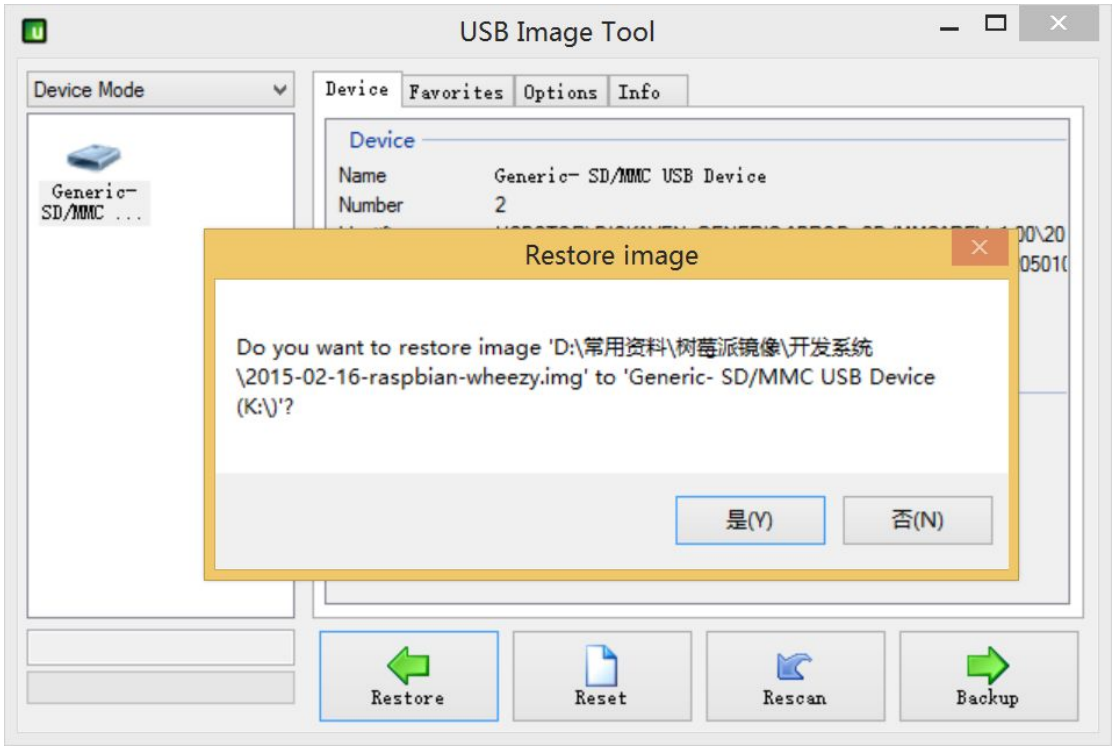
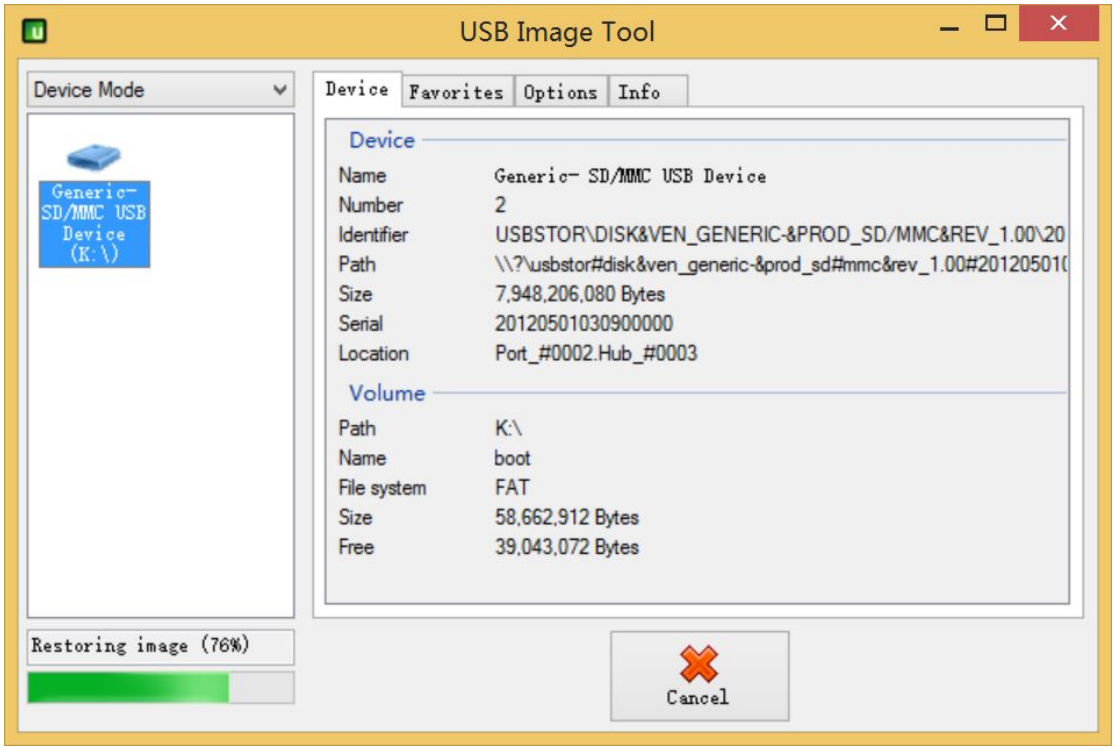
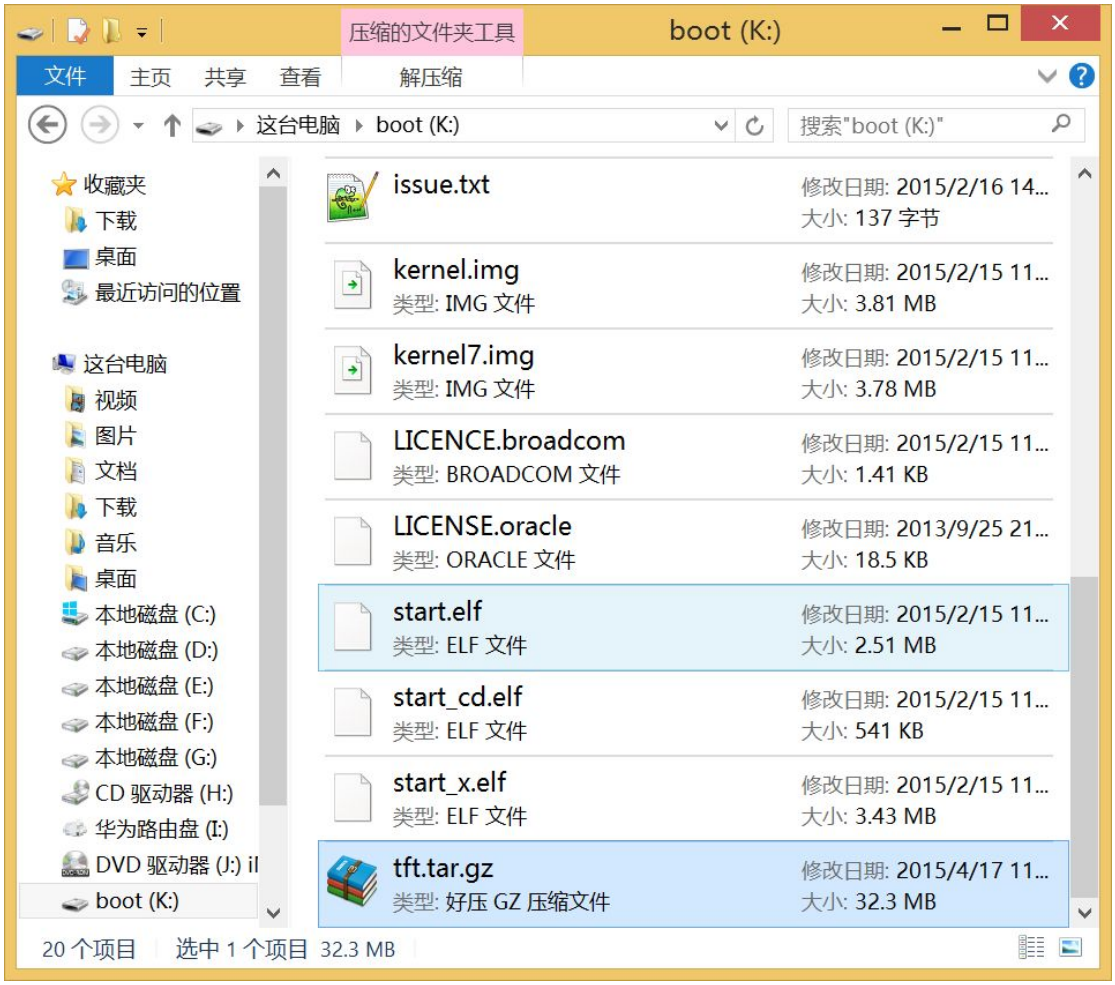
Note: manually install its complexity!

