**Memoria de adb**

## Método de conexión de computadora y Android

### Método 1: (conexión por cable, cable USB requerido)

<https://www.myfreax.com/use-adb-connect-phone-in-ubuntu/>

1. **Instalar android-tools-adb**

*sudo apt update*

*sudo apt install android-tools-adb android-tools-fastboot*

1. **Abrir *the Developer options***
2. **Iniciar el servicio adb**

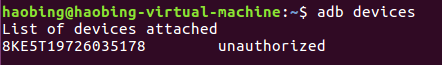
*adb start-server*

1. **Comprobar si está conectado a Android**

*adb devices*

List of devices attached

NX507J device



### Método 2: (conexión inalámbrica, la primera conexión requiere un cable USB)

1. Conectar el dispositivo Android y la computadora que ejecuta adb a la misma red de área local, como el mismo WiFi.

2. Conectar su dispositivo a su computadora con un cable USB. Asegúrese de que la conexión sea exitosa (ejecute *adb devices* para ver si el dispositivo puede aparecer en la lista).

3. Dejar que el dispositivo escuche las conexiones TCP / IP en el puerto 5555:

*adb tcpip 5555*

4. Desconectar la conexión USB.

5. Encuentrar la dirección IP del dispositivo.

6. Conectar el dispositivo por dirección IP.

*adb connect <device-ip-address>*

La <dirección del dispositivo> aquí es la dirección IP del dispositivo que se encuentra en el paso anterior.

7. Verificar el estado de la conexión.

*adb devices*

Si puedes ver

<dirección-de-dispositivo>: dispositivo 5555

La conexión es exitosa.

Si no puede conectarse, confirme que el dispositivo Android y la computadora estén conectados al mismo WiFi, y luego realice el paso adb connect <device-ip-address> nuevamente;

Si eso no funciona, reinicie adb a través de adb kill-server e intente nuevamente desde el principio.

8. Desconectar inalámbrico

*adb disconnect <device-ip-address>*

## Ingresar automáticamente una aplicación

### Método 1: Android adb shell obtiene coordenadas de posición de la pantalla táctil

<https://blog.csdn.net/liu_zhen_wei/article/details/12559277>

1. **Paso 1: Calcular la relación**

*adb shell getevent -p | grep -e "0035" -e "0036"*

Obtenga el ancho (0035) y la altura (0036) en el sistema de eventos

Tome el teléfono móvil actual que uso como ejemplo, el comando generará la siguiente información:

0035  : value 0, min 0, max *1602*, fuzz 0, flat 0, resolution 0

0036  : value 0, min 0, max *2503*, fuzz 0, flat 0, resolution 0

0035（anchura） max *1602*

0036（altura） max *2503*

*adb shell wm size*

(Resolución) es 1080p：1080（anchura） \* 1920（altura）

Calcular la relación (Ratio)：

rateW = 1080(anchura) / 1602(0035 max en evento) = *0.674*

rateH = 1920(altura) / 2503(0036 max en evento) = *0.767*

1. **Paso 2: toque la pantalla para calcular las coordenadas de la ubicación del toque**

*adb shell getevent | grep -e "0035" -e "0036"*

Tomando el teléfono móvil actual que uso como ejemplo, el comando generará la siguiente información:

/dev/input/event0: 0003 0035 00000*341*

/dev/input/event0: 0003 0036 00000*8ec*

Convierta datos de posición después de 0035 y 0036 de hexadecimal a decimal

<https://tool.oschina.net/hexconvert>

width = 0x341 = 3\*16\*16 + 4\*16 + 1 = *833*

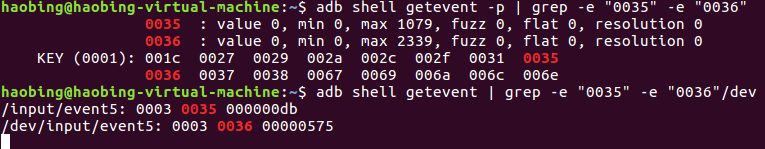
height = 0x8ec = 8\*16\*16 + 14\*16 + 12 = *2284*

Esta es la posición en el sistema de eventos. Conviértalo en una posición de pantalla.

screenW = width\*rateW = 833\*0.674 = *561*

screenH = height\*rateH = 2284\*0.767 = *1751*

Las coordenadas de la posición de la pantalla que se acaba de hacer clic son *(561, 1751）*

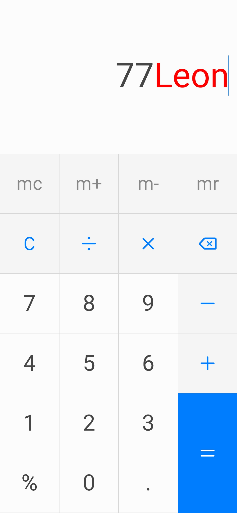


=======================================385=======================================

O utilice la herramienta de prueba automatizada para generar directamente las coordenadas de la ubicación del clic

