PENMOUNT DEVICE DRIVER USERS' GUIDE FOR LINUX

Version 2.0A

07/Feb/'17

PenMount®
TOUCH SCREEN SOLUTIONS

Preface

Disclaimer

The information in this document is subject to change without notice. The manufacturer makes no representations or warranties regarding the contents of this manual and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Furthermore, the manufacturer reserves the right to revise this publication or make changes in the specifications of the product described within it at any time without notice and without obligation to notify any person of such revision.

Trademarks

PenMount is a registered trademark of SALT International Corp. Microsoft and Windows are registered trademarks of Microsoft Corp. Other product names used in this manual are the properties of their respective owners and are acknowledged.

Copyright

Revision Table

Date	Revision	Changes
07/Jan/2014	1.0	First release
23/Jan/2014	1.1	Add Chapter 5, description of the SDK.
12/Jun/2014	1.2	Add section 3.8 that describes the penmount.dat file format.
16/Sep/2015	1.3	Add descriptions on new Xorg releases
01/Dec/2015	1.4	Textual refinement and editing
03/Jun/2016	1.5	(1) Revise 1.1 Supported Device (2) Add information about Xorg X Server 1.18
30/Nov/2016	2.0	Add: 2.5.4 Fedora 25 calibration information
07/Feb/2017	2.0A	PM1415, PM1715, and PM2204 are added to the supported device list.

Table of Content

1. Introduction

This document is the device driver users' guide for PenMount touch screen controllers. The "Device Driver" term in this document refers to the input modules for X Server on x86 platforms, and does cover the "Linux Kernel Driver". In addition, this document does not provide information on systems that do not use X Server.

1.1. Supported Device

The supported control boards and interfaces are listed in the table below.

Series	Product Name	USB	RS-232 / UART	I²C[1]
PenMount P2-02 Series	PM1100		v	
	PM2101		v	v
PenMount P2-03 Series	PM1200	v	v	
	PM1201	v	v	v
	PM2201	v	v	v
PenMount P2-04 Series	PM1300A	v		
	PM1302	v	v	v
	PM1400A	v	v	
	PM1401	v	v	
	PM1401A	v		v
	PM1500	v	v	v
PenMount P2-06 Series	PM1110A		v	
	PM1210	v	v	v
	PM2103	v	v	v
	PM2203 PM2203B	v	v	v
	PM2203C	v		
	PM2204	v	v	v
	PM2300	v	v	v
	PM2351	v		
PenMount P2-08 Series	PM1310	v	v	v
	PM1410	v	v	v
	PM1415	v		v
	PM1710	v	v	
	PM1711	v		v
	PM1715	v		v
PenMount 5000 Series	PM5126	v		v
	PM51A5	v		
PenMount 6000 Series	PM6200	v	v	
	PM6202	v	v	

[1] I²C interface currently requires using kernel driver or PMT101 bridge board to operate with X Window system.

	PM6300	v	
	PM6500	v	v
	PM6005	v	v
PenMount 9000 Series	PM9026		v
	PM9036		v

1.2. Supported Systems

The PenMount X Server Device Driver supports the following X Server versions.

Xorg Release	X Server Version	Distribution	Configuration	Multiple Touch Support
X11R6.8		RHEL4	xorg.conf	
X11R7.0	Xorg server 1.0	Ubuntu 6.06		
X11R7.1	Xorg server 1.1	RHEL5		
X11R7.2	Xorg server 1.2	openSUSE 10.2		
	Xorg server 1.3	Fedora 8		
X11R7.3	Xorg server 1.4	Debian 5 Ubuntu 8.04 LTS openSUSE 11.0	99-x11-penmount.fdi	
X11R7.4	Xorg server 1.5	Fedora 10 Ubuntu 8.10 openSUSE 11.1		
	Xorg server 1.6	Fedora 11 Ubuntu 9.04 Ubuntu 9.10 openSUSE 11.2		
X11R7.5	Xorg server 1.7	Debian 6 Ubuntu 10.04 LTS Fedora 12		
	Xorg server 1.8	Fedora 13 openSUSE 11.3	99-input-penmount.conf	
X11R7.6	Xorg server 1.9	Fedora 14 Ubuntu 10.10		
	Xorg server 1.10	Fedora 15 Ubuntu 11.04 openSUSE 12.1		
	Xorg server 1.11	Fedora 16		
X11R7.7	Xorg server 1.12	Fedora 17 openSUSE 12.2		○
	Xorg server 1.13	Fedora 18 Ubuntu 12.10 Ubuntu 13.04 openSUSE 12.3		○
	Xorg server 1.14	Fedora 19/20 Ubuntu 13.10 openSUSE 13.1		○
	Xorg server 1.15	Ubuntu 14.04 CentOS/RHEL 7		○
	Xorg server 1.16	Ubuntu 14.10 Fedora 21 openSUSE 13.2		○

	Xorg server 1.17	Ubuntu 15.04 Fedora 22		○
	Xorg server 1.18	Ubuntu 16.04 Ubuntu 16.10 Fedora 23 Fedora 24 openSUSE 42.2		○
	Xorg server 1.19	Fedora 25 Ubuntu 17.04		○

2. Installing Device Driver

This chapter provides information on installing the device drivers on various Linux platforms.

2.1. Finding a Proper Device Driver Package

There are many versions of PenMount Linux device drivers available for download on the PenMount website. Each of them are for a specific version of Linux distributions, such as Ubuntu 12.04 LTS, or Red Hat Enterprise Linux 6.

- For Listed Distributions

Please find and download and install the device driver according to the distribution used. For example, if you are using Ubuntu 12.04 LTS, please download the U12.04 32&64bit V4.0.4 device driver. Device drivers for other Ubuntu versions, such as 8.04, 10.04 will not work on 12.04.

- For Other Distributions

If the Linux distributions you use are not listed on the PenMount website, please follow the steps below to find a compatible driver.

1. Open the /var/log/Xorg.0.log file.
2. Find the version of Xorg X Server used in the file. For example, in Puppy Linux 5.4 uses Xorg XServer 1.12.3.
3. Browse the PenMount website, and find a device driver for a distribution that matches the same X Server minor version. For example, Fedora 17 uses Xorg Xserver 1.12.0, so its device driver should be compatible on Puppy Linux 5.4.

2.2. Using Installation Script

To install the PenMount device driver, please start the installation with the following command:

```
./install.sh
```

The install.sh file uses shell scripts to automate the installation process, including detecting device existence and copying device driver files and utilities to proper directories. The following are sample messages shown when installing on CENTOS 5.

```
[penmount@localhost pmLinux-RHEL5]$ ./install.sh
=======================================
      PenMount X Installation
=======================================
[Installer] Stopping PenMount Utilities ...
[sudo] password for penmount:
[Installer] Copying PenMount Utilities ...
[Installer] Copying PenMount Resource Files...
[Installer] Setting up System ...
[Installer] Setting up X Server ...
Setting up PenMount for X Server 7.1 (The X.Org Foundation)
+-----------------------------------------------+
|          PenMount Setup Utility               |
```

```
+-------------------------------------------------+
[PenMount] Version 4.1.0

Detect PenMount USB
-----------------------
PenMount 6000 (14E1:6000) @ /dev/input/event5

Detect PenMount COM
-----------------------
[Tip] If cannot detect PenMount RS-232 device, please uninstall old device driver
first !

Detecting /dev/ttyS0 ,  38400 bps ..... NOT FOUND
Detecting /dev/ttyS0 ,  19200 bps ..... FOUND
Detecting /dev/ttyS1 ,  38400 bps ..... NOT FOUND
Detecting /dev/ttyS1 ,  19200 bps ..... NOT FOUND
Detecting /dev/ttyS1 ,   9600 bps ..... NOT FOUND
Detecting /dev/ttyS2 ,  38400 bps ..... FAIL
Detecting /dev/ttyS2 ,  19200 bps ..... FAIL
Detecting /dev/ttyS2 ,   9600 bps ..... FAIL
Detecting /dev/ttyS3 ,  38400 bps ..... FAIL
Detecting /dev/ttyS3 ,  19200 bps ..... FAIL
Detecting /dev/ttyS3 ,   9600 bps ..... FAIL
Detecting /dev/ttyS4 ,  38400 bps ..... FAIL
Detecting /dev/ttyS4 ,  19200 bps ..... FAIL
Detecting /dev/ttyS4 ,   9600 bps ..... FAIL
Detecting /dev/ttyS5 ,  38400 bps ..... FAIL
Detecting /dev/ttyS5 ,  19200 bps ..... FAIL
Detecting /dev/ttyS5 ,   9600 bps ..... FAIL
Detecting /dev/ttyS6 ,  38400 bps ..... FAIL
Detecting /dev/ttyS6 ,  19200 bps ..... FAIL
Detecting /dev/ttyS6 ,   9600 bps ..... FAIL
Detecting /dev/ttyS7 ,  38400 bps ..... FAIL
Detecting /dev/ttyS7 ,  19200 bps ..... FAIL
Detecting /dev/ttyS7 ,   9600 bps ..... FAIL
Detecting /dev/ttyUSB0 ,  38400 bps ..... FAIL
Detecting /dev/ttyUSB0 ,  19200 bps ..... FAIL
Detecting /dev/ttyUSB0 ,   9600 bps ..... FAIL
Detecting /dev/ttyUSB1 ,  38400 bps ..... FAIL
Detecting /dev/ttyUSB1 ,  19200 bps ..... FAIL
Detecting /dev/ttyUSB1 ,   9600 bps ..... FAIL

Detection Result
-----------------------
PenMount 6000 (14e1:6000) @ /dev/input/event5
PenMount 6000 Serial @ /dev/ttyS0, 19200 bps

+----------------------------------+
|        Select PenMount           |
+----------------------------------+
      1 : PenMount 6000 @ /dev/input/event5
      2 : PenMount 6000 @ /dev/ttyS0, 19200 bps
```

```
Which one? (1-2) => 1
Setting up xorg.conf...
Writing configurations to /etc/penmount/penmount.dat
[Installer] Copying PenMount X Input module file :  0.6
========================================
   PenMount Installation Finished !
========================================
[Installer] Please restart the system to make changes take effect !
```

