GT9XX for all 驱动移植说明书

v2.8——Nov. 28, 2017

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

GT9XX for all 驱动移植说明书

一、驱动基本信息

支持芯片型号	GT900 系列(不支持 GT9F)
I2C 设备地址(7 位)	0x5d, 0x14
I2C 寄存器地址	16 bits
APK 工具/ADB 工具	支持
自动升级	支持
HotKnot	不支持
Gesture	支持
支持平台	MTK or QCOM
支持 sensorID 数	6

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：

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。

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。

gt9xx_update.c(Recommended)：驱动用于支持 GT9XX 固件升级功能，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。

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。

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

Kconfig(Required)：驱动编译配置文件。

Makefile(Required)。

三、 移植到目标系统

1. 复制文件

将 reference driver 目录下的 gt9xx 文件夹复制到$(KER_SRC)/drivers/input/touchscreen/目录下。

2. 修改 Makefile

在 touchscreen/目录下，打 开 Makefile 文件，添加一条编译指令如下

```
obj-$(CONFIG_TOUCHSCREEN_GT9XX)    +=    gt9xx/
```

3. 修改 Kconfig

在 touscreen/Kconfig 文件中添加如下内容：

```
source “drivers/input/touchscreen/gt9xx/Kconfig”
```

通过这种方式可以将 gt9xx 驱动的 Kconfig 包含到系统中。通过 make menuconfig 可以修改 gt9xx 的编译选项，除 TOUCHSCREEN_GT9XX 为必选项外，TOUCHSCREEN_GT9XX_UPDATE 和 TOUCHSCREEN_GT9XX_TOOL 可以根据需要选择或者取消选择。

4. 在 DTS 中添加 TP 设备信息

由于触摸屏为 I2C 设备，因此通常是将设备信息放在对应的 I2C 总线下。系统 dts 通常位于$(KER_SRC)/arch/arm64[or arm]/boot/dts 目录下。下面是将 TP 挂载到 i2c0 下面的 dts 示例：

```
&i2c0 {
      gt928-i2c@5d {
            compatible = "goodix,gt9xx";
            reg = <0x5d>;
            status = "okay";
            …
   };
};
```

详细信息参考 dtsi/gt9xx_dts.txt 文档

DTS 中各属性含义说明：

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

✧ compatible = "goodix,gt9xx";

该值用于与驱动进行匹配，如果修改了该属性，则对应的需要修改驱动中 compatible 的值。

✧ reg = <0x5d>;

设置 TP 的 I2C 地址，可以根据需要修改为 0x14。

✧ interrupt-parent = <&msm_gpio>;

该属性与 interrupts = <13 0x2800>;配合，用于设置中断号。如果配置了 irq-gpios 可以删除这两个属性。

✧ pinctrl-names = "default", "int-output-low","int-output-high", "int-input";

用于配置 pinctrl 状态，当需要使用 int 引脚来进行选址操作时，需要配置对应的 pinctrl 状态从而控制 int 引脚的输入输出状态(如果内核版本低于 3.18 的情况下可以不使用 pinctrl，同样如果使用固定 i2c 地址也可以不配置 pinctrl)。目前驱动仅支持以下几种状态，

pinctrl-0 = <&ts_int_default>; /* irq 引脚的默认状态，通常为输入 无上下拉电阻 */

pinctrl-1 = <&ts_int_output_low>; /* 控制 int 输出低 */

pinctrl-2 = <&ts_int_output_high>; /* 控制 int 输出高 */

pinctrl-3 = <&ts_int_input>; /* 控制 int 为输入 */

✧ gpio 设置

reset-gpios = <&msm_gpio 12 0x0>;

irq-gpios = <&msm_gpio 13 0x2800>;

irq-flags = <2>;

设置中断出发沿<1>为上升沿， <2>为下降沿。更多出发方式设置可以查看 $(KER_SRC)/include/linux/irq.h 中的宏定义值。

✧ 分辨率配置

touchscreen-max-id = <11>; /* 最大支持的手指数目，如果需要支持笔设备，该值必须大于等于 11，通常不用修改该值，使用默认值即可（即使固件只支持单指也不用修改） */

touchscreen-size-x = <1080>; /* x 最大值 */

touchscreen-size-y = <1920>; /* 有最大值 */

touchscreen-max-w = <512>; /* width 最大值 */

touchscreen-max-p = <512>; /* pressure 最大值 */

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

✧ touchscreen-key-map

touchscreen-key-map = <172>, <158>;

/*KEY_HOMEPAGE=172, KEY_BACK=158，KEY_MENU=139*/

如果有多个按键可以在这里写多个值（做多支持 4 个按键），如果不支持按键，直接删除该属性即可。如果配置了该属性，驱动即认为支持按键。

✧ goodix,slide-wakeup = <0>;

