GT9XX for Android 驱动移植说明书

一、驱动基本信息

支持芯片型号	GT911 GT9110 GT913 GT915 GT918 GT927 GT928 GT960 GT968 GT910 GT912 GT960F GT950 GT968F
I2C 设备地址(7 位)	0x5d、0x14
I2C 寄存器地址	16 位
APK 工具/ADB 工具	支持
自动升级	固件头文件，搜寻 bin 文件
支持 Sensor ID 数	6 个

二、驱动文件说明

一般情况下，驱动参考资料包的 *reference drivers* 文件夹下面包含以下几个文件，下面对每个文件的功能和使用方法进行说明：

1. gt9xx.c*(Required)*：驱动主功能文件，用来实现驱动的挂载、读取上报坐标、休眠唤醒处理等触摸屏驱动的基本功能。
2. gt9xx.h*(Required)*：驱动头文件，包含驱动中要用到的一些宏和常量的定义、外部变量和函数的声明等。
3. gt9xx_update.c*(Recommended)*：驱动用于支持固件升级的文件，对于触摸屏驱动来说，该文件不是必需的，但是强烈推荐在驱动中增加该功能，以便于您使用的触控 IC 在必要时升级为最新版本的固件。
4. gt9xx_firmware.h*(Recommended)*：默认存放头文件升级默认固件数组，数组默认为空。如需开启兼容 GT9XXF 模式（GTP_COMPATIBLE_MODE 置 1），您需要将 GT9XXF Firmware Headers 中相应 GT9XXF 文件夹下的 *gt9xx_firmware.h* 替换驱动中的同名文件。
5. goodix_tool.c*(Recommended)*：驱动中用于支持 *gtp_tools.apk* 工具和 ADB 工具的文件，该工具可以在装成整机后再 Android 上层对触控 IC 进行测试、调试、检测等功能，强烈推荐在驱动中增加此功能，特别是使用 COB（触控 IC 直接 layout 在主板上）模式的 TP 时，此工具能极大的方便整机上的 TP 调试。

Goodix Confidential

地址：深圳市福田保税区腾飞工业大厦B座13层　邮编：518000　电话：+86-755-3333 8828　传真：+86-755-3333 8788　Email：info@goodix.com

三、驱动移植 STEP_BY_STEP

1. **复制文件**：将 *reference driver* 文件夹中的所有文件复制到 kernel 的 *drivers/input/touchscreen/* 目录下。

2. **修改 Makefile**：在 *drivers/input/touchscreen/*目录下，打开 *Makefile* 文件，并在文件中增加以下条目（注意不同的(.o)文件之间用空格分开）：。

```
obj-y += gt9xx.o gt9xx_update.o goodix_tool.o
```

3. **添加设备**：找到 kernel 中初始化 I2C 总线的板级文件，如本驱动的开发平台 real6410 开发板是位于 *arch/arm/mach-s3c6410/ mach-smdk6410.c* 文件中，如需要将触摸屏驱动挂载 I2C0 总线上，则按以下方法添加 TP 的 i2c 设备驱动即可，0x5d 为该型号触控 IC 的 i2c 从设备地址，具体为多少需参阅该型号芯片的 datasheet，"Goodx-TS"为 i2c 设备驱动名，必须与驱动参考代码中的 GTP_I2C_NAME 保持相同。

```
static struct i2c_board_info i2c_devs0[] __initdata  =
{
    { I2C_BOARD_INFO("Goodix-TS", 0x5d),},
};
```

4. **修改参考代码**：一般情况下，移植过程中只需修改 *gt9xx.h* 文件中的内容即可，打开该头文件，按照注释中的提示移植，重点注意 TODO part 的修改即可。