The installation script will need administrative privilege to copy files to system directories, so it will prompt for a password when starting.

```
========================================
      PenMount X Installation
========================================
[Installer] Stopping PenMount Utilities ...
[sudo] password for penmount:
```

The following messages indicate that the PenMount 6000 USB is found and the device name is /dev/input/event5.

```
Detect PenMount USB
------------------------
PenMount 6000 (14E1:6000) @ /dev/input/event5
```

The following messages indicate that the PenMount device is found using 19200 baud rate on /dev/ttyS0, which is COM1. When using the installation script, it scans from COM1 to COM8, and also checks USB and RS-232 devices if available.

```
Detect PenMount COM
------------------------
[Tip] If cannot detect PenMount RS-232 device, please uninstall old device driver
first !

Detecting /dev/ttyS0 ,  38400 bps ..... NOT FOUND
Detecting /dev/ttyS0 ,  19200 bps ..... FOUND
Detecting /dev/ttyS1 ,  38400 bps ..... NOT FOUND
Detecting /dev/ttyS1 ,  19200 bps ..... NOT FOUND
Detecting /dev/ttyS1 ,   9600 bps ..... NOT FOUND
Detecting /dev/ttyS2 ,  38400 bps ..... FAIL
Detecting /dev/ttyS2 ,  19200 bps ..... FAIL
Detecting /dev/ttyS2 ,   9600 bps ..... FAIL
```

```
Detecting /dev/ttyS3 ,  38400 bps ..... FAIL
Detecting /dev/ttyS3 ,  19200 bps ..... FAIL
Detecting /dev/ttyS3 ,   9600 bps ..... FAIL
Detecting /dev/ttyS4 ,  38400 bps ..... FAIL
Detecting /dev/ttyS4 ,  19200 bps ..... FAIL
Detecting /dev/ttyS4 ,   9600 bps ..... FAIL
Detecting /dev/ttyS5 ,  38400 bps ..... FAIL
Detecting /dev/ttyS5 ,  19200 bps ..... FAIL
Detecting /dev/ttyS5 ,   9600 bps ..... FAIL
Detecting /dev/ttyS6 ,  38400 bps ..... FAIL
Detecting /dev/ttyS6 ,  19200 bps ..... FAIL
Detecting /dev/ttyS6 ,   9600 bps ..... FAIL
Detecting /dev/ttyS7 ,  38400 bps ..... FAIL
Detecting /dev/ttyS7 ,  19200 bps ..... FAIL
Detecting /dev/ttyS7 ,   9600 bps ..... FAIL
Detecting /dev/ttyUSB0 ,  38400 bps ..... FAIL
Detecting /dev/ttyUSB0 ,  19200 bps ..... FAIL
Detecting /dev/ttyUSB0 ,   9600 bps ..... FAIL
Detecting /dev/ttyUSB1 ,  38400 bps ..... FAIL
Detecting /dev/ttyUSB1 ,  19200 bps ..... FAIL
Detecting /dev/ttyUSB1 ,   9600 bps ..... FAIL
```

Please note that the installation script supports only one device, so if more than one device is attached to host PC, it will prompt for selecting one of them for installation.

```
Detection Result
-----------------------
PenMount 6000 (14e1:6000) @ /dev/input/event5
PenMount 6000 Serial @ /dev/ttyS0, 19200 bps

+-----------------------------------+
|         Select PenMount           |
+-----------------------------------+
      1 : PenMount 6000 @ /dev/input/event5
      2 : PenMount 6000 @ /dev/ttyS0, 19200 bps

Which one? (1-2) => 1
```

After installation finishes, please restart the system for changes to take effect.

2.3. Manual Installation

The installation script introduced in the previous section actually runs the “pm-setup” utility for detecting and setting up the device. The pm-setup utility supports manual installation mode, by adding the “–m” parameter.

```
pm-setup –m
```

The pm-setup utility will then interact with users to collect device settings.

1. PenMount Model

```
+-------------------------------------+
|           PenMount Model            |
+-------------------------------------+
    1. PenMount 6000 series (USB)
    2. PenMount 5000 series (USB)
    3. PenMount 6000 series (Serial)
    4. PenMount 9000 series (Serial)
    5. PenMount PCI series (USB)
    6. PenMount PCI series (Serial)

Which one? (1-6) => 3

*****  PenMount 6000 series (Serial)  *****
```

2. A/D Bit, which is only used when target model is PenMount 6000 device.

```
+--------------------+
|       A/D Bit      |
+--------------------+
    1. A/D 10-bit
    2. A/D 12-bit

Which one? (1-2) => 1

*****  10-bit  *****
```

3. If choosing Serial devices in step 1, pm-setup will ask for the attached serial port.

```
+----------------------------------------+
|               Serial Port              |
+----------------------------------------+
    1. COM1, Serial Port 1 (/dev/ttyS0)
    2. COM2, Serial Port 2 (/dev/ttyS1)
    3. COM3, Serial Port 3 (/dev/ttyS2)
    4. COM4, Serial Port 4 (/dev/ttyS3)
    5. Other Serial Port

Which one? (1-5) => 1

*****  /dev/ttyS0  *****
```

4. If choosing Serial devices in step 1, pm-setup will ask for the baud rate used.

```
+-------------------+
|      Baudrate     |
+-------------------+
    1. 19200 bps
    2.  9600 bps

Which one? (1-2) => 1

*****  19200 bps  *****
```

5. Settings for the Press and Hold gesture.

```
+-------------------------------+
|     Press and Hold Button     |
+-------------------------------+
    1. Right button (default)
    2. Middle button
    3. Left button
Which one? (1-3) => 1

*****  button 3  *****
```

6. The click mode setting. If you do not know which to choose, please press 1.

```
+------------------------------+
|          Click Mode          |
+------------------------------+
    1. Mouse Emulation (default)
    2. Pen Input Emulation
    3. Click On Release
    4. Click On Touch
Which one? (1-4) => 1

*****  Mouse Emulation  *****
```

7. The time interval for triggering the “Press and Hold” gesture.

```
+------------------------------+
|           Hold Time          |
+------------------------------+
How long to launch the press and
   hold button? (ms) => 700
```

8. The timing for generating buzzer beep.

```
+------------------------------+
|      Beep Configuration      |
+------------------------------+
    1. Beep when pen down (default)
    2. Beep when pen up
    3. Beep when pen down and pen up
    4. Beep off
Which one? (1-4) => 1
```

9. Device setup finishes with summary.

```
+------------------------------+
|            Finish            |
+------------------------------+
    Model : PenMount 6000 COM
Click Mode: Mouse Emulation
    Beep  : ON (pen-down)
Hold Time : 700 ms
```

2.4. Configuring Device Driver before Installation

During installation, the process will load the settings from the penmount.ini file in the device driver package. Adjusting these settings can be helpful for a better user touch experience.

Please note that most of these settings are adjustable with the PenMount Utility after the device driver is installed.