1) Latest any system programming:



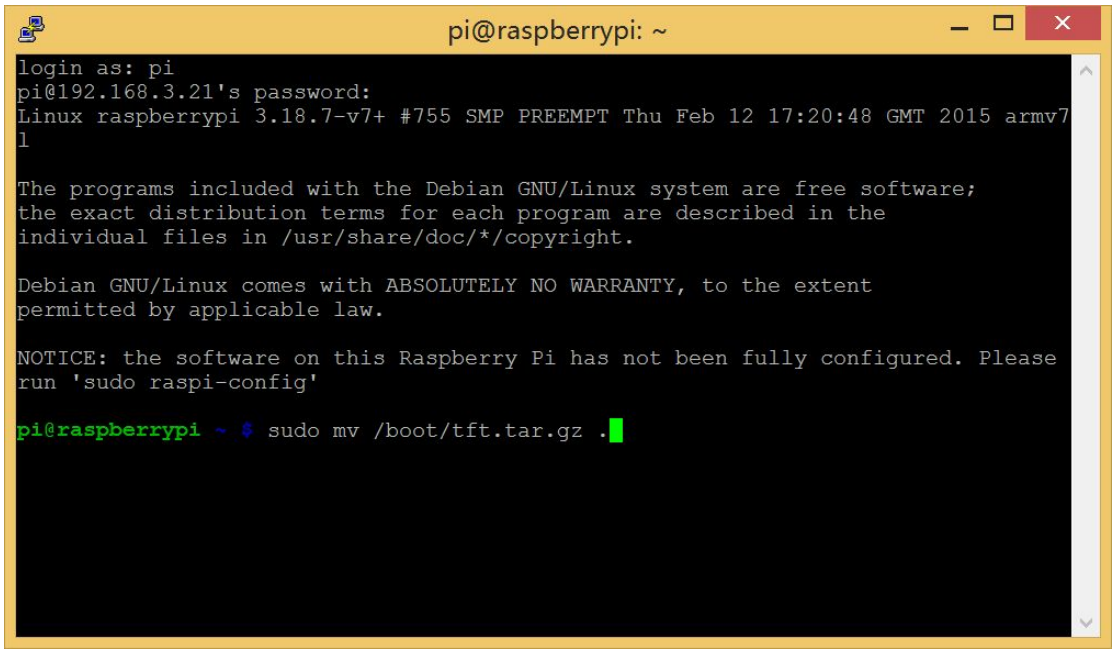


2) Copy the driver package to the / boot partition, this driver package will be constantly updated, and this driver package can not be opened in Windows, for compatibility with different screen follow-up, the drive also supports Pi / Pi2:

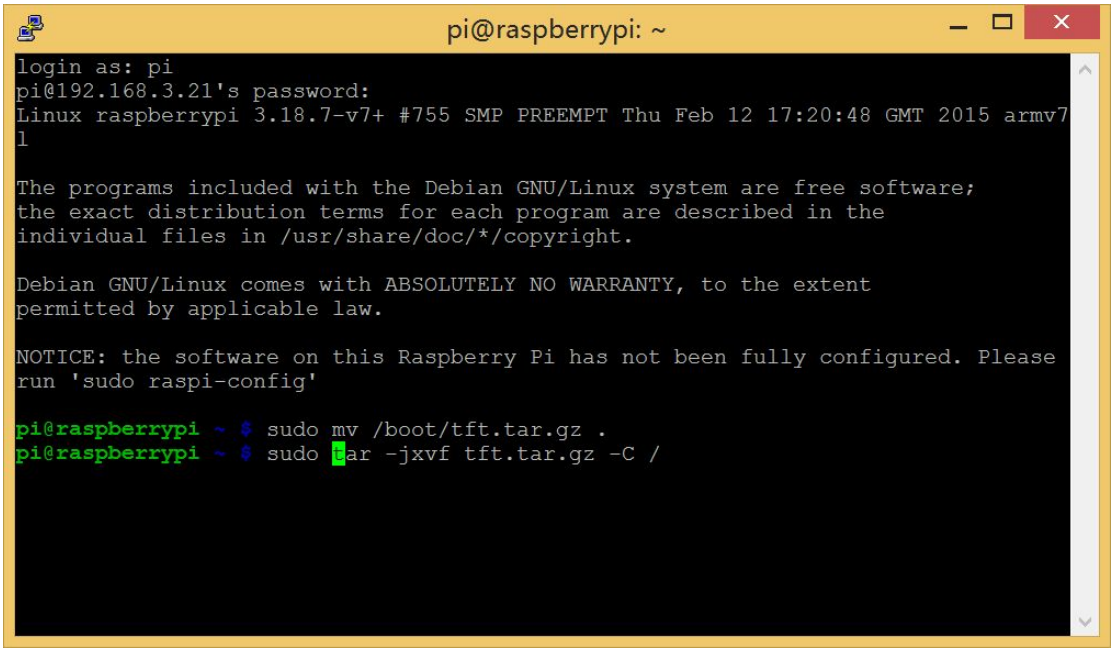


3) Raspberry Pi started using TTL or SSH to log in to Raspberry Pi.

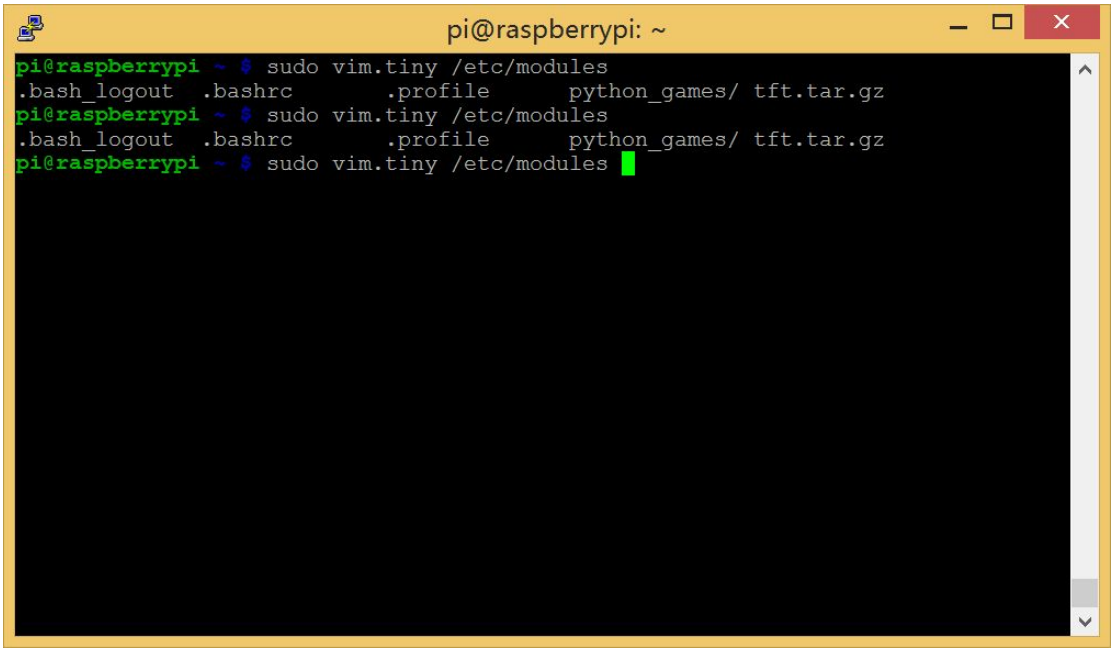
4) To avoid driver installation fails, such as permissions issues, space issues, the driver package to move to the current directory, the following command:



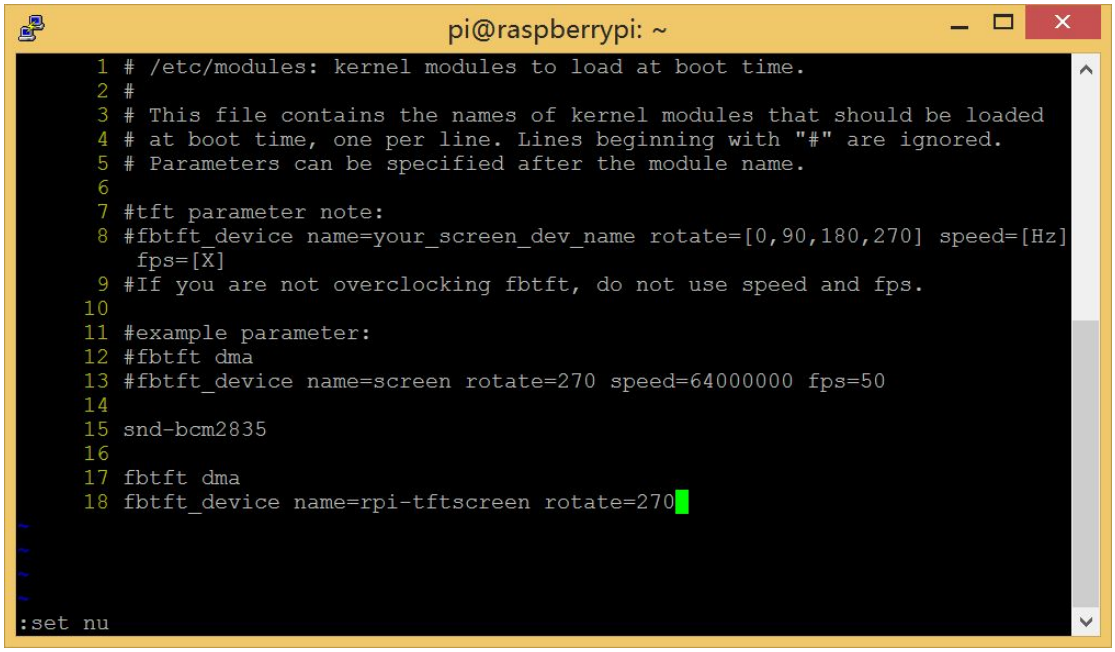
5) Driven decompression is then performed, the following command:



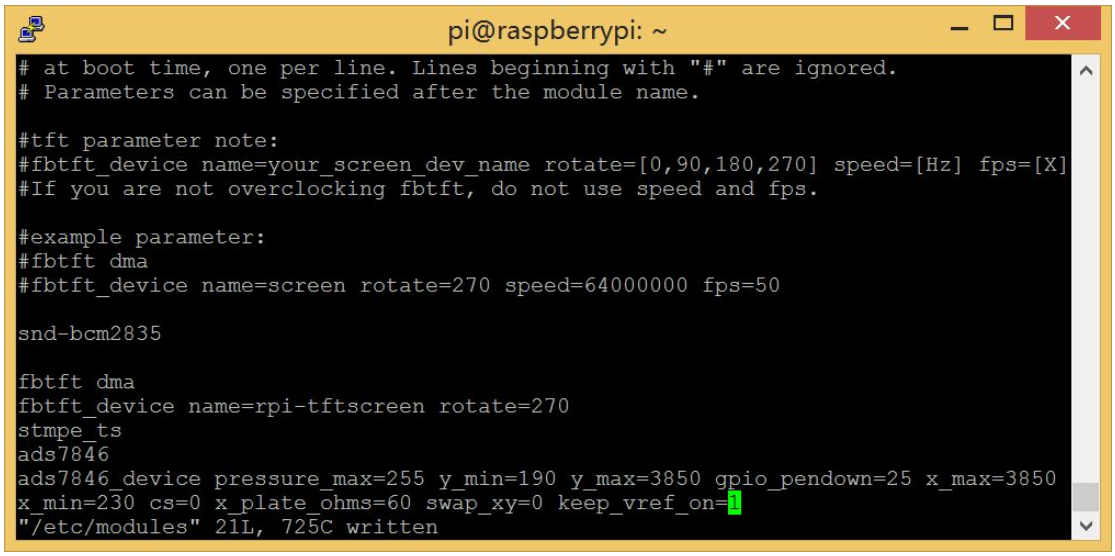
6) Modify / etc / modules file, you can use your familiar editor, such as nano, vim, etc., ordered as follows:



