

# 芯片驱动指导文档(高通平台)

芯片驱动指导文档(高通平台)	
Project name	Touch panel
Document ref	[Document ref]
Version	1.3
Release date	2017.03.06
Owner	Driver Team, FocalTech
Classification	
Distribution List	
Approval	

**This document contains information proprietary to FocalTech Systems, Ltd., and may not be reproduced, disclosed or used in whole or part without the express written permission of FocalTech Systems, Ltd.**

**Copyright © 2017, FocalTech Systems, Ltd  
All rights reserved**

**3/F,Kingdom Sci-Tech Building,  
5th Gaoxinnan Avenue, Hi-Tech Park,  
Nanshan District ,Shenzhen, Gungdong, P.R. China**

**ZIP :518057  
T +86 755 26588222  
F +86 755 26712499  
E [support@focaltech- electronics.com](mailto:support@focaltech-electronics.com)**

**[www.focaltech- electronics.com](http://www.focaltech-electronics.com)**

## Revision History

Date	Version	List of changes	Author	Approved by
2017.03.06	1.3	1. Add test step of “Factory Test” 2. Modify upgrade configuration and description	xiaoligen	
2016.12.29	1.2	1. More IC Support	xiaoligen	
2016.10.31	1.1	1. More IC support 2. Extern mode, gesture update	xiaoligen	
2016.08.30	1.0	1. Initial draft.	xiaoligen	

## 目录

1	概述.....	4
2	基本信息.....	4
3	驱动文件介绍.....	5
4	整合到高通平台.....	5
4.1	移植文件.....	5
4.2	如何编译.....	6
5	驱动功能配置.....	8
5.1	DTS 配置.....	8
5.2	功能模块配置.....	9
5.3	升级设置.....	10
5.4	量产测试设置.....	11
6	ADB 节点接口.....	14
6.1	查找节点.....	14
6.2	功能节点.....	15
6.3	调试节点.....	15

## 1 概述

本文档介绍敦泰科技（FocalTech）Android 驱动程序主要功能、文件结构以及如何移植到高通平台。

**注：本驱动由 Generator.exe 生成!**

## 2 基本信息

基本信息	
支持芯片系列	FT8716、FT8736、FT8006M、FT8201、FT8606、FT8607、FTE716、FT5416、FT5426、FT5435、FT5436、FT5526、FT5526I、FT5446、FT5346、FT5446I、FT5346I、FT7661、FT7511、FT7421、FT7681、FT3C47U、FT3417、FT3517、FT3327、FT3427、FT5626、FT5726、FT5826B、FT5826S、FT7811、FT3D47、FT3617、FT3717、FT3817B、FT6236U、FT6336G、FT6336U、FT6436U、FT3267、FT3367
支持平台	Qcom 所有平台
APK/ADB 工具	支持
其他功能	GESTURE、ESD

### 3 驱动文件介绍

驱动文件存放在 focaltech\_touch 文件夹里，实现了驱动挂载、触摸点上报、休眠唤醒、手势唤醒、FW 升级等功能及 APK 和 ADB 调试调用的接口等功能。下面列表是每一个文件的功能简介：

文件名	属性	功能
Makefile	必选	Makefile 文件
Kconfig	必选	Kconfig 文件
focaltech_core.c	必选	驱动主功能文件，用来实现驱动的挂载、读取触摸数据的上报坐标、休眠唤醒处理等功能
focaltech_core.h	必选	驱动主功能头文件
focaltech_i2c.c	必选	驱动 I2C 通信功能文件
focaltech_flash.c	必选	固件升级功能文件
focaltech_flash.h	必选	固件升级头文件
focaltech_esdcheck.c	必选	ESD 检测功能文件
focaltech_config.h	必选	可配置功能模块的 Enable/disable 头文件
focaltech_common.h	必选	平台通用的定义和声明
focaltech_gesture.c	可选	手势唤醒功能文件
focaltech_ex_mode.c	可选	手套、皮套、充电器等特殊功能支持
focaltech_ex_fun.c	可选	扩展功能实现文件，主要用于支持ADB, APK 调试。该文件不是必需的，但是推荐在驱动中增加该功能
focaltech_sensor.c	可选	Sensor功能文件
focaltech_test	可选	量产测试功能文件夹
focaltech_point_report_check.c	可选	检测无中断后一段时间后全报UP事件
include		
firmware/FT8716_app_sample.i	必选	固件升级所需 app 文件(默认为空, 需进行替换)
pramboot	可选	如果升级需要使用到pramboot文件，则会包含相关文件

