Driver porting guide(Qualcomm Platform)

Project name Touch panel
Document ref
Version 1.0
Release date 2018.08.06
Owner AK
Classification
Distribution List
Approval

This document contains information proprietary to HYCON Technology, Ltd., and may not be reproduced, disclosed or used in whole or part without the express written permission of HYCON Technology, Ltd..
Copyright © 2018,HYCON Technology, Ltd All rights reserved

深圳市福田區車公廟泰然六路蒼松大廈南座 1615 室
ZIP :518042 T +86 755 82720387 F +86 755 83647796
www.hycontek.com

Driver porting guide(Qualcomm Platform)

Revision History

Date	Version	List of changes	Author	Approved by
2018.08.06	1.0	Initial draft.	AK	Bruce Wang

Driver porting guide(Qualcomm Platform)

Catalogue

Driver porting guide(Qualcomm Platform)

1. Overview

This document introduces HYCON technology Android drivers main features, archives and how to develop to Qualcomm platform.

2. Basic Information

Supports chip Series: HY461x;HY462x;HY463x,
Supports Platform: Qualcomm all platforms
APK/ADB Tools: Supported

3. Drivers profile description

The drivers are stored in src folder that can achieve the drives are loaded, the touch points are reported, the sleep is awakened, the firmware upgrade and other functions and APK and the ADB adjust the interface.

Makefile taht is required file.

Hy46xx_ts.h file is required. The head file of the drivers.

HY46xx_ts.c file is required. The functions file of the drivers. It can be achieved to the drivers are loaded, report the touch coordinate, sleep mode wake, etc. As well as the firmware upgrade function, APK test function and other expansion features.

4. Integration into Qualcomm platform

4.1 Transplant file

This section describes how to port our driver. The purpose are to mount the drivers, report touch points, sleep wakeup and the firmware upgrades, and interfaces for APK and ADB debug calls.

1. Copy the hy46xx_ts.c and hy46xx_ts.h files to the kernel/include/linux/input/touchscreen/ directory.
2. Modify the Makefile file in the kernel/include/linux/input/touchscreen/ directory and add a line at the end of this file:

```
Obj-$(CONFIG_TOUCHSCREEN_HYCON) += hy46xx_ts.o
```

4.2 Compilation

After completing 4.1, you can compile the kernel. Enter the compile command at the command line:Make -j4 bootimageBoot.img is generated when the compilation is complete.

4.3 Debugging

After completing 4.2, burn boot.img to the development platform. The easiest way to verify that the driver is successful is to touch the touch screen by hand to see if the reporting function is working.To complete the feature migration, please refer to the ADB command description in 5.2 to verify by typing the relevant command at the command line.

5. Introduction to Macros and ADB Commands Related to Upper APK Applications and ADB Commands

5.1 Introduction to Macros Related to Upper APK Applications and ADB Commands
The two macros HY46XX_FW_UPDATE_ENABLE and HYS_APK_DEBUG in the hy46xx_ts.h file are related to the upper layer APK application and the ADB command call.The relevant code of HY46XX_FW_UPDATE_ENABLE is related to the online upgrade call i file. HYS_APK_DEBUG provides the driver layer interface for Hycon's read data and firmware upgrade APK and ADB command calls. The two are off by default.

5.2 Introduction to ADB Commands
In order to facilitate the debugging of the project, we provide the interface for the ADB command to be called in the driver. This can reduce the difficulty of debugging and speed up the progress of the project during the project debugging phase.
The code that generates the relevant debug node is as follows in the hy46xx_ts.c file:

```
static struct attribute *hy46xx_attributes[] = {
 &dev_attr_hystpfwver.attr,
 &dev_attr_hysfwupdate.attr,
 &dev_attr_hystprwreg.attr,
 &dev_attr_hysfwupgradeapp.attr,
 &dev_attr_hysgetprojectcode.attr,
 NULL
};

static struct attribute_group hy46xx_attribute_group = {
.attrs = hy46xx_attributes
};
int hy46xx_create_sysfs(struct i2c_client * client)
{
 int err;
 err = sysfs_create_group(&client->dev.kobj, &hy46xx_attribute_group);
    if (0 != err)
    {
    dev_err(&client->dev, ″%s() - ERROR: sysfs_create_group() failed.error code: %d\n″, __FUNCTION__, err);
    sysfs_remove_group(&client->dev.kobj, &hy46xx_attribute_group);
    return -EIO;
    }
    else
    {
    mutex_init(&g_device_mutex);
        dev_dbg(&client->dev, "hy46xx:%s() - sysfs_create_group() succeeded.
\n", __FUNCTION__);
    }
 return err;
}
int hy46xx_remove_sysfs(struct i2c_client * client)
{
 sysfs_remove_group(&client->dev.kobj, &hy46xx_attribute_group);
 mutex_destroy(&g_device_mutex);
}
```

The generated node is in the system's /sys/bus/i2c/devices/1-0038/ directory, type cmd in the run on the PC, type adb shell, type cd /sys/bus/i2c/devices /1-0038/ Enter, the following describes how to call the driver provided by the ADB command:

A. Type cat hystpfwver and press Enter to get the current firmware version number.
B. Type cat hystpdriverver and press Enter to get the version number of the current driver.
C. Type echo 1 > hysfwupdate and press Enter to upgrade firmware via .i file
D. Type echo 88 > hystprwreg to enter the value in the 0X88 register.E. Type echo *_app.bin > hysfwupgradeapp and press Enter to upgrade the firmware. You must put the *_app.bin file in the /sdcard/ directory before typing this command.

6. DTSI file description

The Qualcomm platform has a DTS folder in which there are platform-related configuration files such as the following: apq8074-dragonboard.dtsiThe contents related to TP in this file are roughly as follows:

```
  hycon@38 {
    compatible = "hycon, hy46xx_ts";
    reg = <0x38>;
    interrupt-parent = <&msm_gpio>;
    interrupts = <13 0x8>;
    vdd-supply = <&pm8916_l17>;
    vcc_i2c-supply = <&pm8916_l6>;
    hycon,name = "hys";
    hycon,family-id = <0x54>;
    hycon,reset-gpio = <&msmgpio 16 0x00>;
    hycon,irq-gpio =<&msm_gpio 13 0x00>;
    hycon,id1-gpio = <&msm_gpio 109 0x00>;
    hycon,id2-gpio = <&msm_gpio 114 0x00>;
    hycon,id3-gpio = <&msm_gpio 121 0x00>;
    hycon,display-coords = <0 0 480 854>;
    hycon,panel-coords = <0 0 1080 2000>;
    hycon,button-map= <70 150 907>;
    hycon,no-force-update;
    hycon,i2c-pull-up;
    hycon,group-id = <1>;
    hycon,hard-reset-delay-ms = <20>;
    hycon,num-max-touches = <10>;

  };
```

This file is mainly for the configuration of Qualcomm platform such as i2c address, reset, irq pin.