the new configuration to take effect.

## 3.4 PS/2 mice.

The PS/2 mouse always uses IRQ12 – there is no way of changing this (except with a soldering gun.) In the rare case that some other device is using IRQ12, you'll have to rejumper that peripheral to use another IRQ number.

## 3.5 Compiling the kernel.

In order for your busmouse to operate correctly you will need to recompile your kernel with the busmouse support compiled in.

Change to your kernel directory (here assumed to be (`/usr/src/linux`) and do a

```
make config
```

If you are unsure as to your mouse type, the first time you recompile the kernel you may wish to enable all of the busmouse options in the hope that the kernel will autodetect your mouse properly. People have mixed sucess with this: it doesn't always work, but on the other hand it might save you any further compiles.

### 3.5.1 Inport, Logitech and ATI-XL mice.

Answer "y" to the question pertaining to your type of busmouse interface and "n" to all the other busmouse questions. For example, if you have an Inport mouse you should answer "y" to

```
Microsoft busmouse support
```

and "n" to all other busmouse questions. Answer the non-mouse related questions as you usually would.

If you have a Logitech or Inport mouse, edit the file `/usr/src/linux/include/linux/busmouse.h` and change the line which says

```
#define MOUSE_IRQ 5
```

to reflect the interrupt number for your mouse (see section 3.1 for details on finding your interrupt number).

If you have an ATI-XL mouse, edit the file `/usr/src/linux/drivers/char/atixlmouse.c` and change the line which says

```
#define ATIXL_MOUSE_IRQ 5
```

to reflect your mouse's interrupt number.

Due to the vagaries of the PC architecture, if you have set your mouse to use interrupt 2, you must set the #define to use interrupt 9.

Examples

For a mouse on interrupt 3, you should change the line to read

```
#define MOUSE_IRQ 3
```

For a mouse on interrupt 2, you should change the line to read

```
#define MOUSE_IRQ 9
```

Next, compile your kernel as per the instructions which come with it, and boot from the new kernel. You should now have the busmouse support correctly compiled in.

### 3.5.2 PS/2 mice.

To compile the kernel with PS/2 mouse support answer "y" to the question.

```
PS/2 mouse (aka "auxiliary device") support
```

The PS/2 mouse driver actually supports two kinds of devices: the standard PS/2 Auxiliary Device controller and a special PS/2 mouse interface chip from Chips & Technologies which is used in the Texas Instruments Travelmate and Gateway Nomad laptops. To compile in support for the trackballs on these computers, answer "y" to the

```
C&T 82C710 mouse port support (as on TI Travelmate)
```

question. Note that you will still have to answer "y" to the question about the standard PS/2 driver to even get a chance to answer this question, since the 82C710 driver is actually an add-on to the standard PS/2 mouse driver.

When configured both for a standard PS/2 mouse device and the 82C710 device, the driver first tries to locate a 82C710 chip at boot time. Failing this, the standard driver is used instead, so using a kernel configured for both types of interface on a machine with a standard PS/2 mouse port should work too. However, there has been one report of a falsely detected 82C710 chip, so to be on the safe side do not configure in support for the 82C710 if you don't need it.

Compile your new kernel and boot from it as you normally would.

### 3.5.3 Selection.

Regardless of your mouse type you should answer "y" to the

```
Selection (cut and paste for virtual consoles)
```

question if you wish to run the selection program (see section 4.1 for more details).

## 3.6 The mouse devices.

Mice under Linux are accessed via the devices in the `/dev` directory. The following table gives a list of interface types and which device you should use.

| INTERFACE | DEVICE | MAJOR | MINOR |
|---|---|---|---|
| Logitech | /dev/logibm | 10 | 0 |
| PS/2 | /dev/psaux | 10 | 1 |
| Inport | /dev/inportbm | 10 | 2 |
| ATI-XL | /dev/atibm | 10 | 3 |

Table 1. Mouse devices.

**Note:**

If you are using your ATI-XL mouse with the Inport driver, you should use the `/dev/inportbm` device, not the `/dev/atibm` device.

The major and minor entries are the device numbers for that particular device.

If you find that you do not have these devices, you should create them first. To do so, execute the following as root.

```
mknod /dev/logibm   c 10 0
mknod /dev/psaux    c 10 1
mknod /dev/inportbm c 10 2
mknod /dev/atibm    c 10 3
```

**Note:**

Some time in the (progressively less) recent history of Linux the names for the busmouse devices have changed. The following device names have been superceded by those above and should be removed: `bmousems, bmouseps2, bmouseatixl, bmouselogitech`.