Option	Description	Default Value
ConfigFile	The file path of the configuration file, which stores the settings for PenMount touch calibration.	/etc/penmount/penmount.dat
ADBit	The A/D bit supported by the PenMount controller. This setting is only used if using PenMount 6000 RS-232.	10: Using 10 bit ADC 12: Using 12 bit ADC
DebugLevel	The debug message level. The higher the value, the more debug messages are logged to the /var/log/Xorg.0.log file.	0: Minimum message logging 3: Maximum message logging
TouchMode	The default operation mode.	MouseEmulation: Using mouse emulating mode ClickOnTouch: Using click on touch mode ClickOnRelease: Using click on release mode PenInputEmulation: Using touch mode. The device driver will generate touch events in X Server 1.12 and later systems.
HoldSwitch	Enable or disable the press and hold gesture.	1: press and hold gesture is enabled. 0: press and hold gesture is disabled.
PressAndHoldButton	The default mouse button event that a press and hold gesture will generate.	RightButton: Right mouse button will be sent. MiddleButton: Middle mouse button will be sent. LeftButton: Left mouse button will be sent.
HoldTime	The time duration for press and hold. The mouse button event will not be generated if pen up occurs within the time duration. This value is in milliseconds.	800: Press and hold for 800 milliseconds will let device driver sent the predefined mouse button event.
HoldArea	The maximum offset allowed for a press and hold gesture. If movement is larger than this amount, the press and hold gesture will not be generated.	32
XMinOffset **XMaxOffset** **YMinOffset** **YMaxOffset**	The edge compensation settings for four touch screen edges. The larger the value, the easier touch can reach the screen edge. The valid value range is from 0 to 30.	10
BeepType	The timing for a buzzer beep.	1

Monitor	The monitor name that the touch screen is mapped to. This value must be set correctly when multiple display monitors are used. This setting is not used when only one display monitor is connected.	default: Touch screen map to a monitor called “default”.
JitterFilter	The value range to filter out jitter. The valid value range is from 0 to 64. Please notice that using a large value might decrease the smoothness of line drawing.	8

2.5. Configuring Device Driver after Installation

After the device driver is installed, settings can be changed in the configuration file or by using the PenMount Utility. However, the modification may be different in different Linux distributions with different versions of X Server and window manager used. For this reason, we only chose some of the most common Linux distributions to describe how to change the device driver settings.

2.5.1. CENTOS 5

The CENTOS 5 is a typical system that uses the traditional xorg.conf file for configuration. There will be a new “InputDevice” section created after device driver is installed.

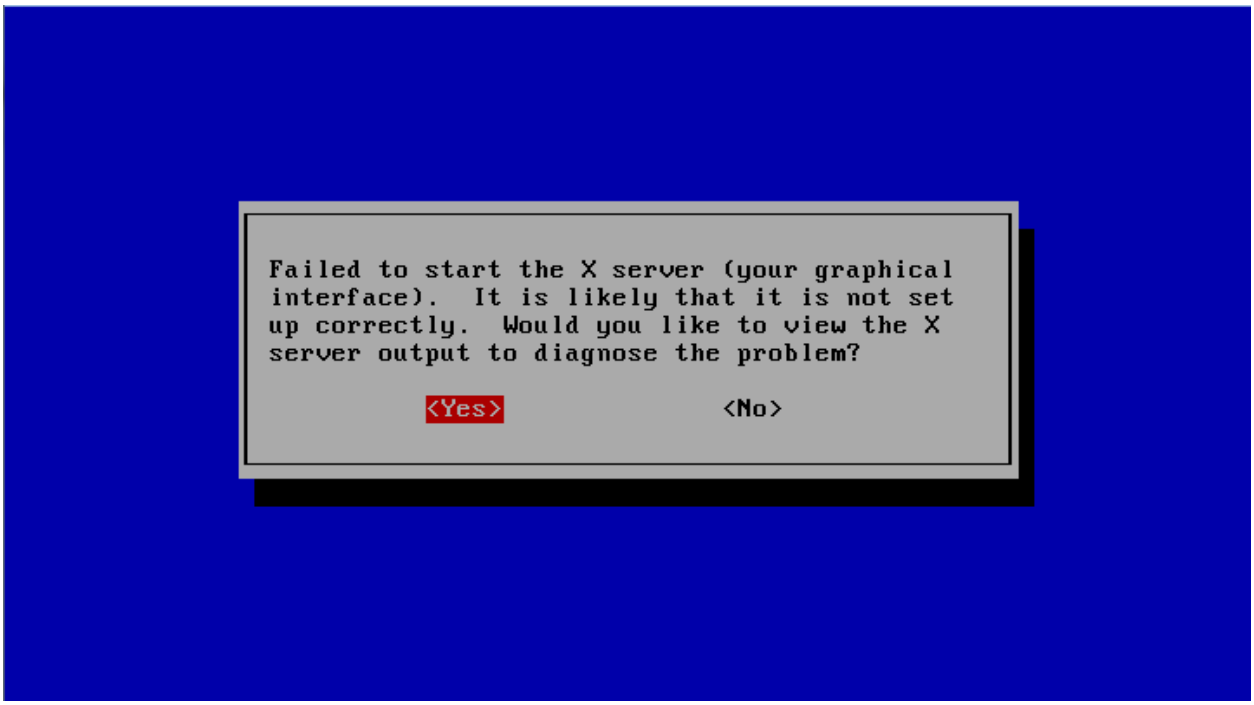
```
Section "ServerLayout"
        # Other ServerLayout settings.
        InputDevice  "PenMount" "SendCoreEvents"
EndSection

Section "InputDevice"
        Identifier  "PenMount"
        Driver  "penmount"
        Option  "Device"        "/dev/input/event5"
        Option  "Protocol"      "PM6000USB"
        Option  "BaudRate"      "19200"
        Option  "ADBit"         "10"
        Option  "ConfigFile"    "/etc/penmount/penmount.dat"
        Option  "RandR"         "0"
        Option  "Monitor"       "default"
        Option  "JitterFilter"  "8"
        Option  "DebugLevel"    "0"
EndSection
```

The device settings that can be changed in the xorg.conf file are listed in the table below:

Option	Description	Default Value
Protocol	The protocol used by the connected PenMount device. This setting must be configured correctly when using RS-232 interface.	PM6000USB: PenMount USB PM6000COM: PenMount 6000 RS-232 PM9000: PenMount 9000 RS-232 PM3000COM: PenMount P2 RS-232
BaudRate	The baud rate used by the PenMount RS-232	9600: 9600 bps

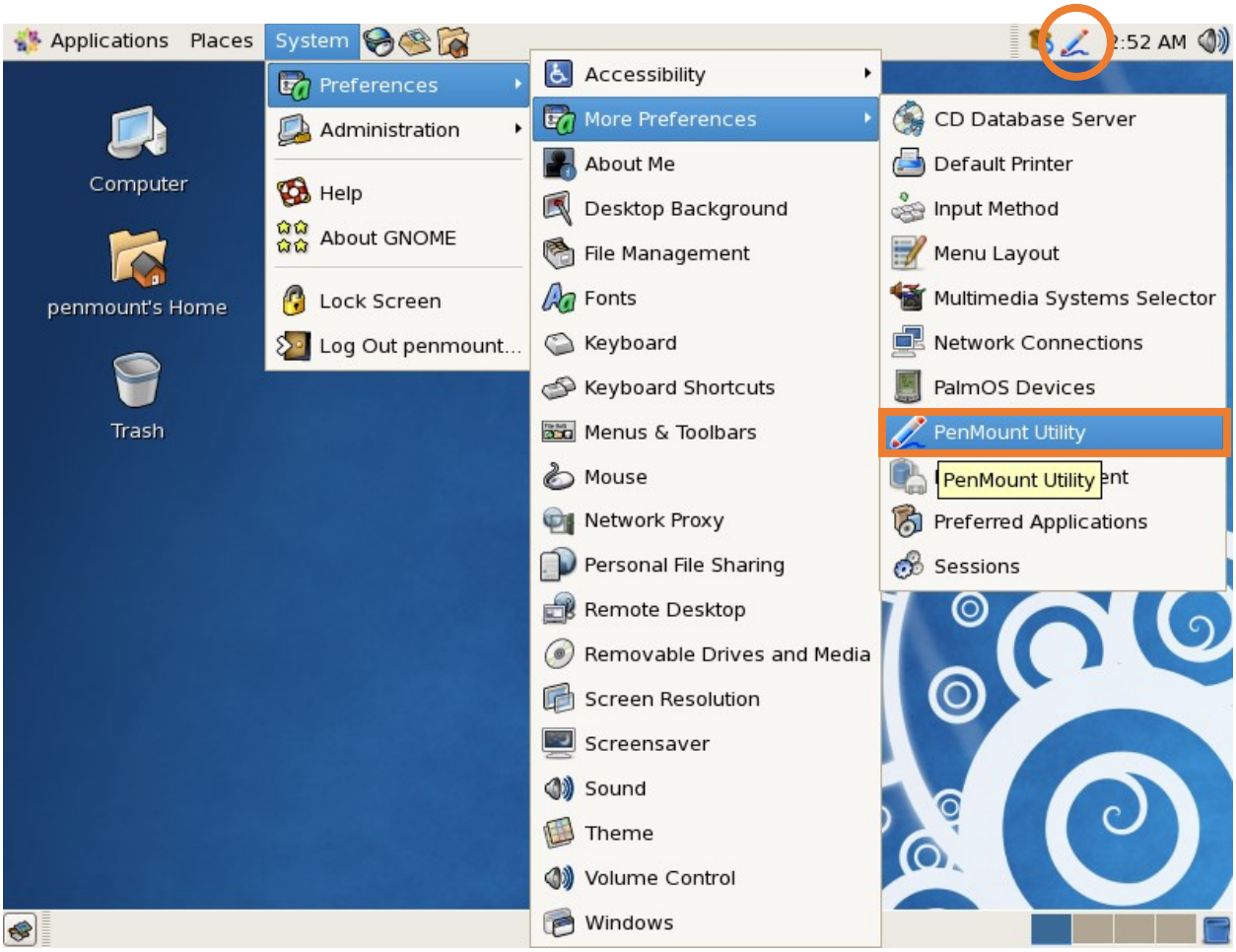
	device.	19200: 19200 bps 38400: 38400 bps
ADBit	The A/D bit supported by the PenMount controller. This setting is only used if using PenMount 6000 RS-232.	10: Using 10 bit ADC 12: Using 12 bit ADC
ConfigFile	The file path of the configuration file, which stores the settings for PenMount touch calibration.	/etc/penmount/penmount.dat
Monitor	The monitor name that the touch screen is mapped to. This value must be set correctly when multiple display monitors are used. This setting is not used when only one display monitor is connected.	default: Touch screen map to a monitor called “default”.
JitterFilter	The value range to filter out jitter. The valid value range is from 0 to 64. Please notice that using a large value might decrease the smoothness of line drawing.	8
DebugLevel	The debug message level. The higher the value is, the more debug messages are logged to the /var/log/Xorg.0.log file.	0: Minimum message logging 3: Maximum message logging

