



SP8810 TouchPannel 驱动介绍

课程名称： SP8810 TouchPanel 驱动介绍

课程类别： 客户培训课程

课程目标：

1. 了解TP驱动代码结构
2. 了解TP驱动工作流程
3. 熟悉TP驱动客户化配置

对象： 客户

课时数： 1H

讲师： 驱动软件工程师

教学法： 面授

主要内容



TP硬件接口介绍



TP驱动代码结构



TP驱动流程



TP驱动客户化配置



移植中常见问题及注意事项

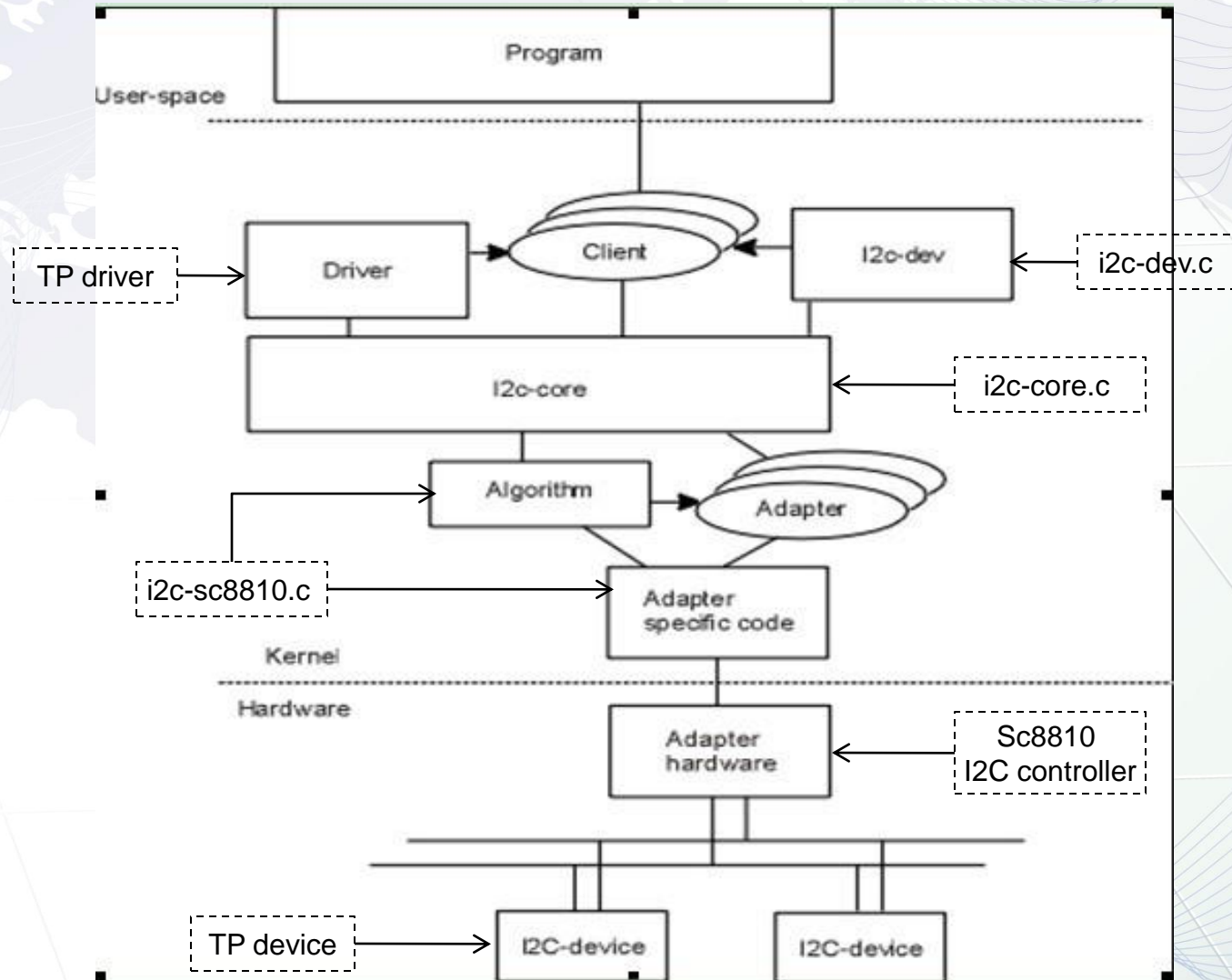
IIC接口

SP8810通常采用I2C接口与Touch Pannel通信，参考设计就是采用I2C接口的通信方式。采用I2C接口方式，通常包括四个pin脚的连接：

- 1、I2C_SDA， I2C数据线；
- 2、I2C_SCL， I2C时钟线；
- 3、CTP_RST， TP复位脚；
- 4、CTP_INT， TP中断脚；