## 4 整合到高通平台

### 4.1 移植文件

将驱动文件打包到 focaltech\_touch 文件夹，并将 focaltech\_touch 文件夹复制到 kernel

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO FOCALTECH SYSTEMS, LTD., AND MAY NOT BE REPRODUCED, DISCLOSED OR USED IN WHOLE OR PART WITHOUT WRITTEN PERMISSION OF FOCALTECH SYSTEMS, LTD.

/drivers/input/touchscreen 目录下

1. 修改 touchscreen 目录下的 Kconfig 文件, 在这个文件的末尾增加如下行:

```
source "drivers/input/touchscreen/focaltech_touch/Kconfig"
```

2. 修改 touchscreen 目录下的 Makefile 文件, 在这个文件的末尾增加如下行:

```
obj-$(CONFIG_TOUCHSCREEN_FOCALTECH_FTS) += focaltech_touch/
```

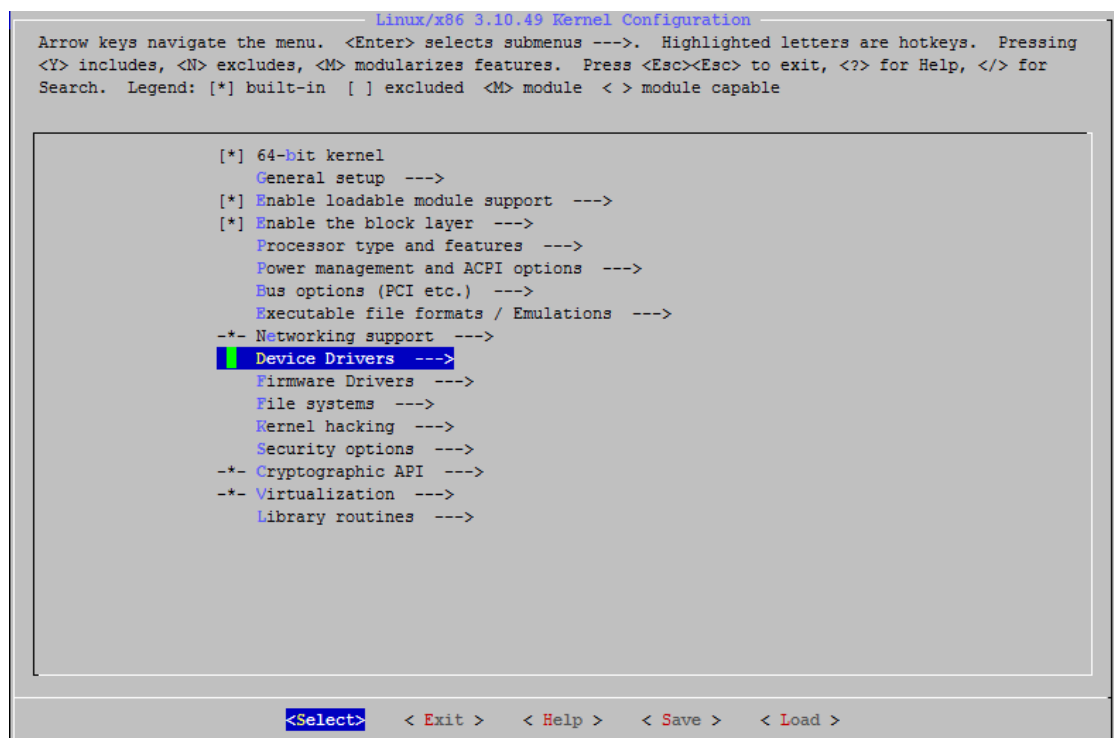
## 4.2 如何编译

- (1) 调出 menuconfig, 选择 fts;