Please modify the settings with care, incorrect settings may prevent the X Server from starting up correctly.



The above screen may also appear when the PenMount USB device is not present when system starts up. In general, most systems that uses xorg.conf do not support USB hot plugging.

For other settings such as press and hold, please use “PenMount Utility”, which can be launched by click on the PenMount icon in system tray, or in “System” -> “Preferences” -> “More Preferences” -> ”PenMount Utility”. Please note that this launch path might be different in other Linux distributions. The usages of the PenMount Utility are described in later sections of this document.



2.5.2. CENTOS 6

For CENTOS 6, PenMount USB devices can be configured with the following file:

```
/etc/X11/xorg.conf.d/99-input-penmount.conf
```

This file contains an “InputClass” section for PenMount, please modify the settings with care.

```
Section "InputClass"
  Identifier     "Penmount Touch Screen"
  MatchProduct      "PenMount"
  MatchDevicePath   "/dev/input/event*"
  Driver         "penmount"
  Option         "Protocol"        "PM6000USB"
```

```
  Option        "BaudRate"        "19200"
  Option        "ADBit"          "10"
  Option        "ConfigFile"      "/etc/penmount/penmount.dat"
  Option        "RandR"          "0"
  Option        "Monitor"         "default"
  Option        "JitterFilter"    "8"
  Option        "DebugLevel"      "0"
EndSection
```

For PenMount RS-232 devices, they can be configured with the following file:

```
/usr/share/hal/fdi/policy/20thirdparty/99-x11-penmount.fdi
```

This file is written in XML that contains device info settings. Please modify the settings with care.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<deviceinfo version="0.2">
  <device>
    <match key="info.capabilities" contains="serial">
      <match key="linux.device_file" contains="/dev/ttyS0">
        <merge  key="input.x11_driver"            type="string">penmount</merge>
        <append key="info.capabilities"           type="strlist">input</append>
        <append key="info.capabilities"           type="strlist">input.mouse</append>
        <merge key="input.device"                 type="string">/dev/ttyS0</merge>
        <merge key="input.x11_options.Protocol"     type="string">PM6000COM</merge>
        <merge key="input.x11_options.Monitor"      type="string"></merge>
        <merge key="input.x11_options.BaudRate"     type="string">19200</merge>
        <merge key="input.x11_options.ADBit"        type="string">10</merge>
        <merge key="input.x11_options.ConfigFile"  type="string">/etc/penmount/penmount.dat</merge>
        <merge key="input.x11_options.JitterFilter" type="string">0</merge>
        <merge key="input.x11_options.DebugLevel"   type="string">0</merge>
      </match>
    </match>
  </device>
</deviceinfo>
```

2.5.3. Ubuntu 12.04 LTS

Ubuntu 12.04 LTS uses the following file for configuring the PenMount device:

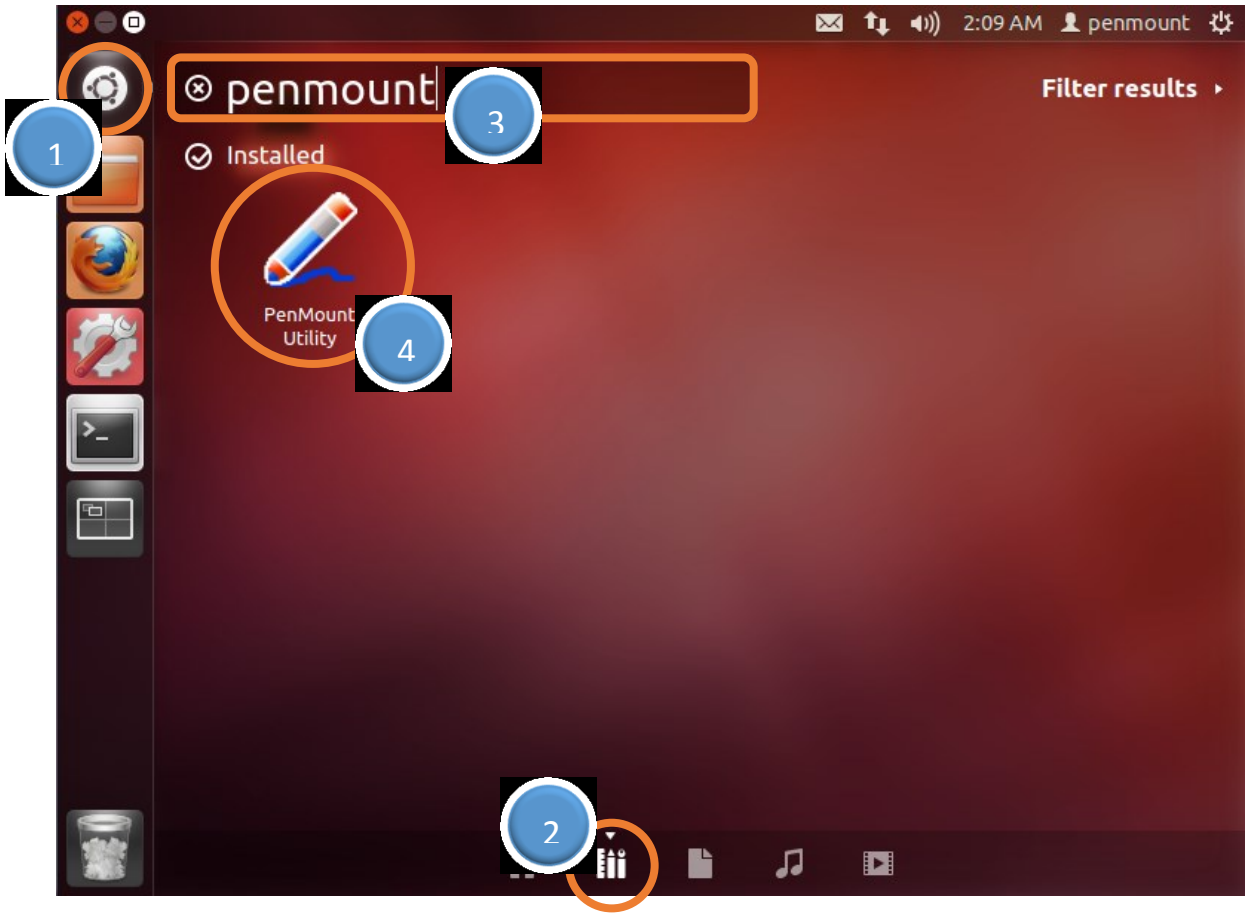
```
/usr/share/X11/xorg.conf.d/99-input-penmount.conf
```

This file contains an “InputClass” section for PenMount, please modify the settings with care.