1. **Automating Input Events on Android**

<https://www.rightpoint.com/rplabs/automating-input-events-abd-keyevent>



1. **Command:**

Introducción del adb

<https://github.com/mzlogin/awesome-adb/blob/master/README.en.md>

<https://blog.csdn.net/zhonglunshun/article/details/78362439>

Android usa el comando para simular clics y deslizar

<https://blog.csdn.net/lei7143/article/details/79088095>

Use ADB para enviar una operación táctil simulada

<https://blog.csdn.net/jgw2008/article/details/52913543>

Monkey Comentarios (artículos de guiones --- obtener coordenadas)

<https://blog.csdn.net/daihuimaozideren/article/details/77539800>

1. **Configuración horizontal y vertical**

<https://yinguobing.com/change-android-device-orientation-with-adb/>

Primero deshabilite la rotación automática:

*adb shell content insert --uri content://settings/system --bind name:s:accelerometer\_rotation --bind value:i:0*

Este comando establece la pantalla en modo horizontal:

*adb shell content insert --uri content://settings/system --bind name:s:user\_rotation --bind value:i:1*

Este comando establece la pantalla en modo vertical:

*adb shell content insert --uri content://settings/system --bind name:s:user\_rotation --bind value:i:0*

1. En pantalla vertical（0/2）

Abrir app：adb shell input tap 1116 1354

home：*adb shell input keyevent 3*

Fondo abierto：*adb shell input keyevent 187*

Presione todo: *adb shell input tap 589 1874*

Cerrar una aplicación en segundo plano：*adb shell input swipe 469 1751 900 1775*

1. En pantalla horizontal（1/3）

Abrir app：*adb shell input tap 1688 935*

Presione todo: *adb shell input tap 959 1161*

Cerrar una aplicación en segundo plano：*adb shell input swipe 500 1037 900 1037*

### Método 2: ingrese a la aplicación buscando el nombre del paquete de la aplicación (más conveniente y simple)

El método para comenzar es:

*adb shell*

*am start -n ｛**package｝/｛package｝.{activity}*

## Lanzar una aplicación automáticamente al iniciar la tablet：

### Método 1: descargar la aplicación de inicio en Google Play (aproximadamente 40 segundos)

Se llama “AutoStart”.

### Método 2: Modificar el archivo apk a través de Android Studio (en teoría, puede optimizar el tiempo de arranque, aún no se ha implementado)

1. **Paso 1: encuentre todos los paquetes (todo el software) en la tableta**

*adb shell pm list packages*

package:com.samsung.android.provider.filterprovider

package:com.monotype.android.font.rosemary

package:com.sec.android.app.DataCreate

package:com.skype.raider

package:com.android.cts.priv.ctsshim

package:com.gd.mobicore.pa

package:org.qtproject.example.touch\_screen

---------------

1. **Paso 2: descubre la ruta de este paquete:**

*adb shell pm path <package>*

**adb shell pm path** **org.qtproject.example.touch\_screen**

package:/data/app/org.qtproject.example.touch\_screen-oTNTK0fNdJxKGbZmQ\_t\_qg==/base.apk

1. **Paso 3: exporta este paquete:**

*adb pull /data/app/org.qtproject.example.touch\_screen-oTNTK0fNdJxKGbZmQ\_t\_qg==/base.apk ~/apks*

1. **Paso 4: Comience a modificar el archivo manifest.xml**
2. **Agregue el manifiesto del archivo de configuración de manifiesto (lo que significa que el programa está instalado en el almacenamiento del teléfono en lugar de la tarjeta SD)**

android:installLocation="internalOnly"

1. **Registrar una transmisión de arranque**

</receiver>-->

<receiver

android:name="com.example.bdmobile.BootBroadcastReceiver"

android:enabled="true"

android:exported="true"

\*\*android:permission="android.permission.RECEIVE\_BOOT\_COMPLETED\*\*"

>

<intent-filter>

\*\*<action android:name="android.intent.action.BOOT\_COMPLETED" />

<category android:name="android.intent.category.DEFAULT" />

<category android:name="android.intent.category.HOME"/>\*\*

</intent-filter>

</receiver>

1. **Encendido para recibir transmisión de arranque**

<uses-permission android:name="android.permission.RECEIVE\_BOOT\_COMPLETED" />

1. **Cree un nuevo receptor de transmisión y opere después de recibir la transmisión**

public class BootBroadcastReceiver extends BroadcastReceiver {

Context mContext;

@Override

public void onReceive(Context context, Intent intent) {

mContext=context;

//if (intent.getAction().equals(action\_boot) || intent.getAction().equals(action\_packageadd)|| intent.getAction().equals(action\_packageremove) || intent.getAction().equals(action\_packagereplace))

if (intent.getAction().equals("android.intent.action.BOOT\_COMPLETED")) {

Log.i("123456","android.intent.action.BOOT\_COMPLETED");

Intent ootStartIntent=new Intent(context,MainActivity.class);

/\* ootStartIntent.setAction("android.intent.action.MAIN");

ootStartIntent.addCategory("android.intent.category.LAUNCHER");\*/

\*\*ootStartIntent.addFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);\*\*

context.startActivity(ootStartIntent);

}

}

}