例如:

```
$ source build/envsetup.sh
$ lunch msm8916_64-userdebug
$ cd kernel
$ make menuconfig
```

出现下图所示的 Kernel Configuration 界面, 选择 Device Drivers -> Input Device Support -> Touchscreens -> Focaltech Touchscreen



如果需要修改驱动的文件夹名(默认为 focaltech\_touch), 定义 "Focaltech Touchscreen" 以下的 "Focaltech ts directory name", 修改为您需要修改的名称.

```
[*] Focaltech Touchscreen  
(focaltech_touch) Focaltech ts directory name (NEW)
```

(2) 编译 boot.img;

```
$ make bootimage -j4
```

## 5 驱动功能配置

DTS 中对分辨率和按键等进行配置, focaltech\_config.h 中对驱动功能模块进行配置.

### 5.1 DTS 配置

示例:

高通平台dts路径: arch/arm64/boot/dts/qcom/apq8016-sbc.dtsi

```
/*
 * KEY_BACK: 158
 * KEY_MENU: 139
 * KEY_HOMEPAGE: 172
 * KEY_SEARCH: 217
 */
focaltech@38{
    compatible = "focaltech,fts"; /* do not modify */
    reg = <0x38>; /* do not modify */
    interrupt-parent = <&msm_gpio>; /* INT脚 */
    interrupts = <13 0x2>;
    focaltech,reset-gpio = <&msm_gpio 12 0x01>; /* RST脚 */
    focaltech,irq-gpio = <&msm_gpio 13 0x02>; /* INT脚 */
    focaltech,max-touch-number = <5>; /* 最大触点数 */
    focaltech,display-coords = <0 0 1080 1920>; /* 分辨率 */
    /* 按键配置 */
    /*focaltech,have-key;*/ /* 无按键 */
    /* 有按键 */
    focaltech,have-key;
    focaltech,key-number = <3>; /* 按键个数 */
    focaltech,keys = <139 172 158>; /* 按键值, 参考上面*/
    focaltech,key-y-coord = <2000>; /* 按键y坐标 */
    focaltech,key-x-coords = <200 600 800>; /* 按键对应的x坐标 */
    /* HD 720P分辨率参考设置值 */
    /*focaltech,display-coords = <0 0 720 1280>;
    focaltech,key-y-coord = <1350>;
    focaltech,key-x-coords = <180 350 540>;*/
};
```

参照focaltech-ts.txt.



## 5.2 功能模块配置

驱动中相关模块是可以通过配置 focaltech\_config.h 中相关的功能宏进行 enable(1)和 disable(0)。

在此之前, 请先选择相应的 IC 类型:

宏	功能	默认
FTS_CHIP_TYPE	当前使用的芯片类型, 根据实际使用的IC进行定义, IC列表在宏定义上方 “chip type defines”	_FT8716

接下来配置功能模块:

宏	功能	默认
FTS_DEBUG_EN	打印 debug log 信息,debug 驱动时建议打开,release 版本时建议关闭	Enable(debug), Disable(release)
FTS_MT_PROTOCOL_B_EN	Linux 多点触控协议开关, enable(B 协议), disable(A 协议)	Enable
FTS_REPORT_PRESSURE_EN	Multi-Touch A/B 上报 pressure 值, 默认打开	Enable
FTS_FORCE_TOUCH_EN	如果打开则上报实际的 pressure, 否则上报恒定的 pressure	Disable
FTS_GESTURE_EN	手势功能开关, enable(开启), disable(关闭)	Disable
FTS_ESDCHECK_EN	ESD 静电保护机制,每隔 1s 检测一次 ESD 状态, 异常则复位 IC, enable(开启), disable(关闭)	Disable
FTS_TEST_EN	量产测试, 提供整机测试功能, 用户可以通过 ADB 命令来执行测试, 以验证 TP 模组之好坏, 开启需加入 focaltech_test 文件夹	Disable
FTS_GLOVE_EN	手套模式支持	Disable
FTS_COVER_EN	皮套模式支持	Disable
FTS_CHARGER_EN	USB 充电器插入拔出检测	Disable
FTS_PSENSOR_EN	proximity 接近感应功能支持	Disable
FTS_SYSFS_NODE_EN	APK/PC 工具调试结点	Enable
FTS_APK_NODE_EN		Enable
FTS_POWER_SOURCE_CUST_EN	自定义 TP 电源控制	Disable