```
Section "InputClass"
  Identifier     "Penmount Touch Screen"
  MatchProduct      "PenMount"
  MatchDevicePath   "/dev/input/event*"
  Driver         "penmount"
  Option         "Protocol"       "PM6000USB"
  Option         "BaudRate"        "19200"
  Option         "ADBit"          "10"
  Option         "ConfigFile"      "/etc/penmount/penmount.dat"
  Option         "RandR"          "0"
  Option         "Monitor"         "default"
  Option         "JitterFilter"    "8"
  Option         "DebugLevel"      "0"
EndSection
```

For PenMount Utility, please follow the steps below to launch this tool.

1. Click on the “Dash Home” button.
2. Click on the “Applications” button.
3. Type in “penmount” in the Search bar.
4. Click on the “PenMount Utility” icon.



2.5.4. Fedora 25

In Fedora 25, the default session is changed from Xorg to GNOME. To use the functionalities provided by PenMount, please consider changing session types during user login.

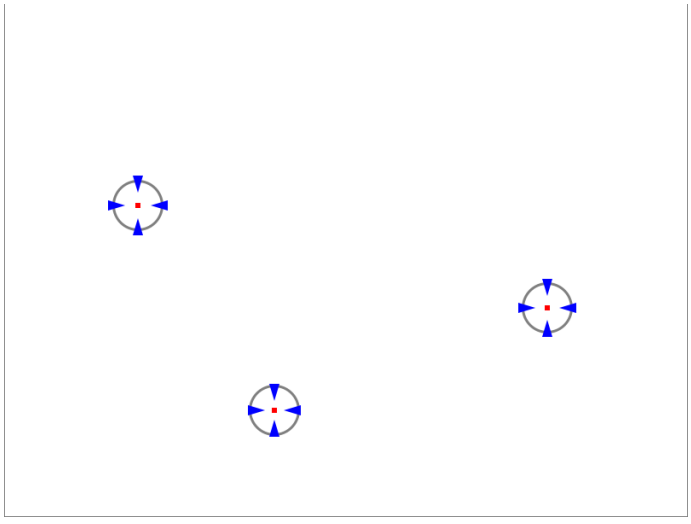
Session	Using Xorg X Server	Touchscreen Calibration
GNOME		3 Point Mode
GNOME Classic	○	PenMount 4/9/16/25 Mode
GNOME on Xorg	○	PenMount 4/9/16/25 Mode

- *GNOME Session*

For GNOME session, resistive touchscreen calibration is not supported by default, and PenMount provides limited calibration support by using customized system libraries.

Choosing the “PenMount Utility” icon or running the “pm-wcalib” utility launches the 3 point mode calibration interface.

Three points will appear on screen sequentially, please use touchscreen to press on the points one by one. Please avoid mouse clicks during calibration, since this utility responses to mouse button events, the calculated calibration parameters will be incorrect.



Please notice that the 3 point mode calibration night not be as accurate as what PenMount calibration provides, if accuracy is required, please consider change to “GNOME on Xorg” session.

- *GNOME Classic / on Xorg Session*

For GNOME Classic or Xorg Sessions, PenMount device driver and utility are supported. Please reference Chapter 3 for more information.

2.6. Troubleshooting

This sections provide some instructions on common device driver installation issues.

2.6.1. Cannot Detect Device

Here is a quick check list for troubleshooting this issue.

1. Check if PenMount device is powered and connected to the host computer.
   This could be easily verified by testing with a Microsoft Windows system on the same hardware platform.
2. Check if the USB or UART kernel drivers are working.
   This could be checked by running “dmesg”, for USB devices, there should be messages similar to the following:

```
usb 2-2.1: new full speed USB device number 4 using uhci_hcd
usb 2-2.1: New USB device found, idVendor=14e1, idProduct=6000
usb 2-2.1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
usb 2-2.1: Product: PenMount USB
usb 2-2.1: Manufacturer: DIALOGUE INC
usb 2-2.1: configuration #1 chosen from 1 choice
dracut: Switching root
input: DIALOGUE INC PenMount USB as
/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-2/2-2.1/2-2.1:1.0/input/input5
generic-usb 0003:14E1:6000.0002: input,hidraw1: USB HID v1.01 Mouse [DIALOGUE INC PenMount
USB] on usb-0000:02:00.0-2.1/input0
```

   For COM Ports, there should be messages similar to the following:

```
Serial: 8250/16550 driver, 4 ports, IRQ sharing enabled
serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
serial8250: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
00:0a: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
00:0b: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
```

3. For PenMount RS-232 devices, please also make sure that the IRQ and I/O addresses are configured correctly.
   This issue usually happens when the system has more than 2 COM Ports, and IRQ sharing is not working correctly. Please note that having a touchscreen working in Microsoft Windows does not guarantee that it will also work in Linux, since it is the operating system that controls IRQ sharing.

In some cases this issue can be solved by assigning IRQ=0 with the “setserial” utility. For example, to set up /dev/ttyS5, please use the command below:

```
setserial /dev/ttyS5 irq 0
```

4. If the PenMount device still cannot be automatically detected, you may try to use the manual installation method introduced in section 2.3.

2.6.2. Taking a long time to Scan COM Ports when installing USB

The pm-setup utility supports a special parameter “-d0” that skips device detection on COM ports.

```
pm-setup –s –d0
```

2.6.3. Touch not working after Installing Device Driver

There may be several reasons that cause this issue. The most common issue is that the installation was not performed with root permission, which causes the Xorg server to be unable to use the correct PenMount device driver.

To check whether the Xorg is using the correct device driver, please check the log messages in the following location:

```
/var/log/Xorg.0.log
```

For Fedora 22 and later systems, the log file is changed to the following directory.

```
$HOME/.local/share/xorg/Xorg.0.log
```

For example, if a PenMount device driver is used, there should be messages similar as below:

```
(II) config/hal: Adding input device DIALOGUE INC PenMount USB
(**) DIALOGUE INC PenMount USB: Applying InputClass "Penmount Touch
Screen"
(II) LoadModule: "penmount"
(II) Loading /usr/lib/xorg/modules/input/penmount_drv.so
(II) Module penmount: vendor="PenMount Touch Solutions"
      compiled for 1.13.0, module version = 1.2.1
      Module class: X.Org XInput Driver
      ABI class: X.Org XInput driver, version 18.1
```

```
(II) Using input driver 'penmount' for 'DIALOGUE INC PenMount USB'
(**) DIALOGUE INC PenMount USB: always reports core events
(**) Option "RandR" "0"
(**) Option "ADBit" "10"
(**) Option "JitterFilter" "8"
(**) Option "DebugLevel" "0"
(**) Option "Monitor" "default"
(**) Option "ConfigFile" "/etc/penmount/penmount.dat"
(II) [PENMOUNT] Default device : /dev/input/event5 !
(II) [PENMOUNT] Using Protocol : PM6000USB
(II) [PENMOUNT] Using device : /dev/input/event5
(**) Option "config_info"
"hal:/org/freedesktop/Hal/devices/usb_device_14e1_6000_noserial_if0_logicaldev_input"
(II) XINPUT: Adding extended input device "DIALOGUE INC PenMount USB"
(type: TOUCHSCREEN, id 10)
```

For PenMount USB devices, if there is a message similar to the below in the log file, it also indicates that the device driver installation was not successful. In this case, the default "evdev" driver is used instead.

```
(II) Using input driver 'evdev' for 'DIALOGUE INC PenMount USB'
```

2.7. Uninstalling Device Driver

To uninstall the PenMount device driver, please remove the device configurations introduced in section 2.4 which can be one of the following depending on the version of Xorg server used.

/etc/X11/xorg.conf
/usr/share/hal/fdi/policy/20thirdparty/99-x11-penmount.fdi
/usr/share/X11/xorg.conf.d/99-input-penmount.conf
/etc/X11/xorg.conf.d/99-input-penmount.conf

The PenMount device driver binary should also be removed which can be found in the following location:

/usr/lib/xorg/modules/input/penmount_drv.so

3. Using Software

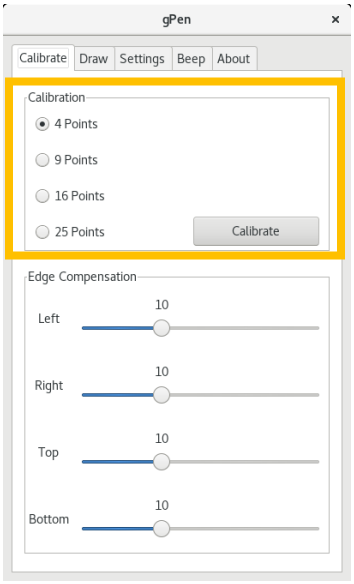
The “PenMount Utility” is a program that allows users to dynamically change device driver settings. Users can also launch this program in the terminal with the following command:

```
gPen
```

Please note that this utility requires GTK+ 2.0 library. If running the program without root permission, the setting changes may not be able to take effect.

3.1. Touch Calibration

The PenMount Utility provides a graphical interface for launching different touch calibration with different modes.

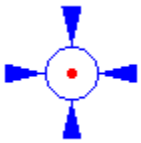


Users may also launch touch calibration in the terminal, with the following command:

```
gCal <mode>
```

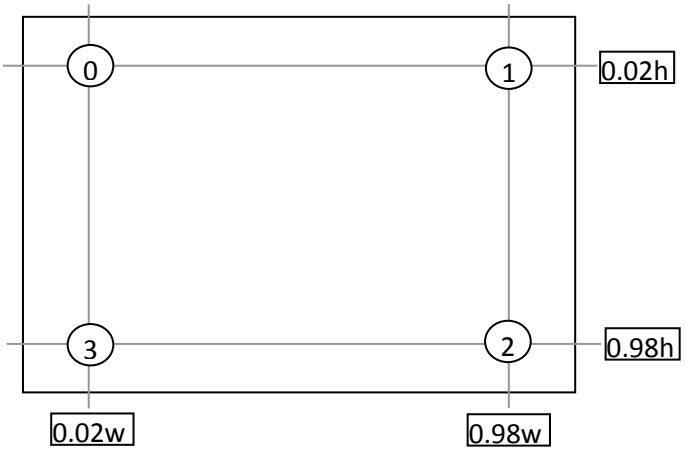
Please note that this utility requires GTK+ 2.0 library. If running the program without root permission, the setting changes may not be able to take effect.

There are four calibration modes supported by PenMount Utility: 4, 9, 16, and 25. During the calibration procedure, a circle with a red dot in the center will appear onscreen:

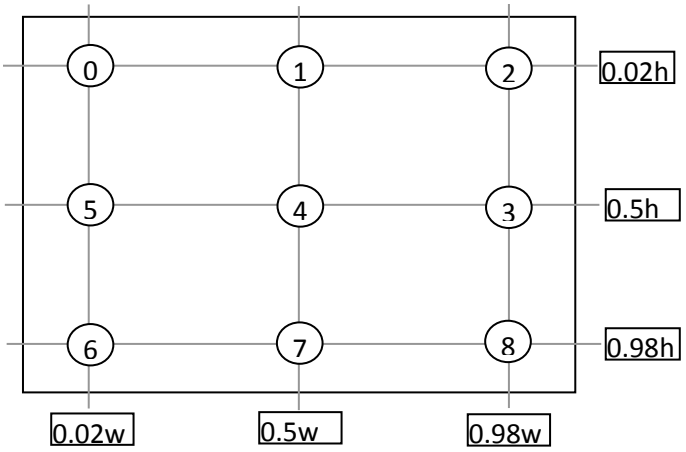


Please click on the red dot to proceed to the next point.

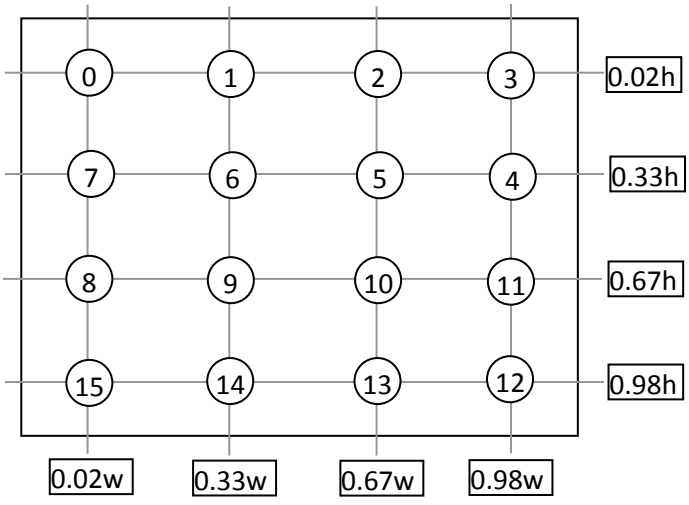
1. 4 Points Calibration Mode



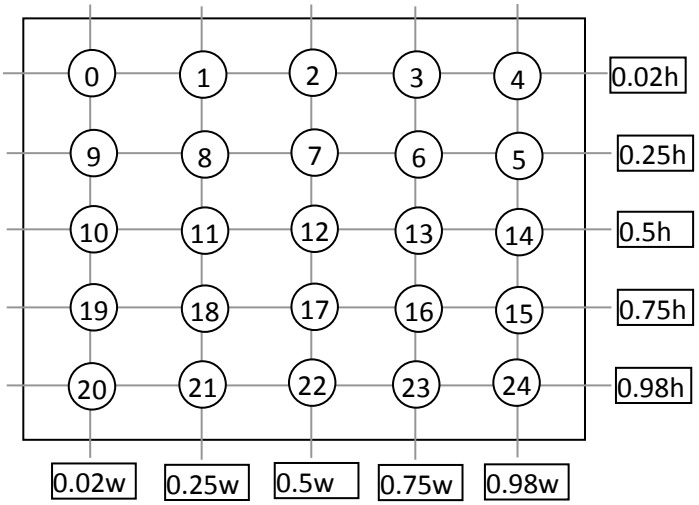
2. 9 Points Calibration Mode



3. 16 Points Calibration Mode



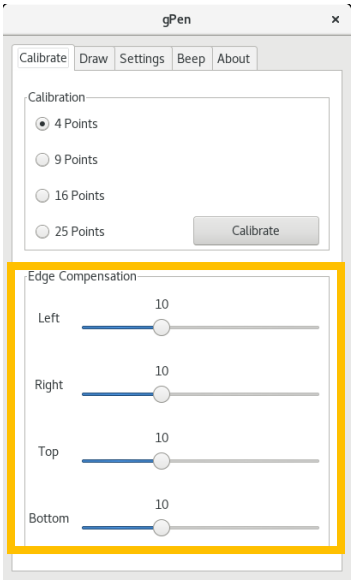
4. 25 Points Calibration Mode



The calibration ends when it collects all the touch point data it needs. The more points used, the more accurate touch will be after calibration.

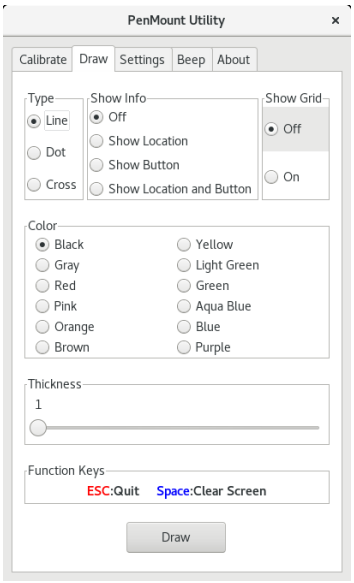
3.2. Edge Compensation

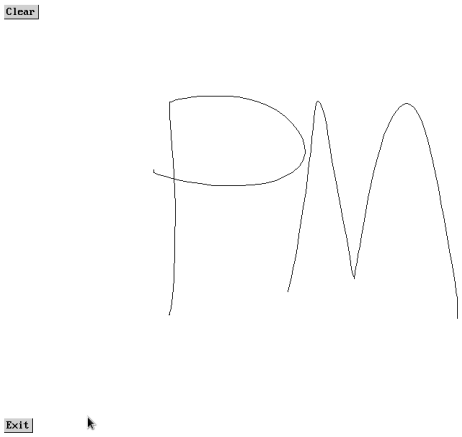
The PenMount Utility also supports adjustments on edges. The term “Edge Compensation” indicates the adjustment amount that makes touching on screen edges easier.



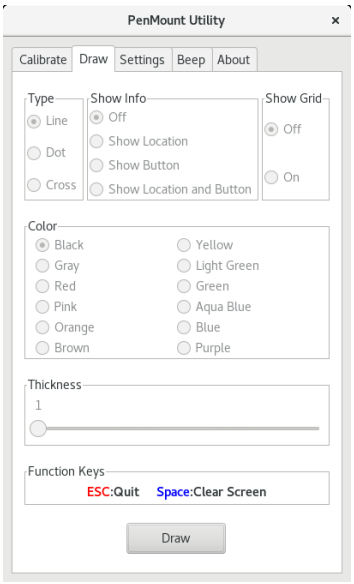
3.3. Draw Utility

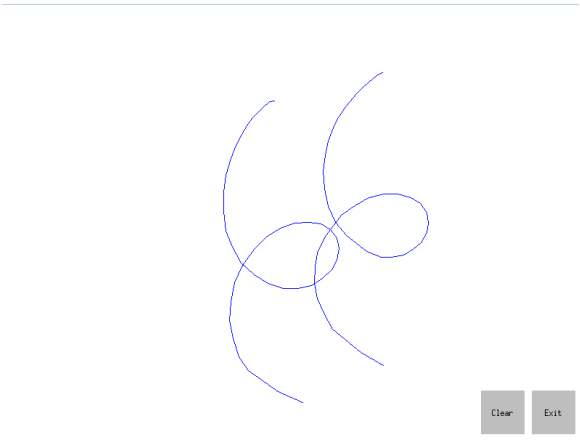
The PenMount Utility provides a draw program to allow users to test touch accuracy. Users may also change attributes such as “Draw Type”, “Show Touch Location and Button”, “Show Grid”, and “Change Color and Thickness”.





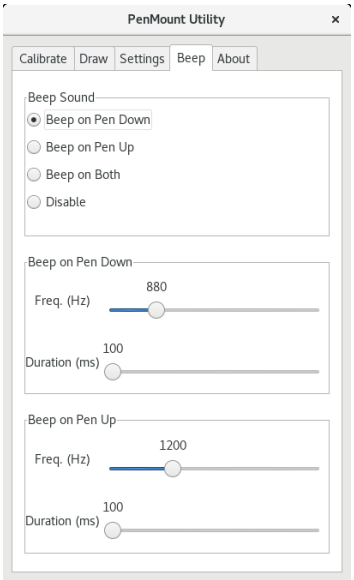
PenMount provides multi-touch testing functions in its Xorg server device driver V4.1 and later versions. If you are using PenMount P2 devices, please switch the operation mode to “Touch” first before testing multi-touch, otherwise only single-touch will be available.





3.4. Buzzer Beep

The PenMount Utility allows users to adjust the settings of touch buzzer beep includes the timing for sending buzzer beep, beep duration, and frequency.



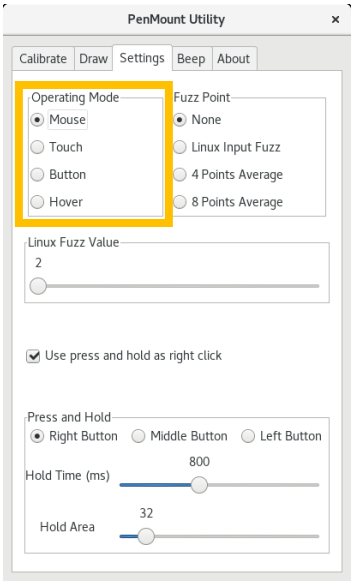
Please note that in some Linux distributions, the buzzer kernel driver (pcspkr) is disabled by default. In that case, changing the buzzer settings in PenMount Utility will not produce any effects.

3.5. Operation Modes

The operating modes control how the PenMount device driver sends input events. There are four modes supported:

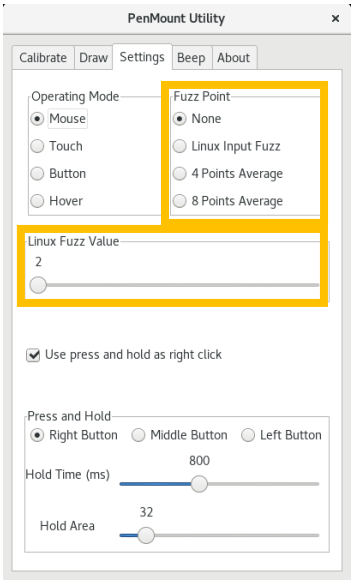
Mode	Motion and Button Event	Touch Event	Description
Mouse	○		ButtonPress: at first pen down MotionNofity: during pen down ButtonRelease: at pen up
Touch		○	XI_TouchBegin: at first pen down XI_TouchUpdate: during pen down XI_TouchEnd: at pen up
Button	○		ButtonPress: at first pen down ButtonRelease: at first pen down MotionNofity: not generated
Hover	○		ButtonPress: at pen up ButtonRelease: at pen up MotionNofity: during pen down

For PenMount P2 or P3 devices, the “Mouse”, “Button”, and “Hover” modes only provide single-touch functionality. Please notice that the “Touch” option will not be available on systems that do not support touch events.



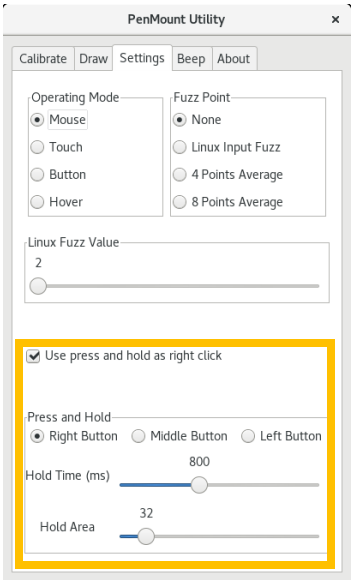
3.6. Smoothness Control

On some systems being affected by noise, the device driver can average touch points to make the output line smoother. There are three modes available, by using 4, 8 or arbitrary points (Linux Input Fuzz mode) for averaging.



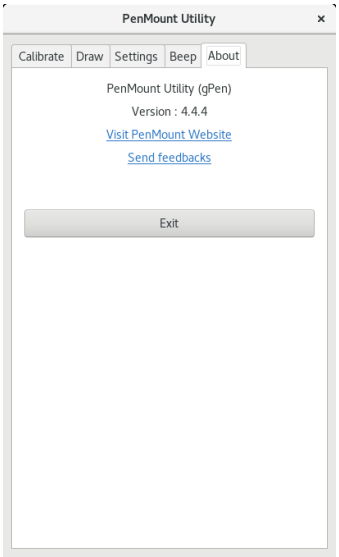
3.7. Press and Hold

The press and hold gesture can also be enabled with the PenMount Utility. It can also adjust attributes such as hold time, valid hold area, and the mouse event sent after hold threshold is reached.



3.8. General Information

This page displays version information and contact information, and also an “Exit” button for systems that does not come with “X” button in the title bar.



Please notice that the utility will display warning messages if running the without sufficient permissions to access the device driver configuration files.

3.9. Device Driver Setting File Format

The PenMount Utility stores calibration and other settings in a binary file, which can be found in the following location.

/etc/penmount/penmount.dat

Since this is not a text file, users need to use commands such as xxd to read the values in this file. The file format is described in the table below.

Offset	Length	Field	Description
0	1	Calibration Type	The calibration type used. Valid values are: 0: Standard calibration. 4: 4 Point advanced calibration. 9: 9 Point advanced calibration. 16: 16 Point advanced calibration. 25: 25 Point advanced calibration.
1	1	Panel Mode	The panel portrait mode, only used in standard calibration mode.
2	1	---	This field is not used.
3	1	Fuzz Point	The smoothing mode.
4	1	Fuzz Value	The value used when using “Linux Input Fuzz”.
5	1	Beep Type	The beep type.
6	1	Press and Hold Button	The button event type send when choosing press and hold gesture.
7	1	Touch Mode	The touch mode.
8	1	Press and Hold Enable	The enable state of the press and hold gesture.
9	3	---	These fields are not used.
12	2	Calib Point [0] X	The X axis value of the calibration point [0].
14	2	Calib Point [0] Y	The Y axis value of the calibration point [0].
16	2	Calib Point [1] X	The X axis value of the calibration point [1].
18	2	Calib Point [1] Y	The Y axis value of the calibration point [1].
20	2	Calib Point [2] X	The X axis value of the calibration point [2].
22	2	Calib Point [2] Y	The Y axis value of the calibration point [2].
24	2	Calib Point [3] X	The X axis value of the calibration point [3].
26	2	Calib Point [3] Y	The Y axis value of the calibration point [3].
28	2	Calib Point [4] X	The X axis value of the calibration point [4].
30	2	Calib Point [4] Y	The Y axis value of the calibration point [4].
32	2	Calib Point [5] X	The X axis value of the calibration point [5].