（1） STEP1 替换配置信息表(REQUIRED)：将对应于您正在使用 TP 的配置信息（一般为 TP 厂提供的(*cfg 或*txt)文件里面的内容），替换 CTP_CFG_GROUP 中的内容。

```
// TODO: define your own default or for Sensor_ID == 0 config here.
#define CTP_CFG_GROUP1 { \
    0x42,0xE0,0x01,0x20,0x03,0x05,0x14,0x01,0x02,0x08,\
    // ...
}
// TODO: define your config for Sensor_ID == 1 here, if needed
#define CTP_CFG_GROUP2 {\
    }
// TODO: define your config for Sensor_ID == 2 here, if needed
#define CTP_CFG_GROUP3 {\
    }
// TODO: define your config for Sensor_ID == 3 here, if needed
```

goodix Confidential

```
    #define CTP_CFG_GROUP4 {\
        }
    // TODO: define your config for Sensor_ID == 4 here, if needed
    #define CTP_CFG_GROUP5 {\
        }
    // TODO: define your config for Sensor_ID == 5 here, if needed
    #define CTP_CFG_GROUP6 {\
        }
```

注意事项：

（1）如果没有设置 Sensor ID（详见附录），请务必将配置信息宏定义在 CTP_CFG_GROUP1，并保持其他几组为空，替换完成后，需要在每行后面增加宏定义的连接符“\”；

（2）如果实际使用的 sensor ID 数多于参考驱动中的 6 组，请参照这三组完成其他组的配置通过 sensor ID 来区分；

（3）如果配置宏的第一行设置了配置信息的写入寄存器 GTP_REG_CONFIG_DATA，则请将配置从第二行开始替换。

（2） STEP2 修改 IO 定义和 IO 操作方式**(REQUIRED)**：将 GTP_INT_PORT 和 GTP_RST_PORT 的定义修改为对应于该项目的引脚定义，另外还需检查后面几个关于 IO 操作的语句是否适用于您正在使用的平台，如果不是，则需修改成相应的操作方式。

```
//STEP_2(REQUIRED): Change I/O define & I/O operation mode.
#define GTP_INT_PORT  S3C64XX_GPN(15)
#define GTP_RST_PORT  S3C64XX_GPL(10)
#define GTP_INT_IRQ   gpio_to_irq(GTP_INT_PORT)
……
#define GTP_GPIO_AS_INPUT(pin) do{\
                        gpio_direction_input(pin);\
                        s3c_gpio_setpull(pin, S3C_GPIO_PULL_NONE);\
                         } while(0)
```

注意事项：*中断脚和复位脚应初始化为悬浮输入态。（悬浮：既不上拉，也不下拉）。*

（3） STEP3 客户自定义参数(OPTIONAL)：如果您需要自己指定分辨率、中断触发方式、支持的最多 TOUCH 数等参数，请在 ON/OFF define 中打开 GTP_CUSTOM_CFG 宏，并参照以下修改参数。

```
//**********************PART1:ON/OFF define*************************
#define GTP_CUSTOM_CFG    1
//**********************PART2:TODO define***************************
……
```


地址：深圳市福田保税区腾飞工业大厦B座13层 邮编：518000 电话：+86-755-3333 8828 传真：+86-755-3333 8788 Email：info@goodix.com

```
//STEP_3(optional):Custom set some config by custom,if need.
#if GTP_CUSTOM_CFG
  #define GTP_MAX_WIDTH    800
  #define GTP_MAX_HEIGHT   480
  #define GTP_MAX_TOUCH    5
  #define GTP_INT_TRIGGER  0
#else
  #define GTP_MAX_WIDTH    4096
  #define GTP_MAX_HEIGHT   4096
  #define GTP_MAX_TOUCH    5
  #define GTP_INT_TRIGGER  1
#endif
```

（4） STEP4 配置触摸按键(OPTIONAL)：如果您正在使用的 TP 带有触摸按键，则需要配置触摸按键，先在 ON/OFF define 中打开 GTP_HAVE_TOUCH_KEY 开关，然后再参照以下设置按键，按键的功能和顺序请在 GTP_KEY_TAB 中按需调整。

```
//**********************PART1:ON/OFF define**************************
#define GTP_HAVE_TOUCH_KEY    1
//**********************PART2:TODO define****************************
……
//STEP_4(optional):If this project have touch key,Set touch key config.
#if GTP_HAVE_TOUCH_KEY
   #define GTP_KEY_TAB {KEY_MENU, KEY_HOME, KEY_SEND}
#endif
```