手势唤醒开关，<1>支持手势唤醒，<0> 或者不设置表示不支持手势唤醒功能。

✧ goodix,type-a-report = <0>;

报点协议选择，驱动默认使用 B 协议报点，如果该属性值设置为 1 则选择使用 A 协议报点。

✧ goodix,driver-send-cfg = <0>;

是否开机发送配置，<0>或者不设置表示不发送，<1>表示发送。

✧ goodix,resume-in-workqueue = <0>;

唤醒流程是否需要放到 workqueue 中，<1>在 workqueue 执行唤醒流程（可以加快亮屏速度）。

✧ goodix,int-sync = <1>;

是否需要 int 同步操作，<1>表示系统支持 int 同步（reset 选址操作同样需要设置该属性）。

✧ goodix,swap-x2y = <0>;

是否调换 x-y 轴，<1>调换，<0>或者不设置则保持默认。

✧ goodix,esd-protect = <1>;

是否开启 esd 防护，<1>开启，<0>或者不设置表示不开启。

✧ goodix,auto-update = <0>;

开机自动升级固件，<1>升级，<0> 不升级。

✧ goodix,auto-update-cfg = <0>;

固件升级开始之前，先尝试获取 goodix_config.cfg 配置文件，并将配置发送到 IC，<1>升级，<0> 不升级。

✧ goodix,power-off-sleep = <0>;

进入休眠状态是否关闭电源，<1>表示休眠时关闭电源，<0>不关闭。该状态需要配置 vdd_ana-supply 或者 vcc_i2c-supply 属性。

✧ goodix,pen-suppress-finger = <0>;

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

笔状态下是否抑制手，<1>表示笔状态下抑制手，<0>或者不设置，则不抑制手。

✧ goodix,cfg-group0 = [ ]

放置对应模组厂的配置信息。命名规范为 goodix,cfg-groupX，X 对应 sensorID 其值可以为为 0,1,2,3,4,5,6。

说明：

a) 大部分属性如果不设置或者设置为<0>均表示不使能。
b) 关于笔设备：目前驱动默认支持笔设备，并且笔和手使用同一个 input 设备进行事件上报。笔如果压力值或者 width 值为 0 则表示悬浮状态；笔按键默认上报键值为 BTN_STYLUS 和 BTN_STYLUS2。
c) 如果需要使用 int 同步，系统必须要支持将 int 引脚设置为输出，如果不支持 int 引脚直接输出(kernel 版本大于 3.18 时)，需要配置 pinctrl 从而实现 int 同步。

5. 修改参考代码

a) 添加驱动基本信息

通常情况下，移植过程中只需要修改 dts 即可，如果平台不支持 dts 则需要把原本从 dts 中解析的参数放到驱动代码中。驱动中使用的开关量，建议在 probe 函数中以下部分添加：

```
#else
        /* set parameters at here if you platform doesn't DTS */
        pdata->rst_gpio = GTP_RST_PORT;
        pdata->irq_gpio = GTP_INT_PORT;
        pdata->slide_wakeup = false;
        pdata->auto_update = true;
        pdata->auto_update_cfg = false;
        pdata->type_a_report = false;
        pdata->esd_protect = false;
        pdata->max_touch_id = GTP_MAX_TOUCH_ID;
        pdata->abs_size_x = GTP_DEFAULT_MAX_X;
        pdata->abs_size_y = GTP_DEFAULT_MAX_Y;
        pdata->max_touch_width = GTP_DEFAULT_MAX_WIDTH;
        pdata->max_touch_pressure = GTP_DEFAULT_MAX_PRESSURE;
#endif
```

同时原本放在 dts 中的配置信息，需要按以下方式放置在 gt9xx.h 中：

```
// TODO: define your own default or for Sensor_ID == 0 config here.
#define CTP_CFG_GROUP0 { \
    0x42,0xE0,0x01,0x20,0x03,0x05,0x14,0x01,0x02,0x08,\
    // ...
```


Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

```
}
// TODO: define your config for Sensor_ID == 1 here, if needed
#define CTP_CFG_GROUP1 {\
    }
// TODO: define your config for Sensor_ID == 2 here, if needed
#define CTP_CFG_GROUP2 {\
    }
// TODO: define your config for Sensor_ID == 3 here, if needed
#define CTP_CFG_GROUP3 {\
    }
// TODO: define your config for Sensor_ID == 4 here, if needed
#define CTP_CFG_GROUP4 {\
    }
// TODO: define your config for Sensor_ID == 5 here, if needed
#define CTP_CFG_GROUP5 {\
 }
```