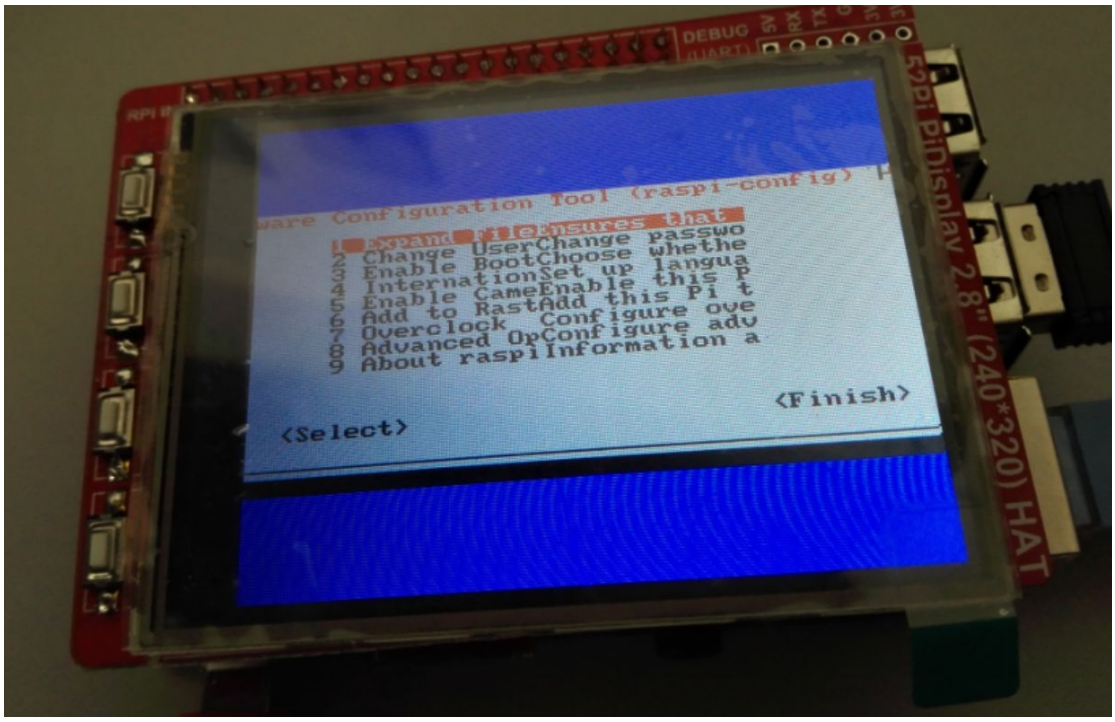
7) Add the following two lines, you can enable the screen driver [Line 17 - 18], if you use the other screen, replace rpi-tftscreen fields:



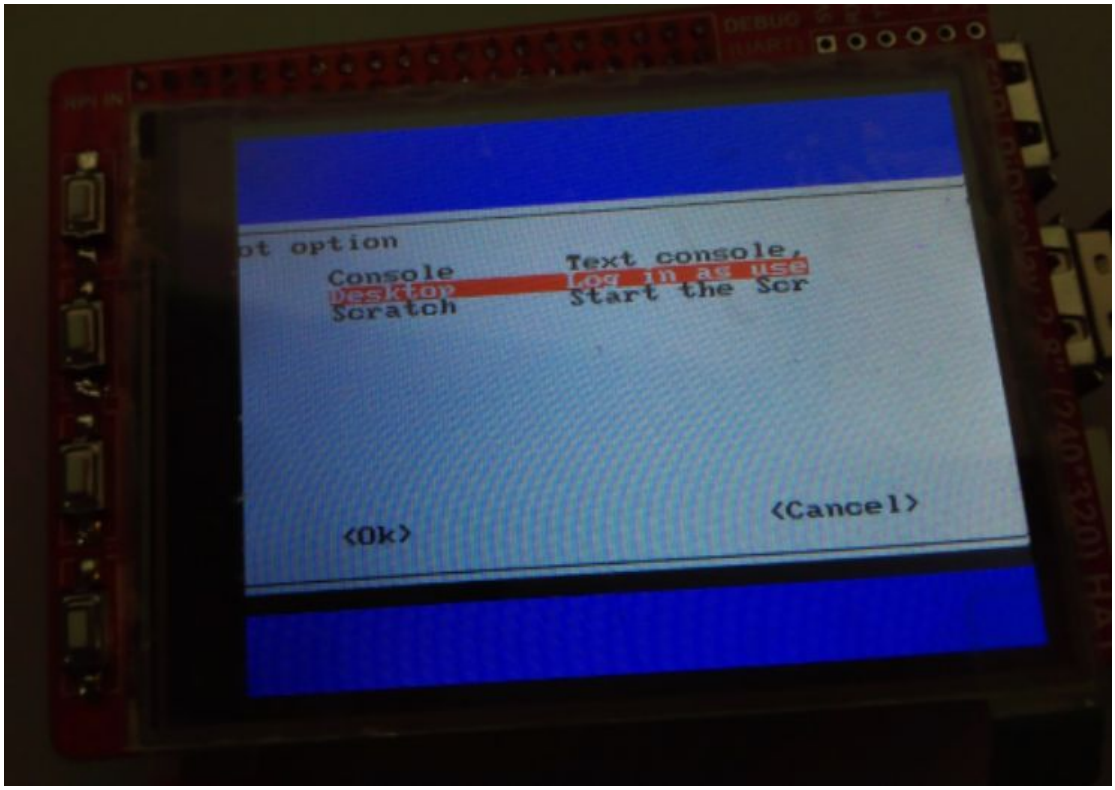
8) If you need a touch screen driver, also add the following line [Line 19 - 21], you can fine-tune these parameters:



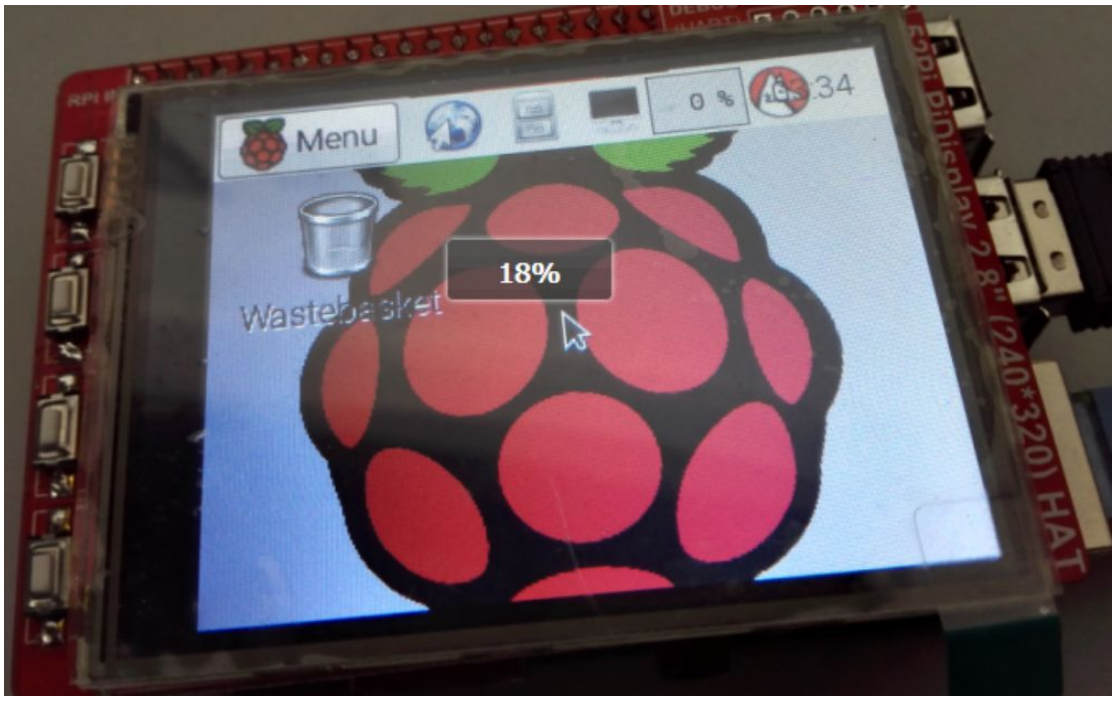
9) Now, you can restart your Raspberry Pi, enter the initial configuration, we can adjust to let him start to the desktop, you can also choose other ways.