（5） STEP5 增加包含文件(OPTIONAL)：在该头文件的前面增加对应您使用平台所必须的 #include 包含文件，这个步骤也是可选的，根据您编译的情况按需加入。

（6） GT9XXF 兼容说明

GT9XXF 目前的 IC 有 GT910，GT912，GT950，GT960F，GT968F。在驱动中做出了兼容处理，其宏开关为 GTP_COMPATIBLE_MODE，如果需开启此开关，请先将 GT9XXF Firmware Headers 文件夹中的对应 GT9XXF 文件夹下的 *gt9xx_firmware.h* 替换驱动中的 *gt9xx_firmware.h*。另请确保 GTP_DRIVER_SEND_CFG 开启，另请开启 GTP_ESD_PROTECT 开关。

即推荐组合：

地址：深圳市福田保税区腾飞工业大厦B座13层　邮编：518000　电话：+86-755-3333 8828　传真：+86-755-3333 8788　Email：info@goodix.com

```
#define  GTP_COMPATIBLE_MODE    1
#define  GTP_DRIVER_SEND_CFG    1
#define  GTP_ESD_PROTECT        1
```

（7） 自动升级说明

使用自动升级您需要开启宏 GTP_AUTO_UDPATE，自动升级有两种方式：

① 搜寻 BIN 文件升级：

GT9XX 预设文件路径为*/data/_goodix_update_.bin* 和 */sdcard/_goodix_udpate_.bin*，GT9XXF 为*/data/_fl_update_.bin* 和*/sdcard/_fl_update_.bin*。

② 固件数组升级：

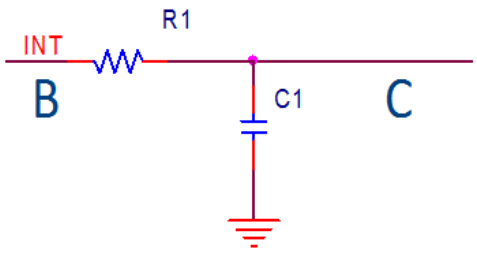
使用 *gt9xx_firmware.h* 中的固件数组 gtp_default_FW 进行升级，您需要开启 GTP_AUTO_UDPATE 与 GTP_HEADER_FW_UPDATE。此种方式 GT9XXF 不支持。

IC 类型非 GT9XXF，如需要自动升级配置，您需要同时开启宏 GTP_AUTO_UPDATE_CFG。配置自动升级路径预设为: */data/_goodix_config_.cfg* 和*/data/_goodix_config_.cfg*。使用①的方式会与 BIN 文件一齐搜寻，如找到则先进行配置升级；使用②的方式，将在升级完固件之后进行文件搜寻升级。

（8） 滑动唤醒相关说明

滑动唤醒相关联的宏为 GTP_SLIDE_WAKEUP。GT9xxS 支持此功能。此功能需添加如下电路（电路详细设计请参考 Datasheet)：

在 INT 引脚上串接 RC 电路，R：680 欧，C：680p，如下图：



B 端接 GT91X INT，C 端接 host INT，host 的 INT 上**不能**接上拉电阻。

Goodix Confidential

四、附录

1. Sensor ID：如果同一个项目中，使用几家 TP 厂的 TP，并且都使用 GOODIX 的同一款 IC，则可以对触控 IC 设置 Sensor ID，主机在初始化的时候发送相应 ID 的配置信息，从而区分不同厂家的 TP。Sensor ID 的设置方法一般是 layout 时对 IC 的某一个或者几个 IO 口进行上拉、下拉或者悬空等设置，每款芯片的设置方法有所差异，具体请参照各 IC 的 datasheet。