b) 修改 I2C 单次读写操作传输的数据长度

部分 MTK 平台下，单次传输的 I2C 数据长度限制在 8bytes 或者 255bytes，因此移植过程中需要根据系统要求修改 gt9xx.c 中 I2C_MAX_TRANSFER_SIZE 对应的值，驱动默认单次传输长度设置为 255bytes。

c) 固件升级

该版本的驱动仅支持 request_firmware 方式的固件升级。驱动加载时执行自动升级，默认搜索的固件名字为 goodix_firmware.bin. 固件通常存放在以下路径中/etc/firmware/、/vender/firmware/等，不同系统支持的路径存在差异，放置固件前请与厂商确认具体支持的路径。如需替换自动搜索的固件名只要修改 gt9xx_update.c 中 GOODIX_FIRMWARE_FINE_NAME 的值即可。

d) 配置文件升级

与固件升级类似，如果打开了 auto_update_cfg 开关，驱动执行固件升级之前会首先到/etc/firmware/、/vender/firmware/等路径下搜索名为 goodix_config.cfg 的配置文件，该文件的以 ASCII 的方式保存固件内容即可，例如文件中的内容为”0x90,0x80,0x70”，配置值用 16 进制表示，各值之间采用逗号分隔。

e) 确认 kernel 是否支持 request firmware 固件升级

采用 request_firmware 方式升级固件时需要按以下步骤操作：

```
### Kernel support
```


Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

Check kernel's configuration and make sure it is properly configured to support this feature.
1. Excute the following command in the root directory of kernel:
```
$ make menuconfig
```
2. Make sure the item `<*>Userspace firmware loading support` is selected.
```
    Device Drivers -->
        Generic Driver Options -->
            <*> Userspace firmware loading support
```

f) 手势唤醒相关说明

该版本驱动不提供动态手势使能和关闭功能，如果需要使能或者关闭某一个手势功能，可以修改 gt9xx.c gtp_gesture_handler()函数，默认函数支持所有的手势，如果不需要某一手势，直接在此处移除即可。

g) 修改主动笔上报的按键值

主动笔按键上报的键值在 gt9xx.c 文件中，修改以下两个宏即可更改上报的键值：

#define GTP_PEN_BUTTON1 BTN_STYLUS
#define GTP_PEN_BUTTON2 BTN_STYLUS2

h) 驱动 log

该版驱动使用内核提供的 log 接口 dev_info\dev_err\dev_dbg。可以使用以下命令抓取 log：

cat /dev/kmsg | grep goodix-ts

默认情况 dev_dbg 的 log 是不会显示的，如果想显示 debug 信息可以在 Makefile 添加以下编译参数:

ccflags-y += -DDEBUG

i) 创建文件结点

驱动默认会创建以下结点：

/proc/gmnode //Guitar_tools.apk 调试使用

/proc/gt9xx_config //可以读写该结点获取当前 IC 的配置信息或者写入新的配置

/sys/bus/i2c/devices/4-005d/ //该目录下默认会创建以下结点：

drv_irq //支持读写，可以通过以下命令查看或者修改 irq 状态：

cat drv_irq 显示当前 irq 使能或者关闭状态

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

echo 1 > drv_irq 使能 irq

echo 0 > drv_irq 关闭 irq

reset // 仅写入，

echo 1 > reset //复位 IC

dofwupdate //仅写入

echo “goodix_firmware_file.bin” > dofwupdate //启动固件升级,依然是使用 request_firmware 的方式进行升级，需要先将固件放到指定的路径下。

productinfo //仅读取

cat productinfo //获取产品固件信息

workmode //仅读取

cat workmode //获取当前 IC 工作模式 doze, sleep or normal mode.

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 SensorID，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，保证 IC 使用的配置一致，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，否则驱动将 IC 配置版本号清为 0x41 （65）。

4. 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

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com

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。

5. I2C 通信速度：为了达到较好的用户体验效果，建议 I2C 速度设置在 300kHZ 以上。

6. CTS 标准：如果需要符合谷歌 CTS(CompatibilityTestSuite)标准，请作如下修改：将驱动中创建的文件结点权限修改为 0644

五、版本修订记录

Revision	Description	Date
V1.0	初次建立	2012-10-01
V1.2	GB 与 ICS 平台代码差异更新	2013-03-13
V1.4	1.STEP2 步骤说明 2.滑动唤醒说明 3.CTS 标准说明	2013-04-18
V1.6	1.兼容模式 GT9XXF 说明 2. GB/ICS 平台差异修正	2013-08-24
V1.8	手势唤醒修正	2014-01-14
V2.6	AndroidM 驱动	2016-07-29
V2.8	规范化驱动代码兼容 MTK 和 QCOM	2017-11-29

Address: Floor 13, Phase B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China
Tel / Fax：+86-755-33338828 Email: info@goodix.com