TP硬件接口

linux内核中的I2C驱动体系构架:



主要内容



TP硬件接口介绍



TP驱动代码结构



TP驱动流程



TP驱动客户化配置



移植中常见问题及注意事项

电容式触摸屏驱动代码主要分为三个部分：驱动初始化/去初始化，命令/数据读写，数据处理。

1、驱动初始化/去初始化，主要完成TP驱动模块的初始化、去初始化、设备挂起和恢复等工作。其中初始化工作包括向I2C核心注册device (i2c_new_device) 和driver (i2c_add_driver)，input设备的申请 (input_allocate_device) 和注册 (input_register_device)，中断号申请 (request_irq)，以及对TP IC寄存器进行初始化配置等；

TP驱动代码结构

- 2、命令/数据读写，包括通过TP控制寄存器操作其IC，以及通过I2C核心读取（ `i2c_transfer` ）触摸屏ADC后的数据等。
- 3、数据处理，包括对触摸数据进行处理，并上报input驱动核心等, 其中区分触摸坐标的上传（ `input_report_abs` ）和触摸虚拟按键的上报（ `input_report_key` ）。

主要内容



TP硬件接口介绍



TP驱动代码结构



TP驱动流程



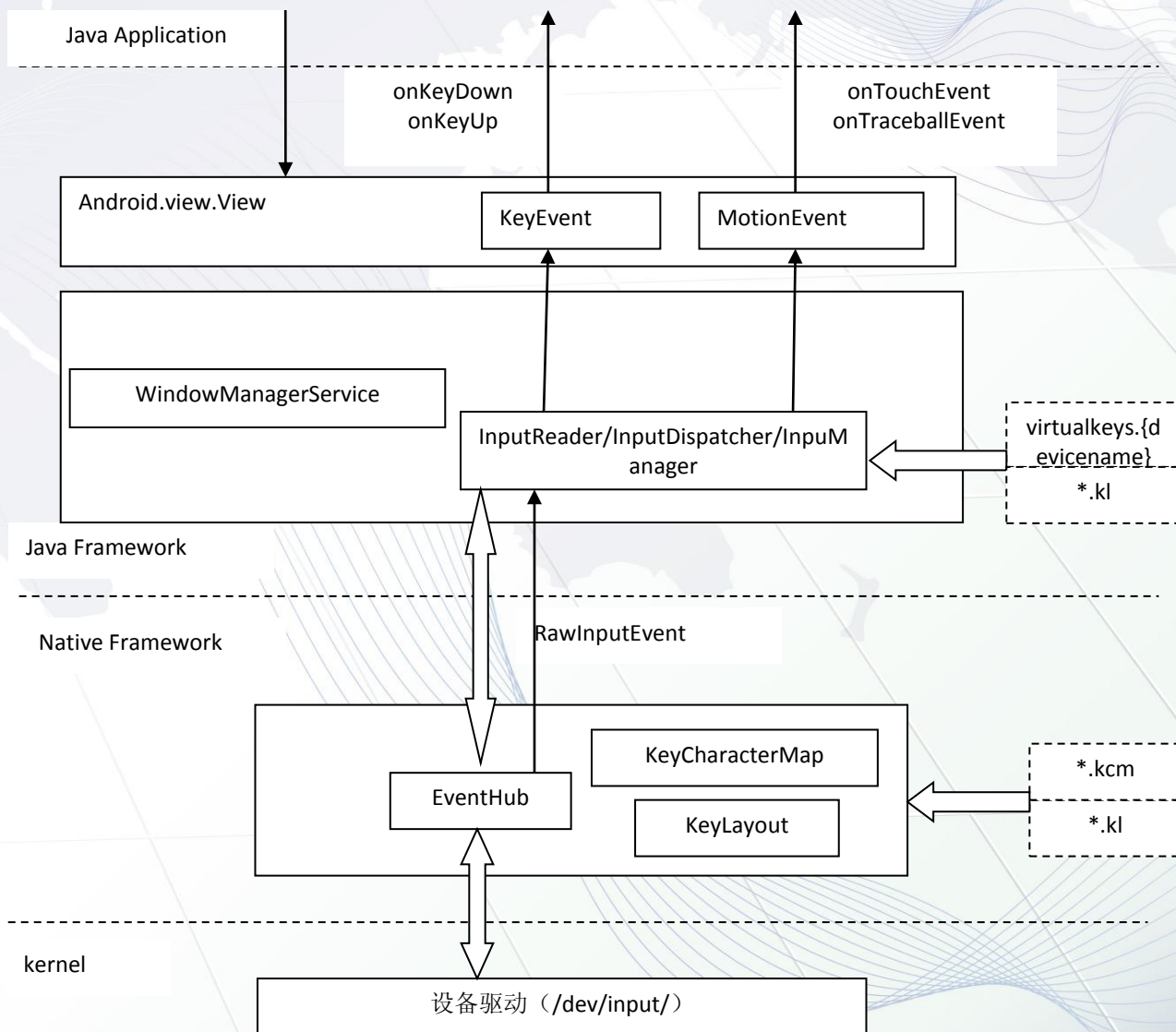
TP驱动客户化配置



移植中常见问题及注意事项

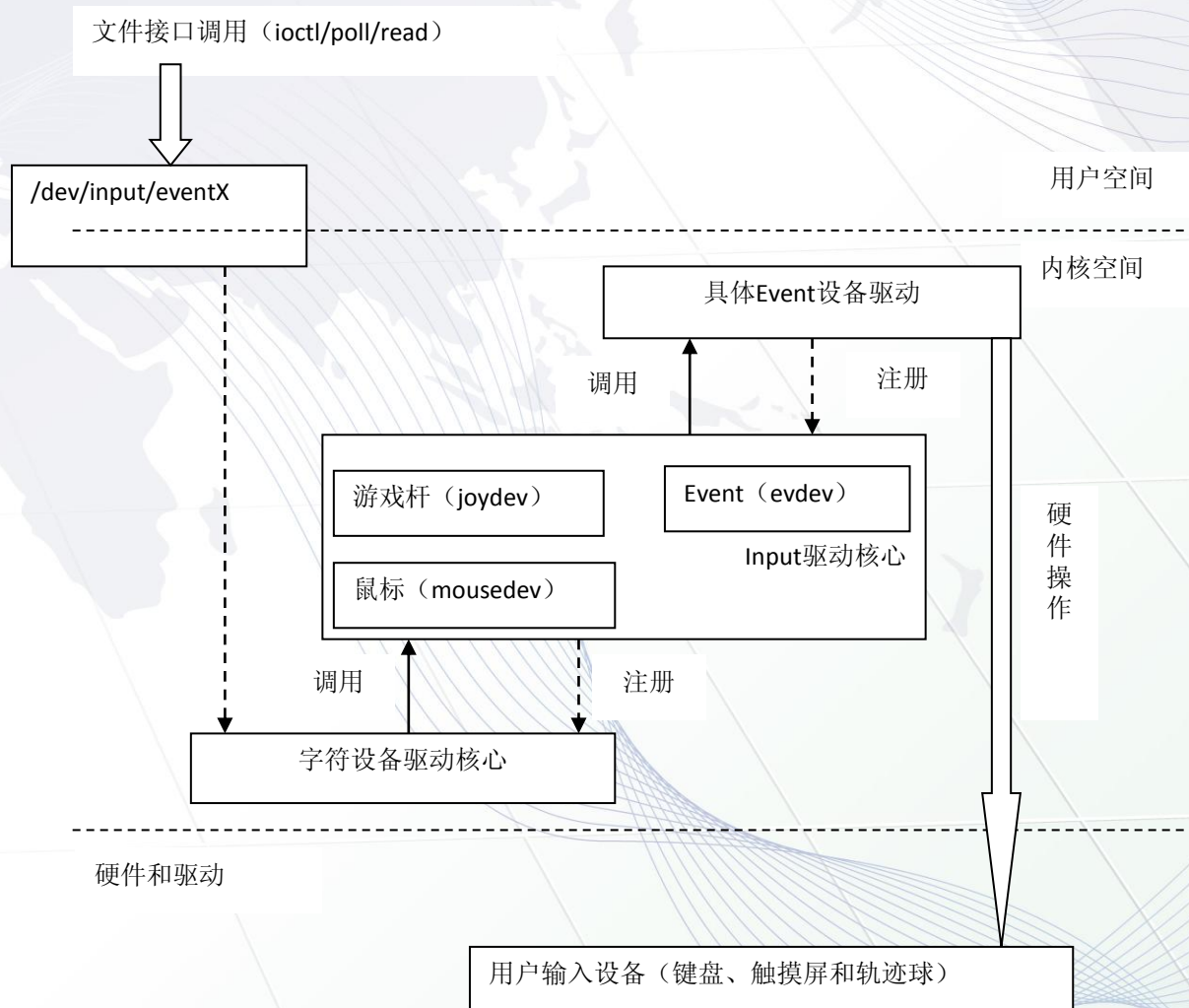
TP驱动流程

android用户输入系统构架:



TP驱动流程

linux内核设备输入系统构架：



- 1、点击触摸屏，SC8810收到TP产生的中断后，调用预先注册的中断处理函数。
- 2、中断处理函数中，读取TP触摸数据，并上传触屏坐标。若同时有多个触点，将上传多组触屏坐标数据。
- 3、若点击触屏虚拟按键区域，则上报相应按键消息（采用TP驱动程序中直接将坐标区域转换成物理按键的方式），或者上传触屏坐标（采用JAVA框架层参照虚拟按键的属性文件/sys/board_properties/virtualkeys. {devicename}转换成按键消息的方式）。

主要内容



TP硬件接口介绍



TP驱动代码结构



TP驱动流程



TP驱动客户化配置



移植中常见问题及注意事项

TP驱动客户化配置

1、在pm_func[]中与TP相关的Pinmap control register的配置，在

./customize/customer_cfg/sp8810ga/kernel/pinmap/pinmap_cfg.c中修改。

```
MFP_CFG_X(SIMCLK2, AF1, DS1, F_PULL_UP, S_PULL_UP, IO_Z), //I2C2_SDA, CTP, LDOSIM2
```

```
MFP_CFG_X(SIMDA2, AF1, DS1, F_PULL_UP, S_PULL_UP, IO_Z), // I2C2_SDL, CTP, LDOSIM2
```

```
MFP_CFG_X(SIMCLK3, AF3, DS1, F_PULL_NONE, S_PULL_NONE, IO_OE), // GPIO, CTP_RST
```

```
MFP_CFG_X(SIMRST3, AF3, DS1, F_PULL_NONE, S_PULL_NONE, IO_IE), // GPIO, CTP_INT, EXT_PULLUP
```


TP驱动客户化配置

2、在gpio_func_cfg[]中与TP相关，需要被外部引用的gpio的配置，在

./customize/customer_cfg/sp8810ga/kernel/gpio/gpio_cfg.c中添加和修改。

```
{&sprd_3rdparty_gpio_tp_rst, 59 | GPIO_DEFAULT_LOW | GPIO_DIRECTION_OUTPUT | GPIO_LOGIC_TRUE},
```

```
{&sprd_3rdparty_gpio_tp_irq, 60 | GPIO_DEFAULT_LOW | GPIO_DIRECTION_INPUT | GPIO_LOGIC_TRUE},
```

- 3、若采用静态注册I2C设备的方式，需在
./customize/customer_cfg/sp8810ga/kernel/i2c/i2c_cfg.c, 对应的i2c_boardinfoX[]中添加和修改TP的
i2c_board_info; 若采用动态注册的方式，就需要获取正确的I2C adapter, 并调用i2c_new_device(adapter, &info), 将TP关联到对应的adapter上。此步中设备的name和i2c_address, 需要设置。
- 4、调用i2c_add_driver(driver)注册TP驱动模块时，需将driver.name与第三步中注册TP设备的device.name相同。

主要内容



TP硬件接口介绍



TP驱动代码结构



TP驱动流程



TP驱动客户化配置



移植中常见问题及注意事项

移植中常见问题及注意事项

- 1、TP驱动向I2C核心注册其驱动时，其驱动名称需与注册TP设备时的设备名一致，否则驱动将不能与设备关联，导致设备驱动无法工作。
- 2、驱动初始化中，申请和注册input设备时，需要注意struct input_dev中的evbit和absbit的设置，需要将evbit的EV_SYN、EV_ABS位置1，若需要虚拟按键功能，还需要将evbit的EV_KEY位置1；只支持单点触摸的话，需要设置absbit的ABS_X, ABS_Y等位，若支持多点触摸，则需要设置absbit的ABS_MT_POSITION_X, ABS_MT_POSITION_Y, ABS_MT_TOUCH_MAJOR等位。若不正确设置这些bit，将会导致上传的事件被忽略掉。

- 3、上传触摸事件时，若为多点触摸，需在上传每一点ABS_MT_POSITION_X, ABS_MT_POSITION_Y, ABS_MT_TOUCH_MAJOR等触摸信息后，上传event code为SYN_MT_REPORT的EV_SYN事件，告诉上层一个触摸点数据结束，可以接收下一个触摸点数据。
- 4、触摸屏也需要上传press和release动作，ABS_MT_TOUCH_MAJOR代表主接触面的长轴的长度，若上传值为零的此类事件，即代表手指release动作。

- 5、触摸屏产生中断，需要去读取ADC数据，并上报触摸事件，这些操作通常放在一个单独的线程中处理，所以需要在接收到中断后，disable该中断，完成这些操作后再enable中断。数据处理工作，不要调用系统延时，应尽可能快速完成，否则可能会因为下一个中断未能及时处理，而导致触摸屏数据丢失。
- 6、建议采用用户空间加载驱动（insmod XXX）的方式调试，这样不仅可以避免由于直接编入内核，导致系统崩溃而无法抓取trace信息；而且也方便通过驱动的卸载和再加载，来快速地验证代码的修改效果。

谢谢