升级相关		
FTS_AUTO_UPGRADE_EN	自动升级开关	Enable
FTS_AUTO_UPGRADE_FOR_LCD_CFG_EN	自动升级 LCD config	Disable
FTS_AUTO_CLB_EN	自动 CB 校验	Disable
FTS_UPGRADE_LCD_CFG	升级 FLASH 中 lcd cfg 区域所使用的.i 文件(需要替	lcd_cfg.i

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO FOCALTECH SYSTEMS, LTD., AND MAY NOT BE REPRODUCED, DISCLOSED OR USED IN WHOLE OR PART WITHOUT WRITTEN PERMISSION OF FOCALTECH SYSTEMS, LTD.

	换)	
FTS_GET_VENDOR_ID_NUM	根据不同 Vendor ID 更新不同的 FW，选择不同屏 (vendor ID)的个数	0
FTS_VENDOR_1_ID	NUM >=1 时第一块屏的 Vendor ID	0x0
FTS_VENDOR_2_ID	NUM >=2 时第二块屏的 Vendor ID	0x0
FTS_VENDOR_3_ID	NUM ==3 时第三块屏的 Vendor ID	0x0
FTS_UPGRADE_FW_APP	默认自动升级 app.i 文件(需要替换,sample.i 是无效的)，当 NUM >= 1 时,则是对应 VENDOR_1 的升级 app.i 文件	Sample.i
FTS_UPGRADE_FW2_APP	对应 VENDOR_2 的升级 app.i 文件	Sample.i
FTS_UPGRADE_FW3_APP	对应 VENDOR_3 的升级 app.i 文件	Sample.i
压力测试		
FTS_UPGRADE_STRESS_TEST	压力测试，使能后，会重复升级指定的次数。驱动调好后，建议做压力测试，以检测升级功能的可靠性	Disable
FTS_UPGRADE_TEST_NUMBER	压力测试次数	1000

### 5.3 升级设置

自动升级 FTS\_AUTO\_UPGRADE\_EN 是默认开启的，移植时升级所使用到的文件按照以下进行配置：

#### FTS\_CHIP\_TYPE:

选择对应的 IC 类型

#### FTS\_GET\_VENDOR\_ID\_NUM

如果此专案不止一个屏，则需设置此宏，以区分不同的 FW 升级：

0-不判断 Vendor ID，采用默认的 FW FTS\_UPGRADE\_FW\_APP

1 -判断第一个FTS\_VENDOR\_1\_ID，采用FW FTS\_UPGRADE\_FW\_APP，如果不匹配，则不升级

2-判断FTS\_VENDOR\_1\_ID& FTS\_VENDOR\_2\_ID, 采用FW FTS\_UPGRADE\_FW\_APP& FTS\_UPGRADE\_FW2\_APP, 根据从TP读到的不同的vendor id升级不同的FW

3-如果此案兼容 3 个屏，才需要设置成 3。采用下列所有 Vendor ID 判断，不同的 Vendor ID 升级不同 FW

#### FTS\_VENDOR\_1\_ID

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO FOCALTECH SYSTEMS, LTD., AND MAY NOT BE REPRODUCED, DISCLOSED OR USED IN WHOLE OR PART WITHOUT WRITTEN PERMISSION OF FOCALTECH SYSTEMS, LTD.