34	2	Calib Point [5] Y	The Y axis value of the calibration point [5].
36	2	Calib Point [6] X	The X axis value of the calibration point [6].
38	2	Calib Point [6] Y	The Y axis value of the calibration point [6].
40	2	Calib Point [7] X	The X axis value of the calibration point [7].
42	2	Calib Point [7] Y	The Y axis value of the calibration point [7].
44	2	Calib Point [8] X	The X axis value of the calibration point [8].
46	2	Calib Point [8] Y	The Y axis value of the calibration point [8].
48	2	Calib Point [9] X	The X axis value of the calibration point [9].

50	2	Calib Point [9] Y	The Y axis value of the calibration point [9].
52	2	Calib Point [10] X	The X axis value of the calibration point [10].
54	2	Calib Point [10] Y	The Y axis value of the calibration point [10].
56	2	Calib Point [11] X	The X axis value of the calibration point [11].
58	2	Calib Point [11] Y	The Y axis value of the calibration point [11].
60	2	Calib Point [12] X	The X axis value of the calibration point [12].
62	2	Calib Point [12] Y	The Y axis value of the calibration point [12].
64	2	Calib Point [13] X	The X axis value of the calibration point [13].
66	2	Calib Point [13] Y	The Y axis value of the calibration point [13].
68	2	Calib Point [14] X	The X axis value of the calibration point [14].
70	2	Calib Point [14] Y	The Y axis value of the calibration point [14].
72	2	Calib Point [15] X	The X axis value of the calibration point [15].
74	2	Calib Point [15] Y	The Y axis value of the calibration point [15].
76	2	Calib Point [16] X	The X axis value of the calibration point [16].
78	2	Calib Point [16] Y	The Y axis value of the calibration point [16].
80	2	Calib Point [17] X	The X axis value of the calibration point [17].
82	2	Calib Point [17] Y	The Y axis value of the calibration point [17].
84	2	Calib Point [18] X	The X axis value of the calibration point [18].
86	2	Calib Point [18] Y	The Y axis value of the calibration point [18].
88	2	Calib Point [19] X	The X axis value of the calibration point [19].
90	2	Calib Point [19] Y	The Y axis value of the calibration point [19].
92	2	Calib Point [20] X	The X axis value of the calibration point [20].
94	2	Calib Point [20] Y	The Y axis value of the calibration point [20].
96	2	Calib Point [21] X	The X axis value of the calibration point [21].
98	2	Calib Point [21] Y	The Y axis value of the calibration point [21].
100	2	Calib Point [22] X	The X axis value of the calibration point [22].
102	2	Calib Point [22] Y	The Y axis value of the calibration point [22].
104	2	Calib Point [23] X	The X axis value of the calibration point [23].
106	2	Calib Point [23] Y	The Y axis value of the calibration point [23].
108	2	Calib Point [24] X	The X axis value of the calibration point [24].
110	2	Calib Point [24] Y	The Y axis value of the calibration point [24].
112	2	Beep frequency on pen down	The beep frequency on pen down.
114	2	Beep duration on pen down	The beep duration on pen down.
116	2	Beep frequency on pen up	The beep frequency on pen up.
118	2	Beep duration on pen up	The beep duration on pen up.
120	2	Press and hold duration	The minimum hold time to become a valid press and hold gesture.
122	2	Press and hold offset	The minimum hold area to become a valid press and hold gesture.
124	2	Left Edge Compensate Value	The compensated value on the left side of the touch panel.
126	2	Right Edge Compensate	The compensated value on the right side of the touch

		Value	panel.
128	2	Top Edge Compensate Value	The compensated value on the top side of the touch panel.
130	2	Bottom Edge Compensate Value	The compensated value on the bottom side of the touch panel.
132	4	CRC	The CRC of this file.

4. Tiny X Server

The Tiny X Server (also called KDrive) is a small sized X Server that is used in distributions such as Tiny Core Linux. The standard PenMount device driver for X Server cannot be used on Tiny X and we recommend using the built-in tslib driver and the utilities provided by tslib for calibration. For this reason, if the Tiny X server you are using does not come with tslib support, you will need to follow the instructions in this section to configure and compile the Tiny X server again.

4.1. Configure the Tiny X Server

To configure the Xorg X Server for building TinyX (KDrive) with tslib support, please first modify configure.ac file and then run autoconf to generate the configure file.

4.1.1. X11R7.1

The X11R7.1 release is the first version of Xorg X Server that includes kdrive support and requires more modifications to use the tslib driver than later X11 releases.

To configure the Xorg X Server for building TinyX (KDrive) with tslib support, please first modify configure.ac file and then run autoconf to generate the configure file.

Please find and replace the following sections in configure.ac file according to the table below.

Before	After
AM_CONDITIONAL(TSLIB, false)	AM_CONDITIONAL(TSLIB, true)
---	AC_DEFINE(TOUCHSCREEN,1,[Have touchscreen support])

After the configure file is generated, please enable the following options during configuration.

Options	Description
--enable-kdrive	Build kdrive servers (default: no)

4.1.2. X11R7.2

For X11R7.2 release, please first modify configure.ac file and then run autoconf to generate the configure file.

Before	After
AC_DEFINE(TSLIB,1,[Have touchscreen support])	AC_DEFINE(TOUCHSCREEN,1,[Have touchscreen support])

After the configure file is generated, please enable the following options during configuration.

Options	Description
--enable-kdrive	Build kdrive servers (default: no)
--enable-tslib	Build kdrive tslib touchscreen support (default: disabled)

4.1.3. X11R7.3 and above

For X11R7.3 and later X11 releases, there is no need to modify configure.ac file, please proceed with configure directly with the following options enabled.

Options	Description

--enable-kdrive	Build kdrive servers (default: no)
--enable-tslib	Build kdrive tslib touchscreen support (default: disabled)

4.2. Adding the tslib Driver

The tslib driver is included in the KDrive source code. For X11 R7.1 and R7.2, however, if you are using Xvesa, the source code needs to be modified to use tslib. Please follow the instructions below and then proceed to the server building process.

[hw/kdrive/vesa/vesainit.c]