2. IC 固件和配置信息：固件是 IC 内部运行的程序，固件是针对一款 IC 的，而配置信息则是在固件运行的前期对固件进行初始化的一个数组，主机上电后通过 I2C 发送给 IC，IC 才能正常运行，配置信息是针对一款 TP 的，TP 的结构、工艺、通道数等大部分修改都需要通过修改配置信息来适应。

3. 配置版本号与固化配置： GT9XX 配置信息的第一个数据为配置信息版本号，只有发送的配置信息的版本号大于或等于芯片中保存的配置版本号时，发送的配置信息才会被 GT9XX 接受并生效，如果调试过程中发现配置信息发不下去，请首先读出芯片中的配置信息版本号，看是否满足要求。将 IC 配置版本配置为 0x5A(90)以上，驱动将不会发送配置，以此可达到固化配置的目的，否刚驱动将会将 IC 配置版本清为 0x41(65)。

4. SLOT 报点方式：有些 android4.0 系统上层配置必须采用 SLOT 方式报点，此时若驱动依旧采取传统的报点方式，android 上层可能会将上报的坐标识别成相对坐标，如果出现这种现象，请将 GTP_ICS_SLOT_REPORT 宏打开，将报点方式切换到 SLOT 方式即可。详细内容请参考 linux 输入子系统和 android 上层 InputReader.cpp 中关于上报事件的相关资料。

5. ESD 防护机制：是指在驱动中增加一个线程，来查询 IC 的工作状态，如果发现工作异常，则复位 IC，主要用于较强 ESD 条件下的避免 TP 失效，您可以根据 ESD 测试结果来决定是否打开该功能。注意：*该功能使用的前提是 CTP 芯片的 VDD 可由主机控制开关或主机可以通过 RESET 控制 CTP 芯片复位。*

6. 宏开关定义：驱动中 *gt9xx.h* 在 ON/OFF define 部分定义了一些宏开关，以便在调试的过程中使用，0 表示关闭该功能，1 表示打开功能，各开关的释义如下：

```
✧#define GTP_DEBUG_ON          // 调试信息开关，打开则输出调试信息
✧#define GTP_DEBUG_ARRAY_ON    // 调试数组开关，用于调试时打印一片内存的内容
✧#define GTP_DEBUG_FUNC_ON     // 调试函数开关，用于跟踪函数调用流程
✧#define GTP_CUSTOM_CFG        // 客户定制配置开关，用于客户自行修改某些参数
✧#define GTP_HAVE_TOUCH_KEY    // 触摸按键开关，仅带有触摸按键的 TP 需要打开
```

Goodix Confidential

地址：深圳市福田保税区腾飞工业大厦B座13层　邮编：518000　电话：+86-755-3333 8828　传真：+86-755-3333 8788　Email：info@goodix.com

```
✧#define GTP_AUTO_UPDATE       // 开机搜寻 bin 文件进行固件升级
✧#define GTP_HEADER_FW_UPDATE  // 使用 gt9xx_firmware.h 中的固件升级，)需开
                                 启 GTP_AUTO_UPDATE)
✧#define GTP_AUTO_UPDATE_CFG   // 搜寻.cfg 文件升级（需开启 GTP_AUTO_UPDATE)
✧#define GTP_ESD_PROTECT       // ESD 防护机制开关
✧#define GTP_ICS_SLOT_REPORT   // android4.0 以上配置成 slot 方式报点
✧#define GTP_SLIDE_WAKEUP      // 滑动唤醒功能，部分 IC 支持（如 GT915S）
✧#define GTP_DBL_CLK_WAKEUP    // 双击唤醒，需同时开启 GTP_SLIDE_WAKEUP
✧#define GTP_COMPATIBLE_MODE    // 兼容 GT9XXF 模式
```

五、 版本修订记录

文档版本	修订	日期
V1.0	初次建立	2012-08-31
V1.2	SLOT 方式	2012-10-15
V1.4	滑动唤醒，头文件升级	2013-03-11
V1.6	重新排版	2013-06-08
V1.8	GT9XXF 兼容说明，滑动唤醒说明，自动升级说明	2013-08-06

Goodix Confidential