TP 所对应屏厂 1 玻璃的 ID，以区别不同的屏

#### FTS\_VENDOR\_2\_ID

TP 所对应屏厂 2 玻璃的 ID，以区别不同的屏

#### FTS\_VENDOR\_3\_ID

TP 所对应屏厂 3 玻璃的 ID，以区别不同的屏

#### FTS\_UPGRADE\_FW\_APP

自动升级默认 FW，用法如下：

把 App.i 文件放置到/include/firmware 目录下

修改宏中 app.i 的名称，使其与放置到/include/firmware 目录下的文件同名

```
#define FTS_UPGRADE_FW_APP "include/firmware/FT8607_LGE_K6_V0x3D_D01_20160810_app.i"
```

当 FTS\_GET\_VENDOR\_ID\_NUM >=1 时则是对应第一块屏的 FW

#### FTS\_UPGRADE\_FW2\_APP

当 FTS\_GET\_VENDOR\_ID\_NUM >=2 时第二块屏对应的 FW

用法和 FW FTS\_UPGRADE\_FW\_APP 一样

#### FTS\_UPGRADE\_FW3\_APP

当 FTS\_GET\_VENDOR\_ID\_NUM ==3 时第三块屏对应的 FW

用法和 FW FTS\_UPGRADE\_FW\_APP 一样

## 5.4 量产测试设置

打开 focaltech\_config.h 中的宏 FTS\_TEST\_EN，如果您使用的 V1.2 以前的旧版驱动,您需要配置 IC 对应的宏。

(1) 在 focaltech\_config.h 中把测试功能的使能宏 FTS\_TEST\_EN 打开。

```
/*
 * Production test enable
 * 1: enable, 0:disable(default)
 */
#define FTS_TEST_EN 1

/*
 * Nodes for tools, please keep enable
 */
#define FTS_SYSFS_NODE_EN 1
#define FTS_APK_NODE_EN 1
```

(2) 如果您使用的驱动版小于 V1.2,您需要在 focaltech\_test\_config.h 中把红框中 FTS\_CHIP\_TEST\_TYPE 的宏改为所需要测试的 IC，例如 8716 就把宏

FTS\_CHIP\_TEST\_TYPE 改为 FT8716\_TEST, #ifdef FTS\_CHIP\_TYPE 下的内容是为兼容新版驱动, 不用修改。不用再增加具体 IC 类型, 只需确定项目所用到的具体 IC 是属于哪一系列的 IC。

```

: /*-----
: IC Type Test
: -----*/
: #define FT5X46_TEST    0x20
: #define FT6X36_TEST    0x40
: #define FT5822_TEST    0x70
: #define FT8606_TEST    0x90
: #define FT8716_TEST    0xA0
: #define FT3C47_TEST    0xB0
: #define FT8607_TEST    0xC0
: #define FT8736_TEST    0xE0
: #define FT3D47_TEST    0xF0
: #define FTE716_TEST    0x100
: #define FTE736_TEST    0x140
: #define FT8006_TEST    0x130
:
: #ifndef FTS_CHIP_TYPE
:
: #if (FTS_CHIP_TYPE == _FT8716)
: #define FTS_CHIP_TEST_TYPE    FT8716_TEST
: #elif(FTS_CHIP_TYPE == _FT8736)
: #define FTS_CHIP_TEST_TYPE    FT8736_TEST
: #elif(FTS_CHIP_TYPE == _FT8006)
: #define FTS_CHIP_TEST_TYPE    FT8006_TEST
: #elif(FTS_CHIP_TYPE == _FT8606)
: #define FTS_CHIP_TEST_TYPE    FT8606_TEST
: #elif(FTS_CHIP_TYPE == _FT8607)
: #define FTS_CHIP_TEST_TYPE    FT8607_TEST
: #elif(FTS_CHIP_TYPE == _FTE716)
: #define FTS_CHIP_TEST_TYPE    FTE716_TEST
: #elif(FTS_CHIP_TYPE == _FT3D47)
: #define FTS_CHIP_TEST_TYPE    FT3D47_TEST
: #elif(IC_SERIALS == 0x01)
: #define FTS_CHIP_TEST_TYPE    FT5822_TEST
: #elif(FTS_CHIP_TYPE == _FT3C47U)
: #define FTS_CHIP_TEST_TYPE    FT3C47_TEST
: #elif(IC_SERIALS == 0x02)
: #define FTS_CHIP_TEST_TYPE    FT5X46_TEST
: #elif((IC_SERIALS == 0x03) || (IC_SERIALS == 0x04))
: #define FTS_CHIP_TEST_TYPE    FT6X36_TEST
: #endif
:
: #else
: #define FTS_CHIP_TEST_TYPE    FT8716_TEST
: #endif
:

```

(3) 执行量产测试的步骤。

### 3.1、加载测试配置文件

把已经通过 PC 端测试软件 FT\_MultipleTest 测试通过的”Conf\_MultipleTest.ini”文件通过 ADB 命令 copy 到 sdcard 中。

如下图所以, 生成的 Conf\_MultipleTest.ini 文件放在本地 E 盘, 使用 adb 命令 copy 到 sdcard(>adb push E:\ Conf\_MultipleTest.ini /sdcard/)

```

C:\Users\luoguojin>adb push E:\Conf_MultipleTest.ini /sdcard/
12 KB/s (3674 bytes in 0.288s)

```

### 3.2、找到测试节点

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO FOCALTECH SYSTEMS, LTD., AND MAY NOT BE REPRODUCED, DISCLOSED OR USED IN WHOLE OR PART WITHOUT WRITTEN PERMISSION OF FOCALTECH SYSTEMS, LTD.

- 1) >adb root
- 2) >adb remount
- 3) >adb shell
- 4) #cd /sys/bus/i2c/devices/\*-0038, \*代表使用到的 I2C 编号, fts\_test 为测试节点。

```
C:\Users\luoguojin>adb remount
remount succeeded

C:\Users\luoguojin>adb shell
root@msm8916_64:/ # cd sys/bus/i2c/devices/6-0038
cd sys/bus/i2c/devices/6-0038
root@msm8916_64:/sys/bus/i2c/devices/6-0038 # ls
ls
driver
fts_charger_mode
fts_cover_mode
fts_driver_version
fts_dump_reg
fts_fw_update
fts_fw_version
fts_gesture_buf
fts_gesture_mode
fts_glove_mode
fts_hw_reset
fts_irq
fts_module_config
fts_rw_reg
fts_show_log
fts_test
fts_upgrade_app
```

### 3.3、执行测试命令

测试节点 fts\_test 是 sysfs 节点, 可以通过 echo 命令来执行。

输入命令 echo Conf\_MultipleTest.ini > fts\_test, 进行测试。

### 3.4、查看测试结果

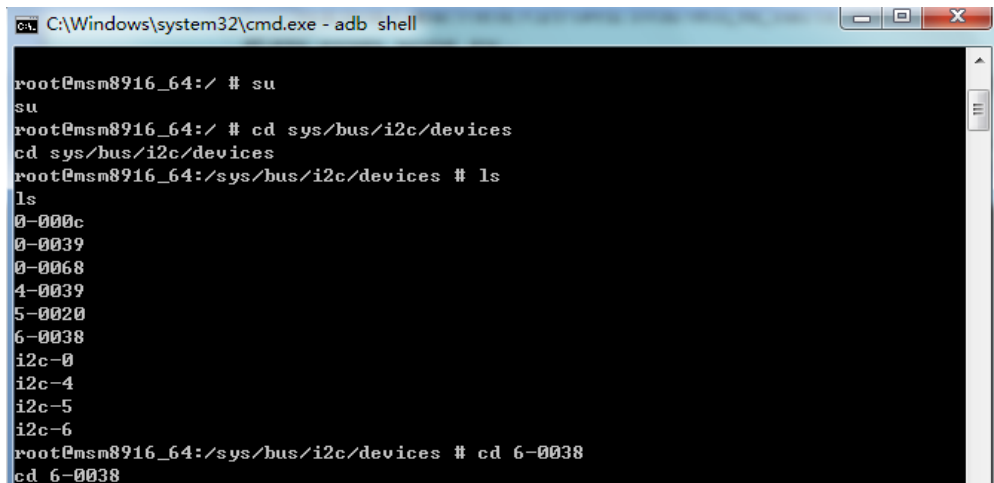
测试过程中通过串口能查看到打印的测试项目、测试数据、测试结果的 log 测试完成后会在 sdcard 中自动生成一个 testdata.csv 的测试数据文件, 一个 testresult.txt 的测试结果文件, 通过 adb 命令 copy 到电脑里, 如图, 就能查看测试结果。

```
root@msm8916_64:/sdcard # ls test*
ls test*
testdata.csv
testresult.txt
root@msm8916_64:/sdcard #
```

