



FocusLCDs.com  
LCDs MADE SIMPLE®

Ph. 480-503-4295 | [NOPP@FocusLCD.com](mailto:NOPP@FocusLCD.com)

TFT | CHARACTER | UWVD | FSC | SEGMENT | CUSTOM | REPLACEMENT

## Application Note FAN4203

### *Calibrating Touch Screens*

This application note is part 2 of our 2 part TFT LCD screen series. Part 1 discusses configuring the screen resolution of a TFT with a Raspberry Pi. Part 2 discusses calibrating the touch screen of a TFT. [Click here to read part 1.](#)

## Calibrating the Touch Screen on Raspberry Pi

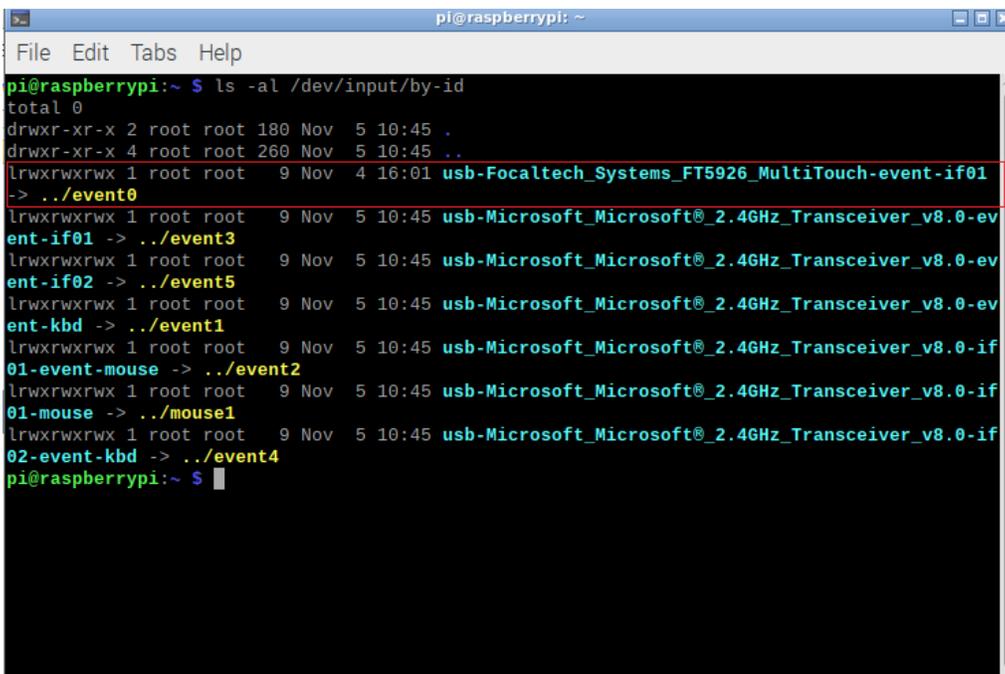
---

After the display is setup with the Raspberry Pi you may want to calibrate the touch feature for accuracy. The display used in this example is a 7" TFT with capacitive touch ([E70RA-HW520-C](#)). This is a continuation of a previous application with the same display. The touch feature is auto enabled upon download of the Raspbian software; however, it may be inconsistent with actual motion. Calibrating the touch feature of a display can be done in just a few simple steps.

1.) First, we will verify that the Raspberry Pi recognizes the display. To do this we will run a command that specifies where the input devices are connected. Open Terminal and execute the following command:

```
ls -al /dev/input/by-id
```

You will get a list of the various connection ports and where they are located. The display is recognized and is located at event0 as specified below.



