

纪录一下keychron键盘 F1-F12功能键不能用的原因

结论：

keychron在插入盒子后与盒子（安卓系统）握手，获取的信号是host是一个安卓系统，因此键位变成安卓系统的了

排查思路

1、插在windows电脑上获取的键位是对的

F10-F12的键位是

```
NATIVE_KEY_RELEASED,keyCode=28,keyText=Enter,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=13
NATIVE_KEY_PRESSED,keyCode=68,keyText=F10,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=121
NATIVE_KEY_RELEASED,keyCode=68,keyText=F10,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=121
NATIVE_KEY_PRESSED,keyCode=87,keyText=F11,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=122
NATIVE_KEY_RELEASED,keyCode=87,keyText=F11,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=122
NATIVE_KEY_PRESSED,keyCode=88,keyText=F12,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=123
NATIVE_KEY_RELEASED,keyCode=88,keyText=F12,keyChar=Undefined,keyLocation=KEY_LOCATION_STANDARD,rawCode=123
```

这几个rawcode是对的。

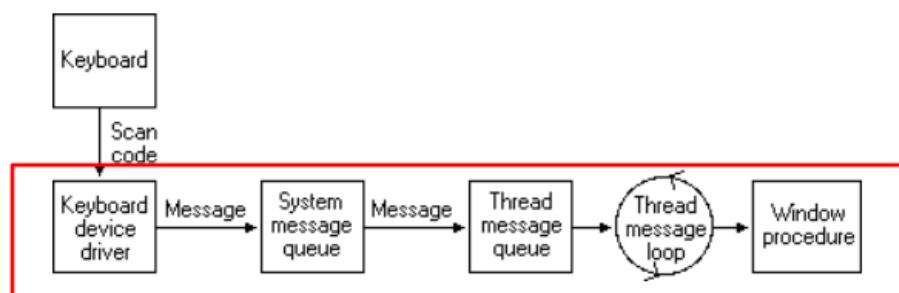
2、插在公司机器上的时候不正确

173、174、175 在windows的定义确实是 音量键，所以也就造成了看到的现象。按fx功能键，结果变成了调音量

keycode给的不对，估计是驱动的问题

这边是微软官方给的一个图，原文在这里<https://learn.microsoft.com/zh-cn/windows/win32/inputdev/about-keyboard-input>

- 和有关击键的其他信息，然后将该消息置于系统消息队列中。系统会从系统消息队列将其发布到相应线程的消息队列。最终，线程的消息循环会删除消息，并将其传递给窗口进行处理。下图演示了键盘输入模型。



大致意思是说，键盘上报的时候上报扫描码，然后驱动负责把扫描码转化为真正的键值码。

我们的问题就是扫描码都不对。扫描码之所以不对是因为键盘布局的问题，键盘布局默认按照安卓的布局了。

解决方案

既然keychron是用的检测握手信号的方式来决定发win的扫描码还是Android的扫描码，那我在键盘与盒子之间增加一个信号转化器就行了。

方案一：使用ardunio + usb host shield

键盘插在ardunio上，然后ardunio把信号在发给电脑。方案可行是可行，但是需要自己编码，需要自己有硬件装备（电烙铁、杜邦线等等）

方案二：买一个转接盒子

万能的淘宝也没有这东西，不过在我的寻找之下发现有那种一拖多的连接器（KVM切换器）。一般用于游戏工作室搬砖的，不过我们只需要usb 2 usb的功能就行了。后来淘宝买了一个就好使了。fx功能键正常使用。