## 6 ADB 节点接口

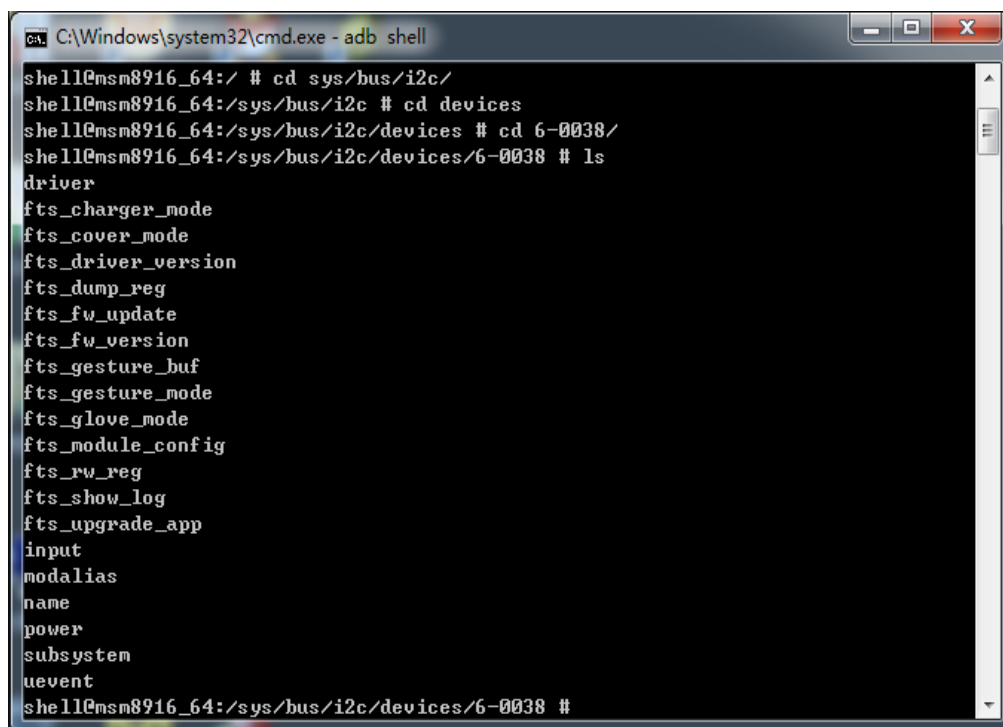
### 6.1 查找节点

生成的节点在系统的/sys/bus/i2c/devices/\*-0038/这个目录下，具体的结点要看设备挂载在哪一路I2C上。



```
C:\Windows\system32\cmd.exe - adb shell
root@msm8916_64:/ # su
su
root@msm8916_64:/ # cd sys/bus/i2c/devices
cd sys/bus/i2c/devices
root@msm8916_64:/sys/bus/i2c/devices # ls
ls
0-000c
0-0039
0-0068
4-0039
5-0020
6-0038
i2c-0
i2c-4
i2c-5
i2c-6
root@msm8916_64:/sys/bus/i2c/devices # cd 6-0038
cd 6-0038
```