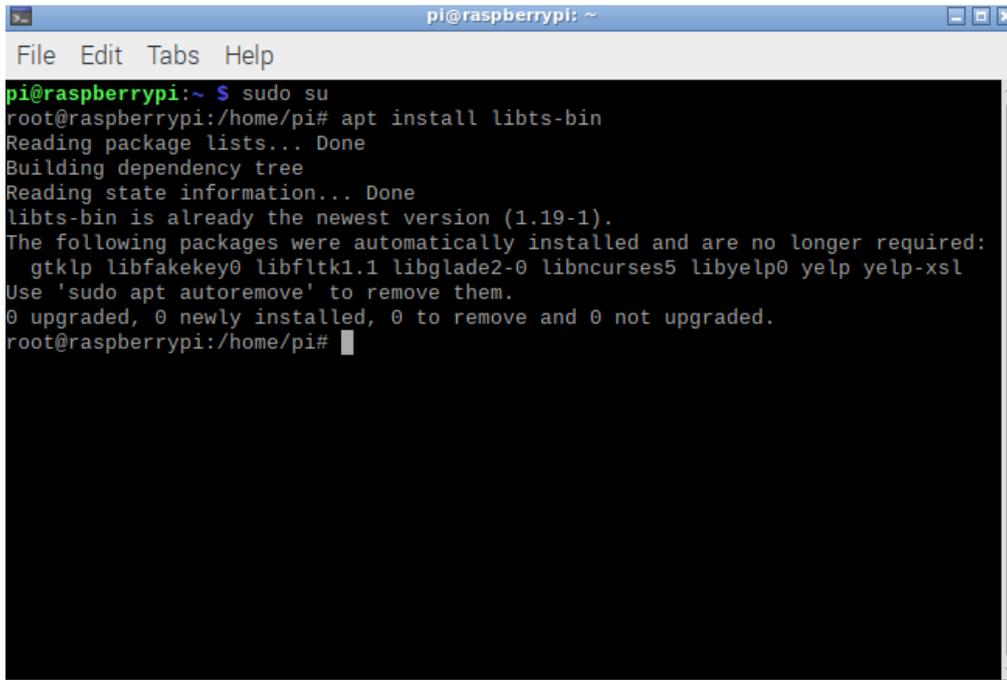
```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~ $ ls -al /dev/input/by-id  
total 0  
drwxr-xr-x 2 root root 180 Nov  5 10:45 .  
drwxr-xr-x 4 root root 260 Nov  5 10:45 ..  
lrwxrwxrwx 1 root root   9 Nov  4 16:01 usb-Focaltech_Systems_FT5926_MultiTouch-event-if01  
-> ./event0  
lrwxrwxrwx 1 root root   9 Nov  5 10:45 usb-Microsoft_Microsoft@_2.4GHz_Transceiver_v8.0-event-if01 -> ./event3  
lrwxrwxrwx 1 root root   9 Nov  5 10:45 usb-Microsoft_Microsoft@_2.4GHz_Transceiver_v8.0-event-if02 -> ./event5  
lrwxrwxrwx 1 root root   9 Nov  5 10:45 usb-Microsoft_Microsoft@_2.4GHz_Transceiver_v8.0-event-kbd -> ./event1  
lrwxrwxrwx 1 root root   9 Nov  5 10:45 usb-Microsoft_Microsoft@_2.4GHz_Transceiver_v8.0-if01-event-mouse -> ./event2  
lrwxrwxrwx 1 root root   9 Nov  5 10:45 usb-Microsoft_Microsoft@_2.4GHz_Transceiver_v8.0-if01-mouse -> ./mouse1  
lrwxrwxrwx 1 root root   9 Nov  5 10:45 usb-Microsoft_Microsoft@_2.4GHz_Transceiver_v8.0-if02-event-kbd -> ./event4  
pi@raspberrypi:~ $
```

2.) Next, we find the program that calibrates the display. There are a few options of programs that calibrate the touch setting, the one I will be using is called `ts_lib`. We can install this through the Terminal. Root privileges will need to be enabled to install the program that is used. Enable super user mode by entering:

```
sudo su
```

Now we will download the program that calibrates the display. This program runs a quick test that verifies user input. Make sure the Raspberry Pi is connected to the internet for this step. To download this file, enter the following in the Terminal:

```
apt install libts-bin
```



```
pi@raspberrypi:~ $ sudo su
root@raspberrypi:/home/pi# apt install libts-bin
Reading package lists... Done
Building dependency tree
Reading state information... Done
libts-bin is already the newest version (1.19-1).
The following packages were automatically installed and are no longer required:
  gtklp libfakekey0 libfltk1.1 libglade2-0 libncurses5 libyelp0 yelp yelp-xsl
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@raspberrypi:/home/pi#
```