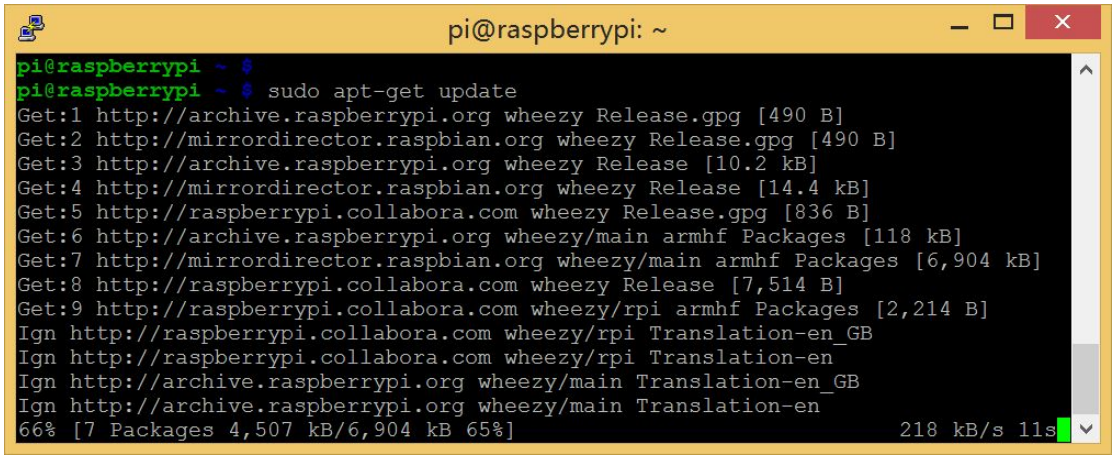
10) Select the boot to the desktop:

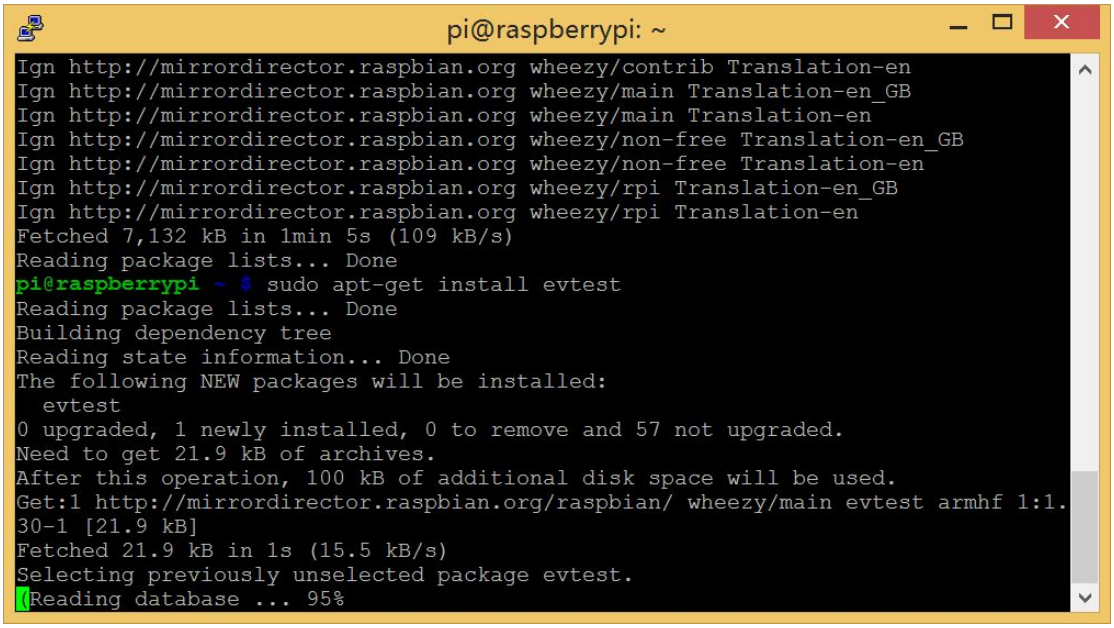


11) Now Restart your raspberrypi, you can boot to the desktop.

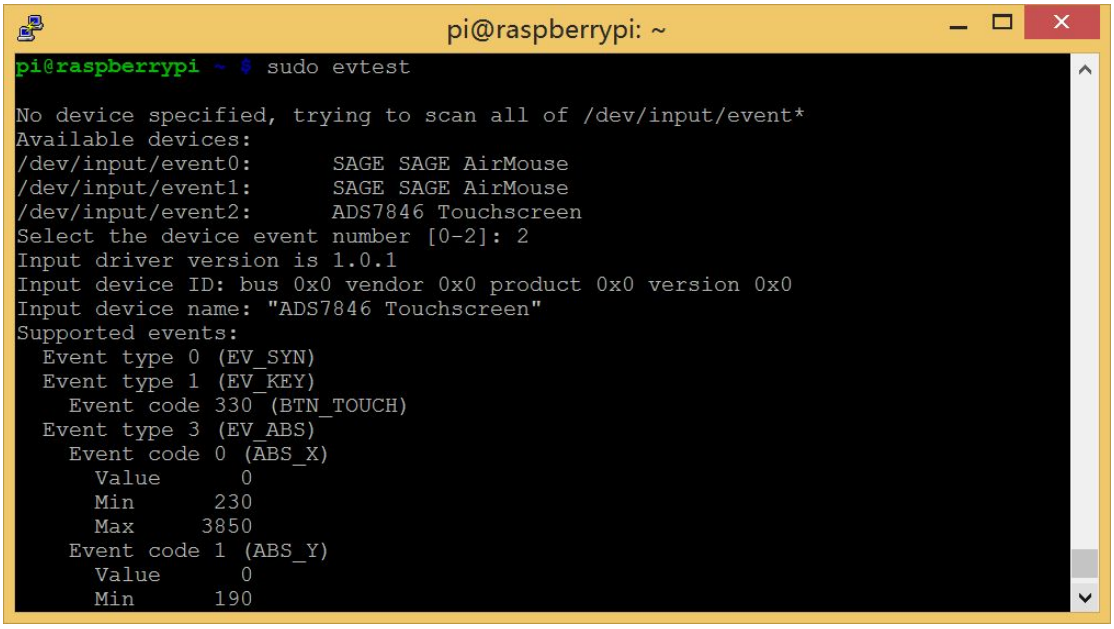


12) Re-use SSH, or serial port, or a keyboard and mouse connected Raspberry Pi, installation evtest tools, below is a method Raspbian:

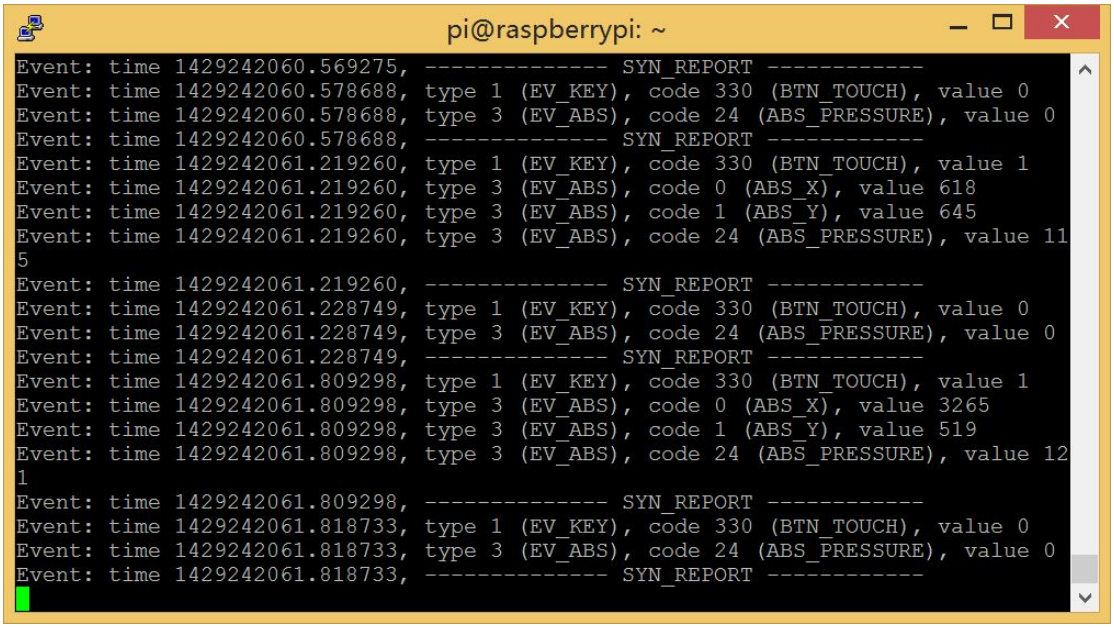




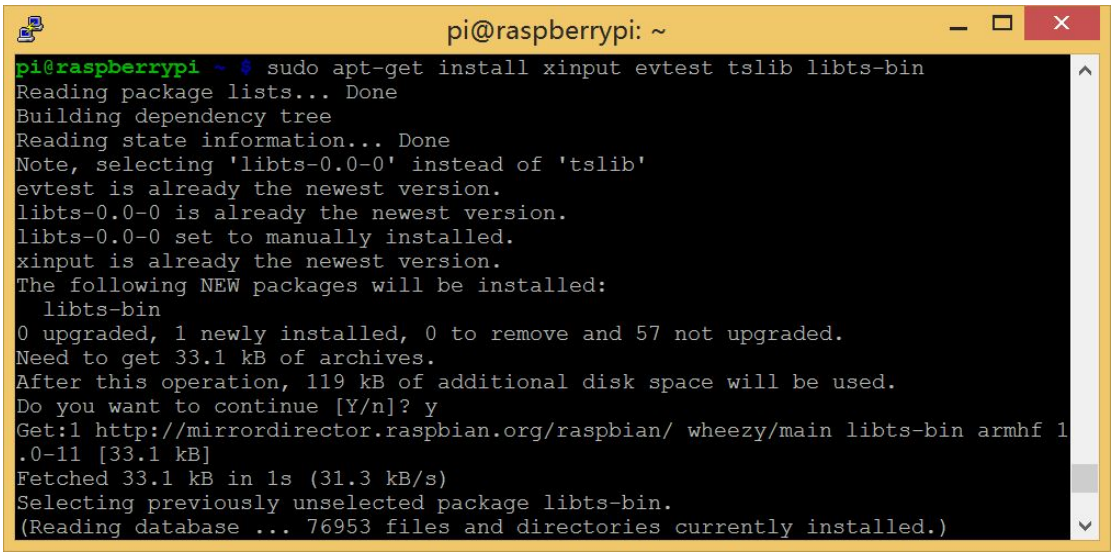
13) Use evtest test touch:



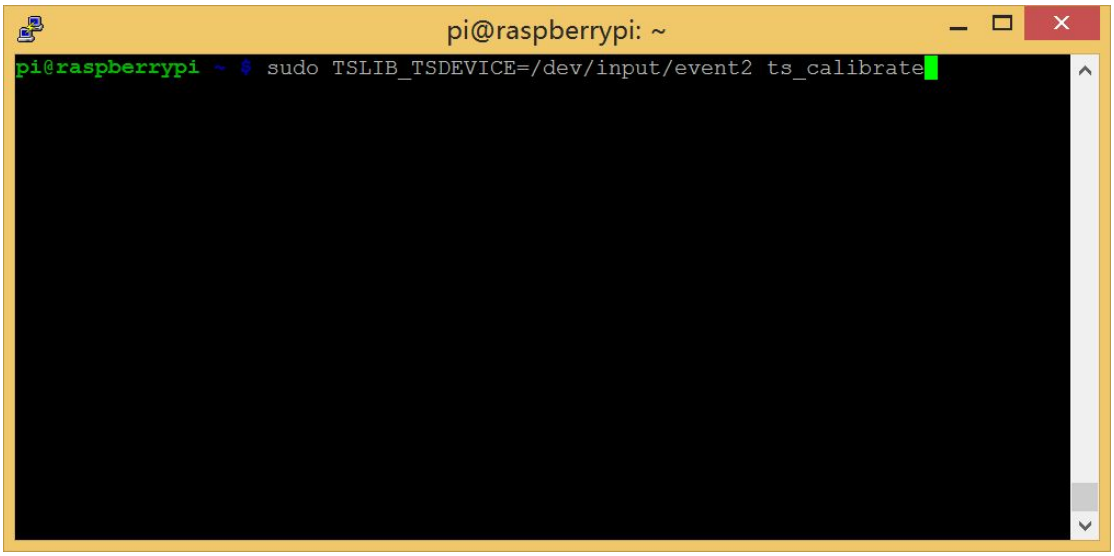
14) Random touch will make a lot of data:



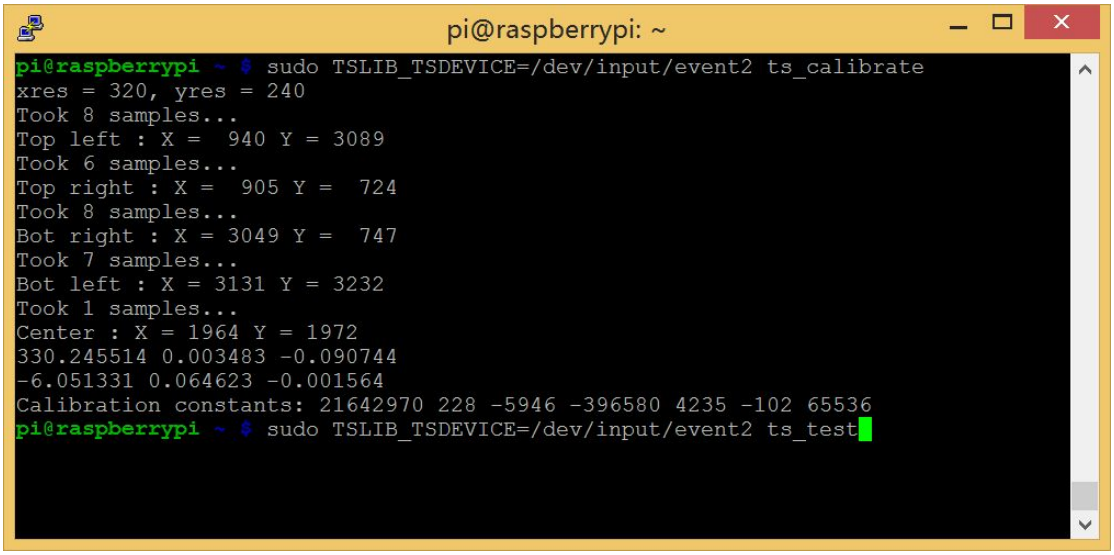
15) Then install the calibration module:

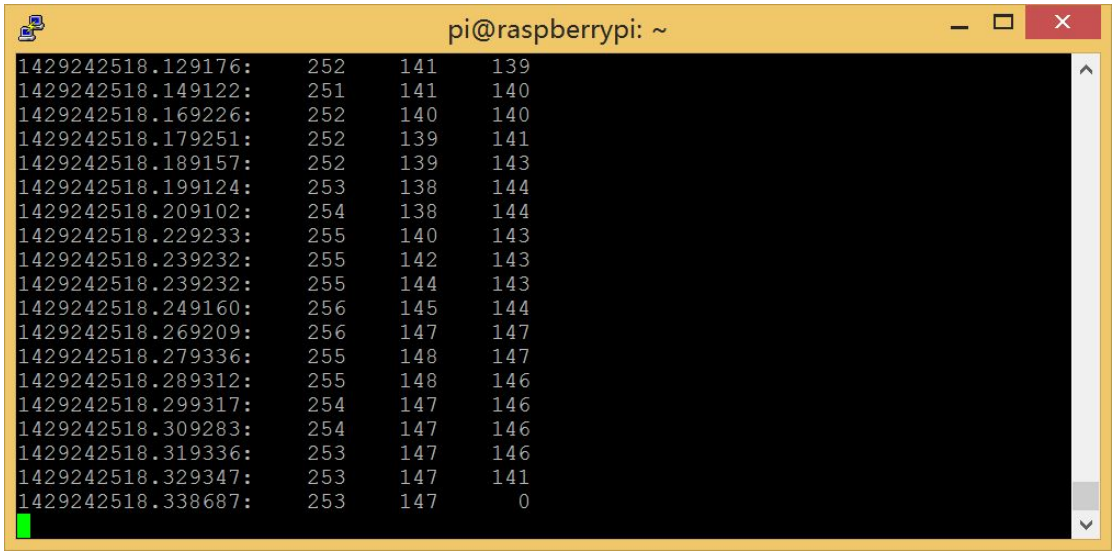


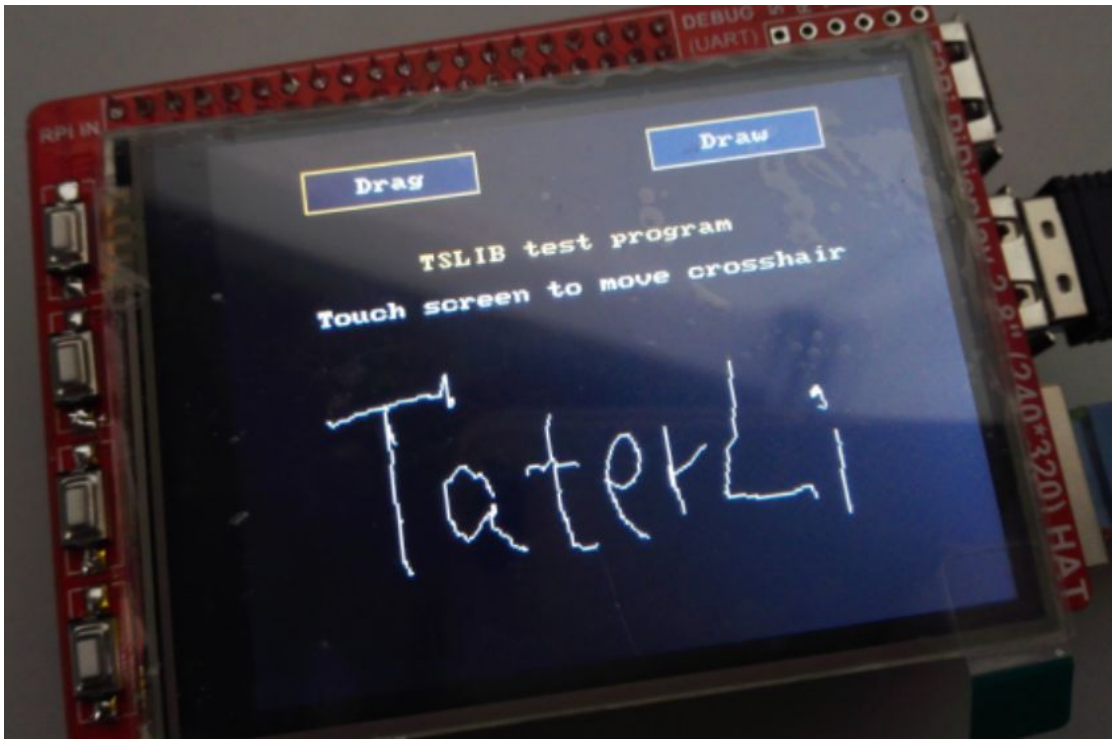
16) Just know that it is / dev / input / event2 for the touch screen, so this correction.



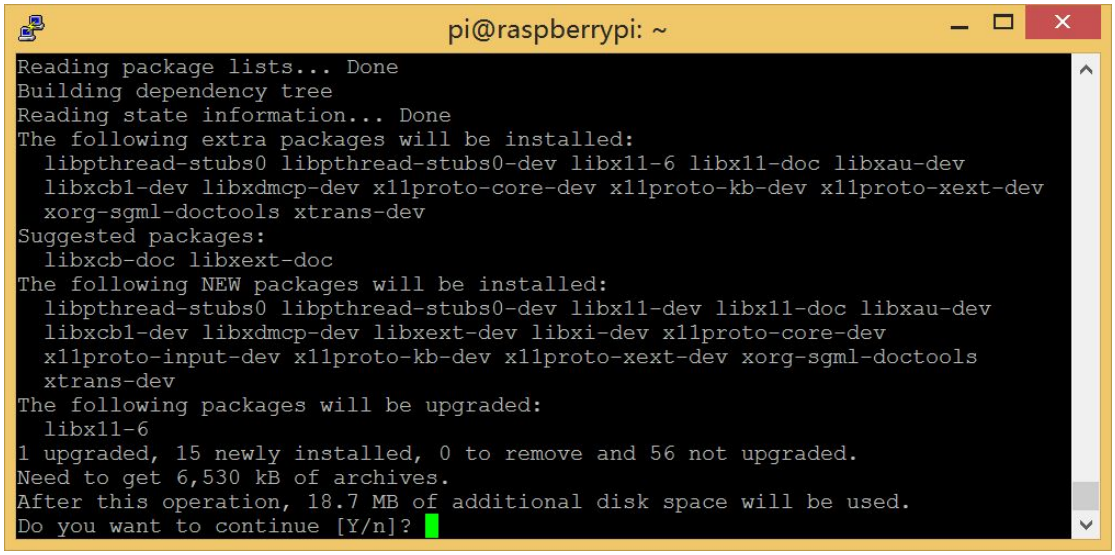
17) You can make some tests:

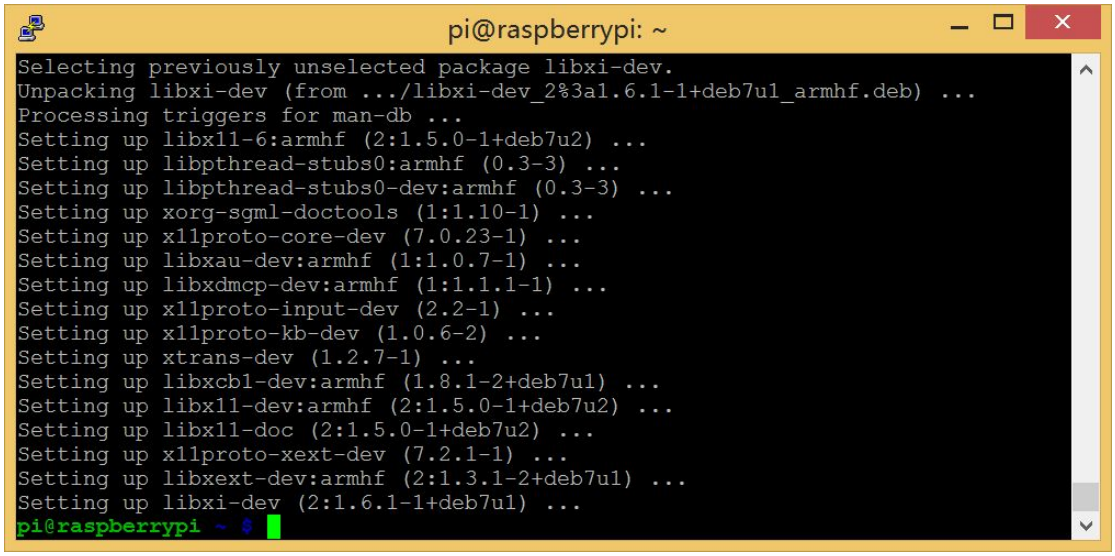




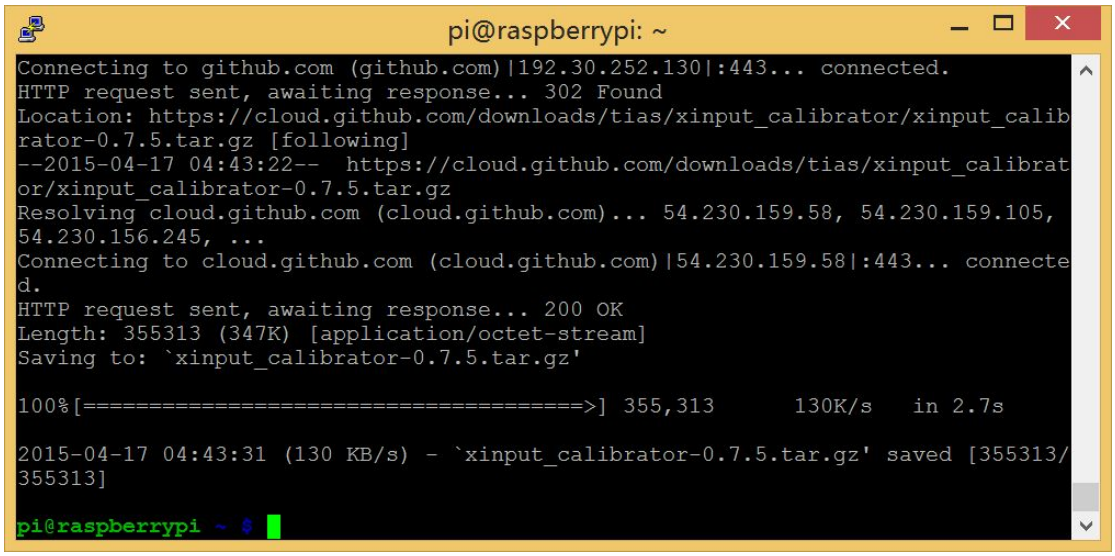


18) Then you can use tslib library[https://github.com/kergoth/tslib], calibration can also use other methods, we recommend xinput-calibrator, if used xinput-calibrator you need to install libx11-dev libxext-dev libxi-dev x11proto-input-dev, Ubuntu systems can be installed directly xinput -calibrator.

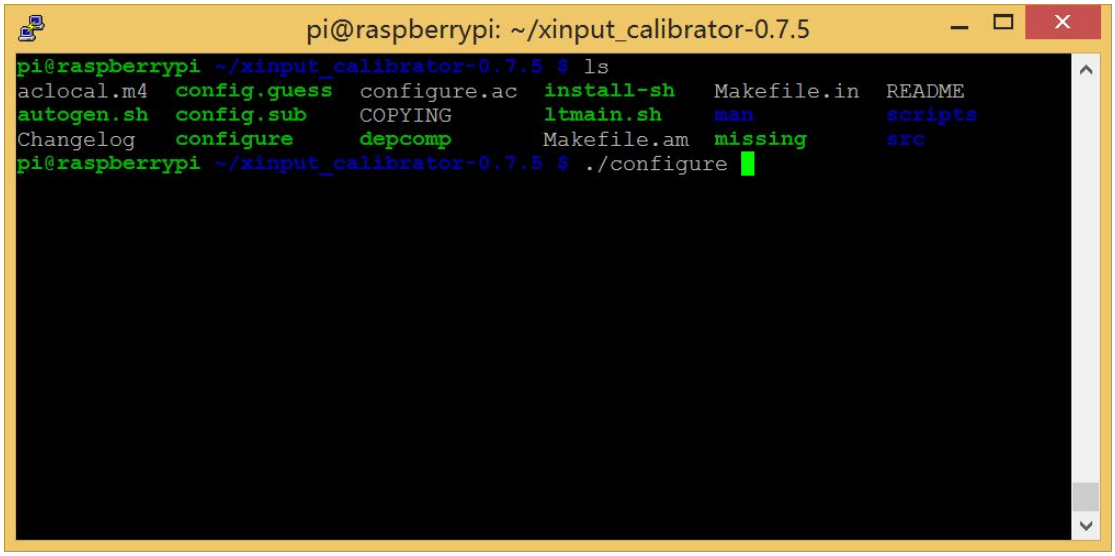


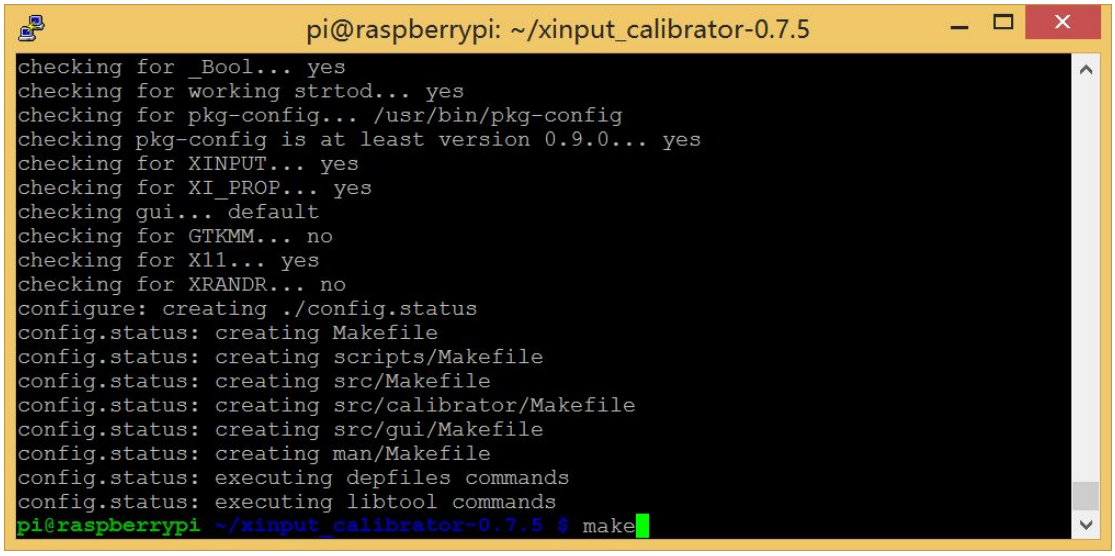


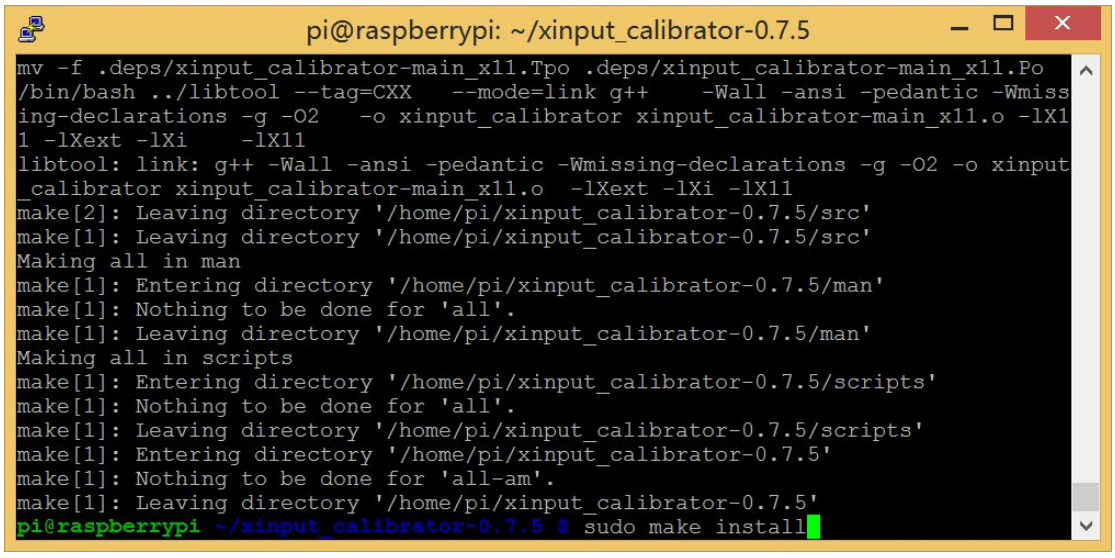
19) Calibration software is provided by https://github.com/tias/xinput_calibrator, the latest version of Download is http://github.com/downloads/tias/xinput_calibrator/xinput_calibrator-0.7.5.tar.gz.