Many people like to create a symbolic link from their mouse device to `/dev/mouse` so that they don't have to remember which device they need to be using. If you have one of the current Linux distributions you will almost certainly find that you have such a link. If you have such a link, or create one, you should make sure that it is pointing to the correct device for your mouse.

### 3.7 Testing your mouse.

Get the file `selection-1.6.tar.gz` from your local Linux ftp site and compile the program "test-mouse" within it, according to the instructions (all you should need to do is type "make test-mouse"). Run the program like this

```
test-mouse -t <mouse arg> -m <mouse dev>
```

where <mouse arg> is "bm" (without the quotes) if you use the BusMouse protocol, or "ps2" if you use the PS/2 protocol, and <mouse dev> is your mouse device name from table 1.

If your mouse is working correctly, you should be able to paint on the screen by holding down the right or left mouse button and dragging the mouse. Push the left and right mouse buttons simultaneously to exit the program.

## 4 Using your mouse.

### 4.1 Selection.

Selection is a program which allows you to do mouse based 'cut- and-paste' between virtual consoles under Linux. Selection can be found as the file `selection-1.6.tar.gz` at your friendly Linux FTP site (such as `sunsite.unc.edu`), and contains instructions for getting it compiled. Some Linux distributions, such as Slackware, come with a precompiled selection binary.

When invoking selection, use the -t switch to selection to indicate which protocol your mouse is using and the -m option to indicate which mouse device you are using. The default is to use the `/dev/mouse` device, so you can omit the -m option if you have the appropriate symbolic link. For example, if you use the BusMouse protocol, selection should be run like this:

```
selection -t bm
```

or if you use the PS/2 protocol:

```
selection -t ps2
```

You should then be able to cut and paste text between virtual consoles using the mouse buttons. Read the documentation with selection, or do a "man selection" for more information on how to operate it.

There have been a couple of reports of selection not working correctly with certain laptop trackballs (under the PS/2 interface). Replacing the line

```
{ 0xcc,   0x00,   0x00,   0x00,   3      }   /* PS/2 */
```

in selection's mouse.c file with the line

```
{ 0xcc,   0x08,   0x00,   0x00,   3      }   /* PS/2 */
```

may help if you are having problems and can't track it down to anything else.

### 4.2 XFree86.

To use your busmouse under XFree86, you will need to set your mouse protocol type in your Xconfig file. If you have a BusMouse protocol mouse, your Xconfig should contain (including the quotes)

```
Busmouse        "/dev/mouse"
```

For PS/2 mice it should have

```
ps/2        "/dev/mouse"
```

If you have a two button mouse, it should also contain the line

```
Emulate3Buttons
```

which will allow you to emulate the use of the middle mouse button by pressing both mouse buttons simultaneously. All other mouse related lines, such as "BaudRate" and "SampleRate" should be commented out, as these have no effect on bus mice.

### 4.3 XFree86 and selection.

Unlike serial mice, you cannot share busmice between processes. This means you will have to kill any copies of "selection" (see section 4.1) you have running before you start up Xfree86. If you try to run X with selection running, you will get errors like the following

```
Fatal server error:
Cannot open mouse (Device or resource busy)
```

Version 1.6 of selection allows you to terminate running copies of selection by executing

```
selection -k
```

This should be done before starting up X11. You may wish to add a line containing the above command to the top of your `startx` script so that the mouse is shut down automatically. If you have an older version of selection which does not support the `-k` switch you will have to kill selection by hand.

## 5 Still can't get your mouse going?

So you've read through this howto a dozen times, done everything exactly as you think you should have, and your mouse still doesn't work? The best advice I can give you is this: experiment. Sure, it's a pain in the posterior, but in the end the only way to find out what is going to work with your mouse is to try all of the alternatives until you have success.

As always, if there is something you don't understand, try reading the manual page first and see if that helps. If you have a specific question, or a problem you think I might be able to help with, feel free to contact me at the address listed at the top of this howto, and I'll see if I can help you out or point you to someone who can.

The `comp.os.linux.help` newsgroup is the appropriate forum for discussion and/or questions regarding mice — please don't post questions to other groups, and especially don't crosspost questions to two or more of the Linux groups, they are more than cluttered enough as it is! When posting, you will get a much better response (and much fewer flames) if you use appropriate Subject: and Keywords: lines. For example:

```
Subject: BUSMICE - Gateway 2000 mouse wont work.
Keywords: mouse busmouse gateway
```