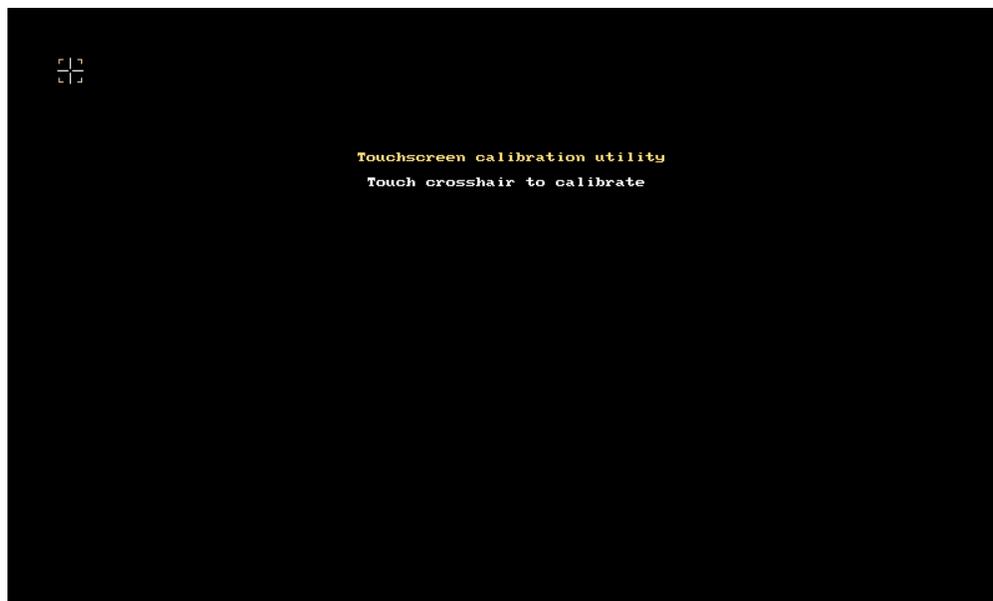
This will install a configuration file for the touch screen. Before the test we need to specify which device we're using and where it is located. The display is located at "event0" as verified in the previous step. This could be a different value for you depending on the setup so be sure to refer to the results of step one. Enter the following commands to define the display variables.

```
export tslib_tsdevice=/dev/input/event0
```

```
export tslib_fbdevice=/dev/fb1
```

```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~ $ sudo su  
root@raspberrypi:/home/pi# apt install libts-bin  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
libts-bin is already the newest version (1.19-1).  
The following packages were automatically installed and are no longer required:  
  gtklp libfakekey0 libfltk1.1 libglade2-0 libncurses5 libyelp0 yelp yelp-xsl  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
root@raspberrypi:/home/pi# export tslib_tsdevice=/dev/input/event0  
root@raspberrypi:/home/pi# export tslib_fbdevice=/dev/fb1  
root@raspberrypi:/home/pi#
```

3.) Now we can run the calibration test. To do this enter the `ts_calibrate` command in the Terminal. You will need to be in super user mode for the calibration test to run. A screen will pop up asking you to touch five different points on the screen, try to touch them as accurately as possible. After the test is finished a list of the results will be displayed in the Terminal.



```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo su
root@raspberrypi:/home/pi# ts_calibrate
xres = 1023, yres = 596
Took 12 samples...
Top left : X = 44 Y = 61
Took 18 samples...
Top right : X = 985 Y = 67
Took 18 samples...
Bot right : X = 984 Y = 538
Took 15 samples...
Bot left : X = 48 Y = 547
Took 16 samples...
Center : X = 515 Y = 304
5.674683 0.983473 -0.003163
-17.258911 0.001635 1.036311
Calibration constants: 371896 64452 -207 -1131080 107 67915 65536
root@raspberrypi:/home/pi#

```

4.) You can now test the calibration by running the `ts_test` command in the Terminal. This will run a short program to verify the calibration results by dragging a pointer or drawing an image. When you are finished testing, press quit, and the test information will be displayed in the Terminal. If you are unsatisfied with the accuracy of the test you can run through the `ts_calibrate` program again.

```

pi@raspberrypi: ~
File Edit Tabs Help
ts_test
root@raspberrypi:/home/pi# ts_test
1573061395.672767:    4    284    255
1573061395.681745:   512    290    255
1573061395.691875:   511    290    255
1573061395.701716:   511    290    255
1573061395.711737:   511    290    255
1573061395.721721:   511    290    255
1573061395.730748:   511    290    255
1573061395.740784:   511    290    255
1573061395.750743:   511    290    255
1573061395.760747:   511    290     0
1573061395.809752:   543    295    255
1573061395.819722:   547    298    255
1573061395.828738:   550    300     0
1573061395.858755:   575    292    255
1573061395.868742:   577    292    255
1573061395.878754:   581    290    255
1573061395.887746:   584    289    255
1573061395.897746:   588    289    255
1573061395.907761:   592    288    255
1573061395.917756:   594    288    255
1573061395.927751:   596    288    255
1573061395.936745:   597    288    255

```

Once you are satisfied with the results and the reboot the system.

```
sudo reboot
```

The touch screen should now be configured to a precise setting. This is just a brief overview of how to calibrate the capacitive touch feature of a display. For further calibration settings you can visit the “ts.config” file. You can open this file from Terminal by entering:

```
sudo nano /etc/ts.conf
```

For more information on these calibration settings and others visit the [tslib documentation](#) website.

## DISCLAIMER

Buyers and others who are developing systems that incorporate FocusLCDs products (collectively, “Designers”) understand and agree that Designers remain responsible for using their independent analysis, evaluation and judgment in designing their applications and that Designers have full and exclusive responsibility to assure the safety of Designers' applications and compliance of their applications (and of all FocusLCDs products used in or for Designers' applications) with all applicable regulations, laws and other applicable requirements.

Designer represents that, with respect to their applications, Designer has all the necessary expertise to create and implement safeguards that:

- (1) anticipate dangerous consequences of failures
- (2) monitor failures and their consequences, and
- (3) lessen the likelihood of failures that might cause harm and take appropriate actions.

Designer agrees that prior to using or distributing any applications that include FocusLCDs products, Designer will thoroughly test such applications and the functionality of such FocusLCDs products as used in such applications.