查看所有节点列表



```
C:\Windows\system32\cmd.exe - adb shell
shell@msm8916_64:/ # cd sys/bus/i2c/
shell@msm8916_64:/sys/bus/i2c # cd devices
shell@msm8916_64:/sys/bus/i2c/devices # cd 6-0038/
shell@msm8916_64:/sys/bus/i2c/devices/6-0038 # ls
ls
driver
fts_charger_mode
fts_cover_mode
fts_driver_version
fts_dump_reg
fts_fw_update
fts_fw_version
fts_gesture_buf
fts_gesture_mode
fts_glove_mode
fts_module_config
fts_rw_reg
fts_show_log
fts_upgrade_app
input
modalias
name
power
subsystem
uevent
shell@msm8916_64:/sys/bus/i2c/devices/6-0038 #
```

## 6.2 功能节点

1. 手势功能节点, 需要打开宏 FTS\_GESTURE\_EN

```
#echo 0 > fts_gesture_mode //Disable Gesture功能
#echo 1 > fts_gesture_mode //Enable Gesture功能,只有向节点fts_gesture_mode写1, 手势功能才会生效
#cat fts_gesture_mode // 查看Gesture的状态
```

2. 手套功能节点, 需要打开宏 FTS\_GLOVE\_EN

```
#echo 0 > fts_glove_mode //Disable手套功能
#echo 1 > fts_glove_mode //Enable手套功能
#cat fts_glove_mode //读取手套功能状态
```

3. 皮套功能节点, 需要打开宏 FTS\_COVER\_EN

```
#echo 0 > fts_cover_mode //Disable皮套功能
#echo 1 > fts_cover_mode //Enable皮套功能
#cat fts_cover_mode //读取皮套功能状态
```

4. 充电器功能节点, 需要打开宏 FTS\_CHARGER\_EN

```
#echo 0 > fts_charger_mode //Disable充电器功能
#echo 1 > fts_charger_mode //Enable充电器功能
#cat fts_charger_mode //读取充电器功能状态
```

## 6.3 调试节点

为了方便项目的调试我们在驱动端提供了供 ADB 命令调用的接口。与生成相关调试节点有关的代码在 focaltech\_ex\_fun.c, 请务必打开 focaltech\_config.h 中的宏 FTS\_SYSFS\_NODE\_EN.

- 1) #cat fts\_driver\_version // 查看driver的版本号
- 2) #cat fts\_fw\_version // 查看firmware的版本号
- 3) #cat fts\_dump\_reg // 查看某些特征寄存器的值
- 4) #cat fts\_show\_log // 查看底层FTS\_INFO/FTS\_ERROR log状态  
#echo on > fts\_show\_log // 开启 FTS\_INFO/FTS\_ERROR  
#echo off > fts\_show\_log // 关闭 FTS\_INFO/FTS\_ERROR
- 5) #echo 00 > fts\_rw\_reg // 读取寄存器0x00的值  
#echo 0040 > fts\_rw\_reg// 写0x00为0x40  
#cat fts\_rw\_reg // 查看上一步读写操作的结果
- 6) #cat fts\_esd\_check // 查看ESD check的状态  
#echo on > fts\_esd\_check // 开启ESD  
#echo off > fts\_esd\_check // 关闭ESD
- 7) #echo \*\_app.bin > fts\_upgrade\_app // 通过.bin文件来升级FW, 必须先把\*\_app.bin放入sdcard目录下
- 8) #echo 1 > fts\_fw\_update //通过驱动中.i文件来升级FW

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO FOCALTECH SYSTEMS, LTD., AND MAY NOT BE REPRODUCED, DISCLOSED OR USED IN WHOLE OR PART WITHOUT WRITTEN PERMISSION OF FOCALTECH SYSTEMS, LTD.

9) Enable/Disable IRQ Node

```
#echo 1 > fts_irq // Enable irq
```

```
#echo 0 > fts_irq //Disable irq
```

10)硬件reset TP Node

```
#cat fts_hw_reset //Reset TP
```