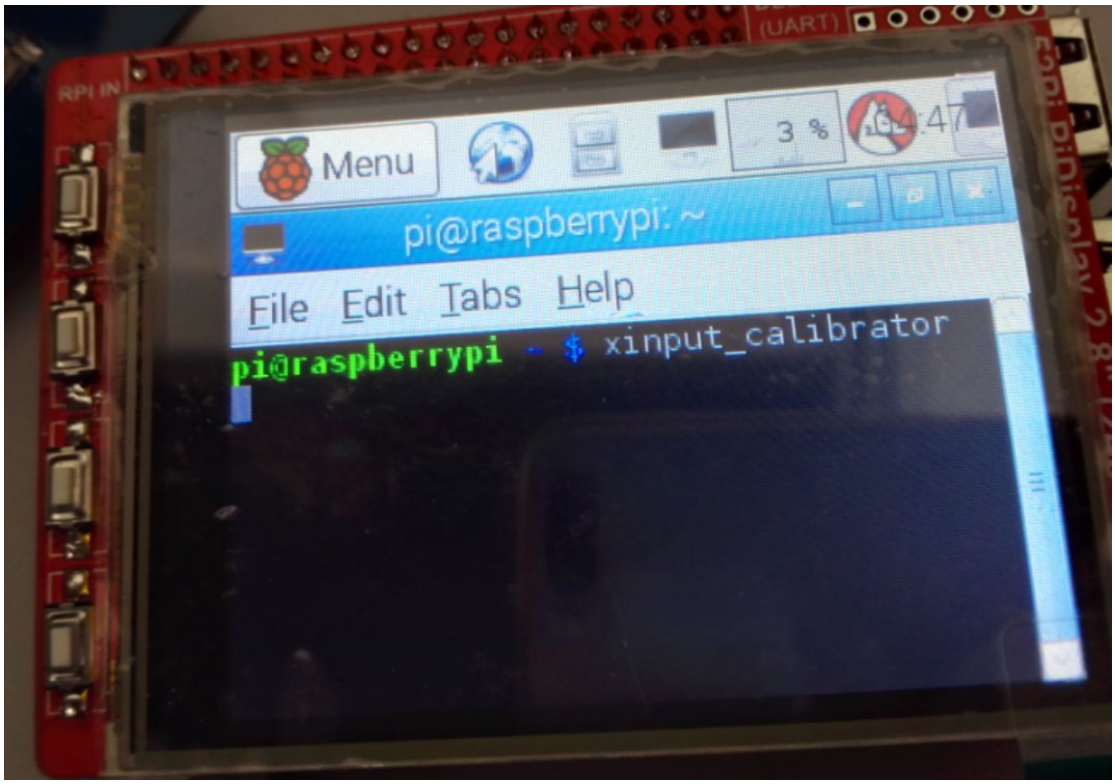
20) Installation:



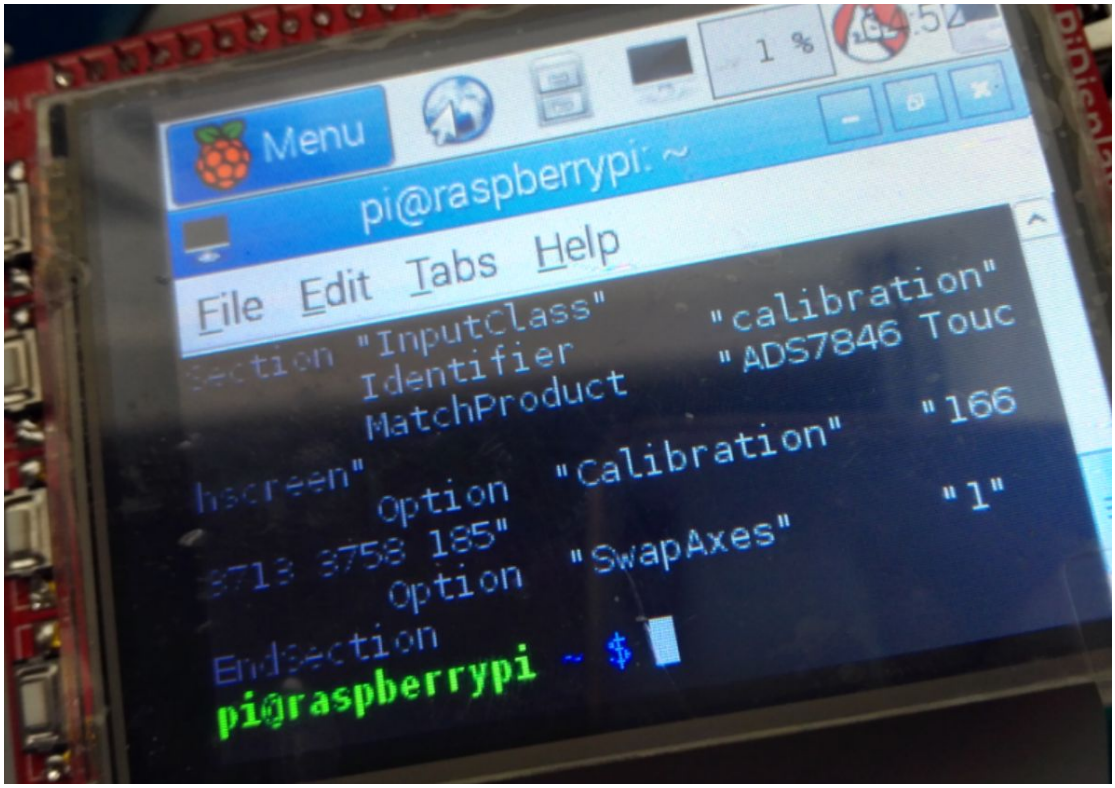




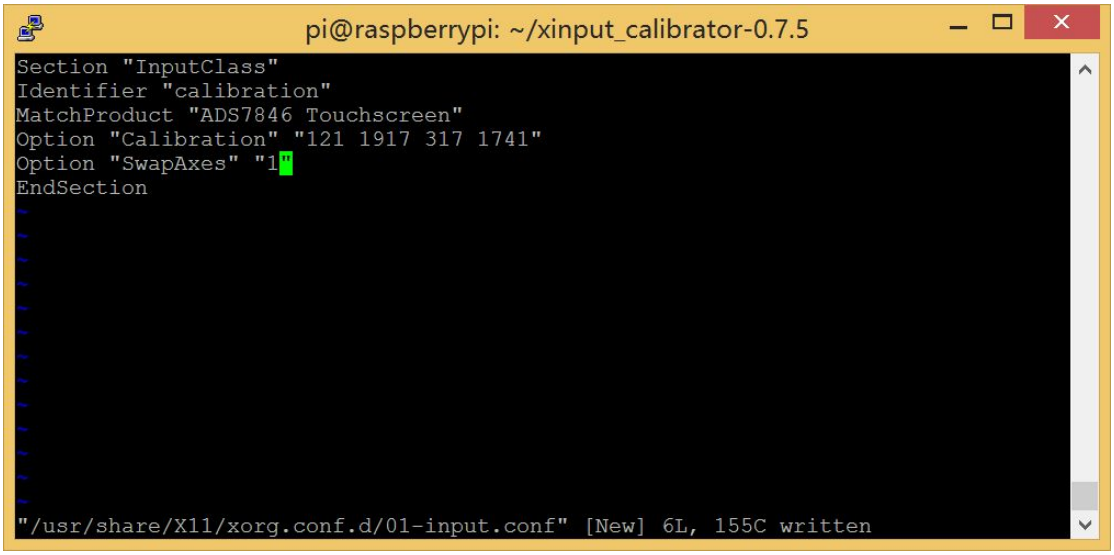
21) Then calibrate the screen, you need to enter this command on the TFT:



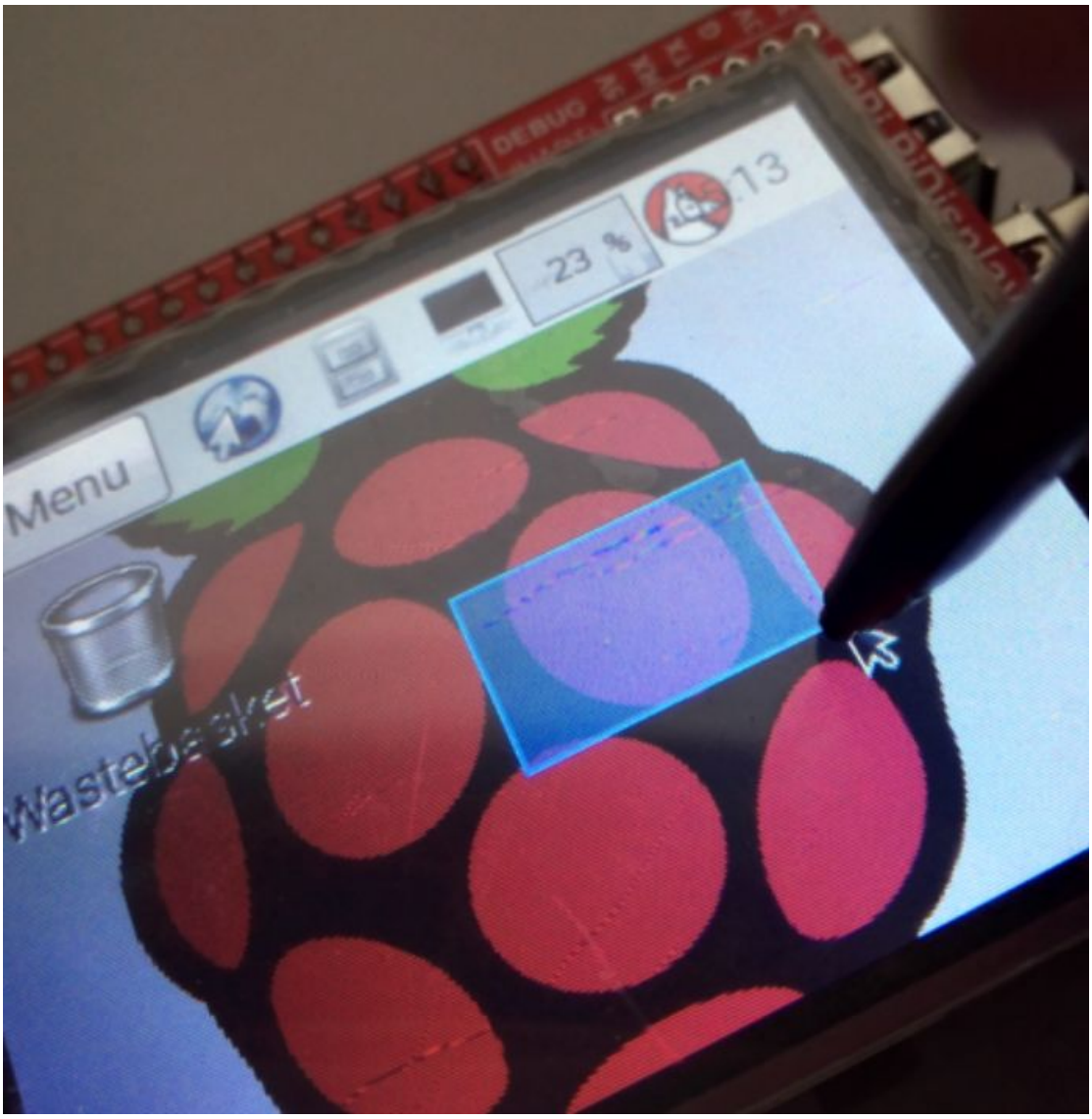
22) Calibration results obtained!