```
void
InitInput (int argc, char **argv)
{
        // …
#if defined(TOUCHSCREEN)
        KdAddMouseDriver (&TsFuncs);
#endif
}
```

4.3. Using the tslib Driver

After building the Tiny X server, the tslib driver can be used. For example, if using Xvesa with 800x600x24 screen resolution, and the PenMount touch device name is /dev/input/event5, please run the following commands with root permissions to launch the server with tslib support.

```
# export the TSLIB_TSDEVICE environment variable, which will be used by ts_calibrate
export TSLIB_TSDEVICE=/dev/input/event5
# Calibrate the touch screen
ts_calibrate
# Run the X server, set mouse driver to tslib.
Xvesa -screen 800x600x24 -mouse tslib,,device=/dev/input/event5
```

5. PenMount SDK for X

The PenMount SDK provides API for developers to control the touch functions that PenMount provides. The latest SDK version is 1.0. If you need extra functions that are not listed in this chapter, please contact PenMount for further support.

5.1. File Contents

The PenMount SDK includes header files and libraries that support both 32 and 64 bit x86 systems.

Type	File Name	Description
Header	penmount_sdk.h	Definitions of the macros and function prototypes.
Library	i686/penmount_sdk.so	The 32 bit shared library file.
	x86_64/penmount_sdk.so	The 64 bit shared library file.
Sample	pmsdk_test.c build.sh	Sample code that demonstrates how to use the API.

5.2. Additional Requirements

There are some additional requirements for using the PenMount SDK which are listed in the table below.

Item	Description
Controller	PenMount 6000, 9000, P2 controllers. Currently only RS-232 interface is supported.
Device Driver	The PenMount device driver for Xorg server.
Library	The following shared libraries is also needed. libX11.so : The X11 library libXi.so : The X Input library

5.3. Using the Library

The PenMount SDK includes a shared library that can be loaded by developers.

5.3.1. Open the Library

Please use dlopen() to open the library file. Please note that you need to provide a correct path to the library file, or the returned value will be NULL.

```
void * pLibrary = NULL;
pLibrary = dlopen("i686/penmount_sdk.so", RTLD_LAZY);
```

5.3.2. Load the Symbols

Please use dlsym() to load the symbols that the library provides. If the symbol does not exist, the retuned value will be NULL.

```
typedef long (*pfnpmsdk_get_version)(void);
pfnpmsdk_get_version pmsdk_get_version = NULL;
pmsdk_get_version = dlsym(pLibrary, "pmsdk_get_version");
```

5.3.3. Build the Code

To build the code, please include the dl library that provides the dlopen() and dlsym() function.

```
gcc pmsdk_test.c -o pmsdk_test -ldl
```

5.4. Functions

This section lists the functions provided by the PenMount SDK.

5.4.1. Get the SDK Version

Syntax

```
long    pmsdk_get_version (    void    )
```

Description

This function returns the version of the SDK.

Return Values

The version number of the SDK. Please use the following macros to get the detailed versions.

Macro	Description
PMSDK_GET_MAJOR_VERSION	The major version value of the SDK.
PMSDK_GET_MINOR_VERSION	The minor version value of the SDK.
PMSDK_GET_BUILD_VERSION	The minor version value of the SDK.

5.4.2. Disable a PenMount Serial Device

Syntax

```
int     pmsdk_disable_device (    void    )
```

Description

This function will disable both the PenMount X driver and the controller.

Return Values

If function succeeds, the return value is nonzero.

If function fails, the return value is zero.

5.4.3. Enable a PenMount Serial Device

Syntax

```
int pmsdk_enable_device ( void )
```

Description

This function will enable both the PenMount X driver and the controller.

Return Values

If function succeeds, the return value is nonzero.

If function fails, the return value is zero.