# 一、多点电容触摸屏驱动框架

 1、电容触摸屏，上报多点触摸信息，通过触摸芯片，比如FT5426，这是一个IIC。多点电容触摸屏本质是IIC驱动。

 2、触摸IC一般都是有INT，当检测到触摸信息以后就会触发中断，那么就要在中断处理函数里面读取触摸点信息。

 3、得到触摸点信息，linux系统如何使用，input设备，linux系统下有触摸屏上报的流程，涉及到input子系统下触摸信息的上报。

 2、触摸协议分为：TypeA和TypeB

 TypeA：一股脑全部上报所有触摸点信息，系统去甄别这些信息属于哪个触摸点的。上报方式：

 ABS\_MT\_POSITION\_X x[0] //第一个点X轴坐标，ABS\_MT\_POSITION\_X， 使用input\_report\_abs

 ABS\_MT\_POSITION\_Y y[0] //第一个点Y轴坐标，ABS\_MT\_POSITION\_Y，

 SYN\_MT\_REPORT //点与点之间使用SYN\_MT\_REPORT隔离，使用input\_mt\_sync

 ABS\_MT\_POSITION\_X x[1] //第二个点X轴坐标，ABS\_MT\_POSITION\_X，

 ABS\_MT\_POSITION\_Y y[1]

 SYN\_MT\_REPORT

 SYN\_REPORT //所有点发送完成以后，input\_sync

 TypeB：适用于触摸芯片有硬件追踪能力的，TypeB使用slot来区分触摸点，slot使用ABS\_MT\_TRACKING\_ID来增加、删除，替换一个触摸点信息。

 ABS\_MT\_SLOT 0 //表示要上报第一个触摸点信息

 ABS\_MT\_TRACKING\_ID 45 //通过调用input\_mt\_report\_slot\_state。

 ABS\_MT\_POSITION\_X x[0] //

 ABS\_MT\_POSITION\_Y y[0]

 ABS\_MT\_SLOT 1 //第二个触摸点，使用input\_mt\_slot。

 ABS\_MT\_TRACKING\_ID 46 //input\_mt\_report\_slot\_state

 ABS\_MT\_POSITION\_X x[1]

 ABS\_MT\_POSITION\_Y y[1]

 SYN\_REPORT //所有点发送完成以后，input\_sync

 上报触摸信息是通过不同的事件来上报的：ABS\_MT\_XXX

# 二、驱动编写与测试

 1、驱动主框架是IIC设备，会用到中断，在中断处理函数里面上报触摸点信息，要用到inuput子系统框架，

 2、设备树IO修改，IIC节点添加：

 INT -> GPIO1\_IO09 GPIO

 RST -> SNVS\_TAMPER9 GPIO

 I2C\_SDA -> UART5\_RXD

 I2C\_SCL -> UART5\_RXD

 3、在I2C2节点下添加ft5426

 /\* zuozhongkai 7寸FT5426 \*/

 ft5426:ft5426@38 {

 compatible = "edt,edt-ft5426";

 pinctrl-names = "default";

 pinctrl-0 = <&pinctrl\_tsc

 &pinctrl\_tsc\_reset>;

 interrupt-parent = <&gpio1>;

 interrupts=<9 0>;

 reset-gpios = <&gpio5 9 GPIO\_ACTIVE\_LOW>;

 interrupt-gpios = <&gpio1 9 GPIO\_ACTIVE\_LOW>;

 status = "okay";

 };

 1、主题I2C框架准备好

 2、复位引脚和中断引脚，包括中断

3、初始化FT5426。

 4、input子系统框架

 5、在中断服务函数里面读取触摸坐标值，然后上报给系统

# 三··、tslib移植