23) Writes the calibration result /usr/share/X11/xorg.conf.d/01-input.conf:

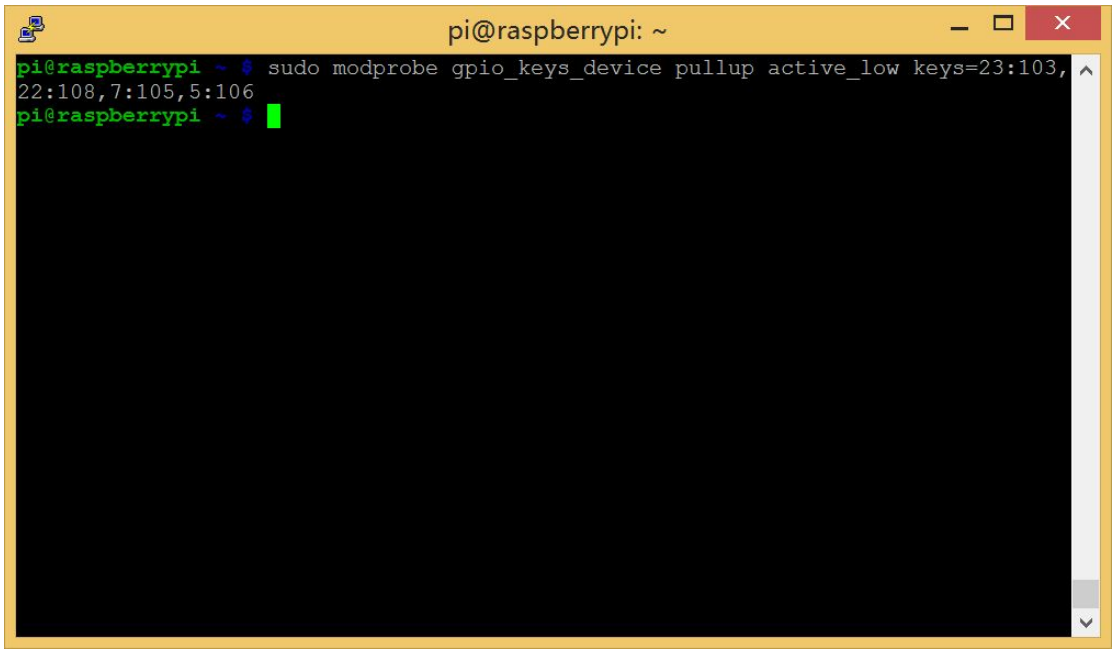


24) Restart your Raspberry Pi, you can use the pi:

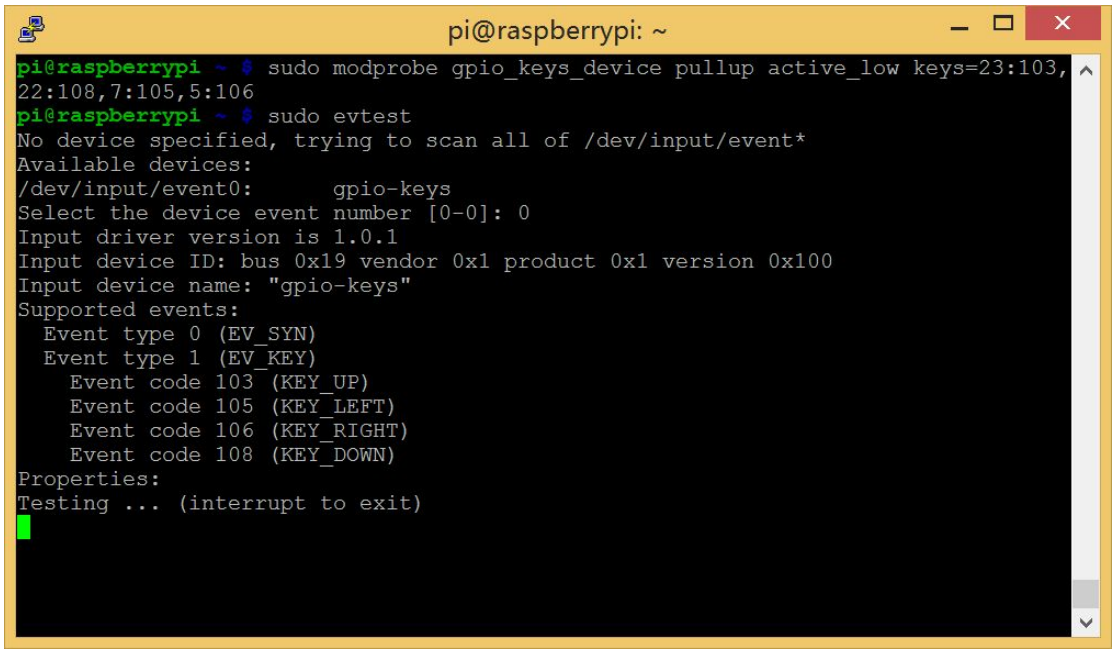


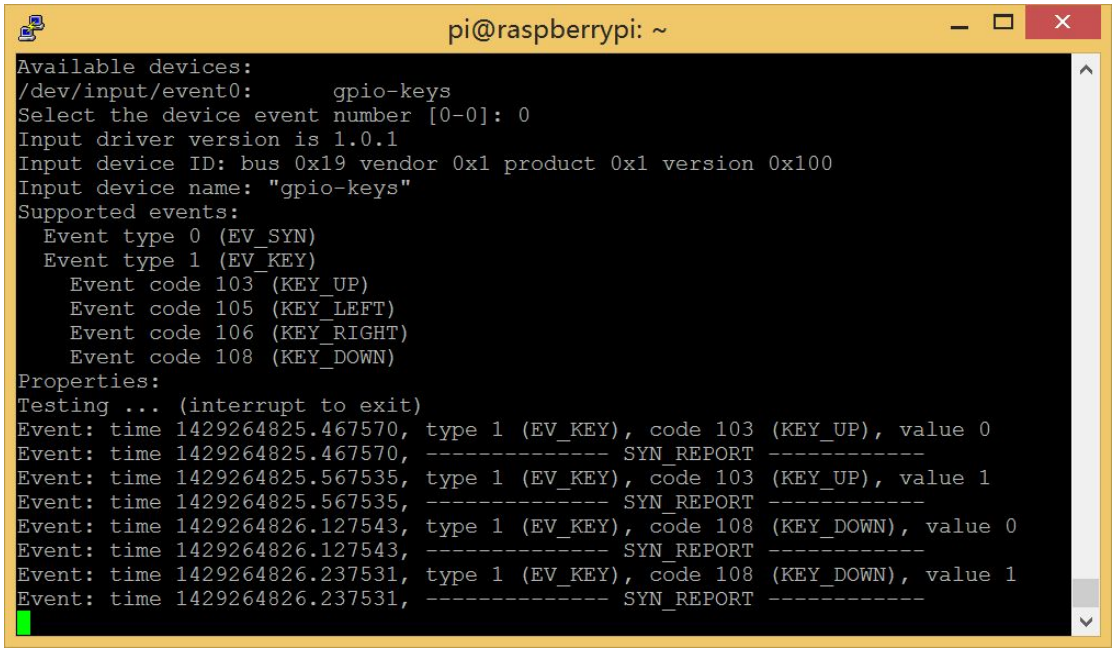
25) Then you can continue to configure the buttons on the screen button [from the top]

respectively control K1 control GPIO23, K2 control GPIO22, K3 control GPIO7, K4 control GPIO5, such as configuration means K1 = KEY_UP [103], K2 = KEY_DOWN [108], K3 = KEY_LEFT [105], K4 = KEY_RIGHT [106]:



26) Followed by the key test[by use evtest]:




pi@raspberrypi: ~
Available devices:
/dev/input/event0:   gpio-keys
Select the device event number [0-0]: 0
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x1 product 0x1 version 0x100
Input device name: "gpio-keys"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 103 (KEY_UP)
    Event code 105 (KEY_LEFT)
    Event code 106 (KEY_RIGHT)
    Event code 108 (KEY_DOWN)
Properties:
Testing ... (interrupt to exit)
Event: time 1429264825.467570, type 1 (EV_KEY), code 103 (KEY_UP), value 0
Event: time 1429264825.467570, -------------- SYN_REPORT ------------
Event: time 1429264825.567535, type 1 (EV_KEY), code 103 (KEY_UP), value 1
Event: time 1429264825.567535, -------------- SYN_REPORT ------------
Event: time 1429264826.127543, type 1 (EV_KEY), code 108 (KEY_DOWN), value 0
Event: time 1429264826.127543, -------------- SYN_REPORT ------------
Event: time 1429264826.237531, type 1 (EV_KEY), code 108 (KEY_DOWN), value 1
Event: time 1429264826.237531, -------------- SYN_REPORT ------------