



HCRTOS SDK ILITEK V5912 触摸屏

1. 目录

HCRTOS SDK ILITEK V5912 触摸屏

- 1. 目录
- 2. 文档履历
- 3. 概述
 - 3.1 编写目的
 - 3.2 读者对象
- 4. 模块介绍
- 5. 模块接口说明
- 6. 模块测试用例与Sample Code
- 7. 模块调试方法
- 8. 常见问题

2. 文档履历

版本号	日期	制/修订人	制/修订记录
1.0	2023.04.17	邱浩佳	新增文档说明

3. 概述

3.1 编写目的

介绍hcartos sdk ilitek_v5912触摸屏驱动的使用

3.2 读者对象

软件开发工程师和技术支持工程师。

4. 模块介绍

- 该模块最多支持10点触摸。
- 该驱动可以直接输出坐标，只需要打开ilitek_ts.h文件中的宏ILITEK_USE_LCM_RESOLUTION。
- 该模块在SDK中的位置：**hcrtos/components/kernel/source/drivers/input/tp/ilitek_v5912**

4.1 设备树配置

在对应的板子设备树文件中，添加响应的节点

```
1 i2c@3 {
2     pinmux-active = <PINPAD_L17 8 PINPAD_L18 8>;
3     devpath = "/dev/i2c3";
4     baudrate = <400000>;
5     status = "okay";
6 };
7
8 ilitek{
9     i2c-devpath = "/dev/i2c3";        //400khz
10    i2c-addr = <0x41>;
11    //reset-gpio = <PINPAD_L24 0>;
12    irq-gpio = <PINPAD_L08 0>;
13    status = "okay";
14 };
```

4.2 menuconfig配置

根据下面路径选中ilitek驱动。

```
1 Location:
2   -> Components
3     -> kernel (BR2_PACKAGE_KERNEL [=y])
4       -> Drivers
5         -> input event (CONFIG_DRV_INPUT [=y])
6           -> tp menu (CONFIG_TP [=y])
```

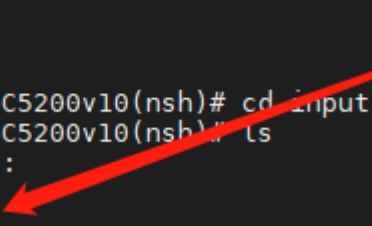
```
--- tp menu
[ ] xpt2046
[ ] hy46xx
[*] ilitek
```

配置完成后，在sdk根目录输入：make kernel-rebuild all，进行编译后烧录，在串口控制台终端输入ls命令既可以查看驱动节点。

```
pcmi2_platform
pcmi1_platform
pcmi0_platform
i2si2_platform
i2si1_platform
i2si0_platform
pcmi_platform
i2si_platform
i2so_platform
hc_rc
hc_ilitek_driver
adc
lvds
ge
llav_vdec
hc16xx_link
driver_late modules:
---none---
others modules:
usb_core
hcmmc_device
projector_auto_start
usb_storage_driver
mmc_blk
hc1600a@dbC5200v10#
```

使用时，需要open对应的input节点。这里open的是/dev/input/event1。具体操作参考Sample Code。

```
hc1600a@dbc5200v10# nsh
hc1600a@dbc5200v10(nsh)# cd dev
hc1600a@dbc5200v10(nsh)# ls
/dev:
auddec
audsink
avsync0
avsync1
bus/
dis
efuse
fb0
ge
i2c0
i2c3
input/
lvds
mmz
mtdblock0
mtdblock1
mtdblock2
mtdblock3
mtdblock4
null
persistentmem
pq
sf_prodetect
sndC0i2so
sndC0spo
sndC1spin
tv_decoder
uart0
uart_dummy
viddec
vidsink
vindvp
hc1600a@dbc5200v10(nsh)# cd input
hc1600a@dbc5200v10(nsh)# ls
/dev/input:
event0
event1
hc1600a@dbc5200v10(nsh)#
```



5. 模块接口说明

该模块没有提供额外接口。

6. 模块测试用例与Sample Code

介绍本模块相关的测试用例及相关Sample Code

```
1  #include <stdlib.h>
2
3  #include <poll.h>
4  #include <unistd.h>
5  #include <stddef.h>
6  #include <stdio.h>
7  #include <fcntl.h>
```

```

7  #include <sys/ioctl.h>
8  #include <hcuapi/input.h>
9  #include <kernel/lib/console.h>
10
11 #define BUF_SIZE 1024
12
13 static void print_help(void) {
14     printf("*****\n");
15     printf("input test cmds help\n");
16     printf("\tfor example : input_test -i1\n");
17     printf("\t'i'    1 means event1\n");
18     printf("*****\n");
19 }
20
21 static int input_test(int argc, char *argv[])
22 {
23     int fd;
24     struct input_event t;
25     struct pollfd pfd;
26     char input_buf[BUF_SIZE];
27     char *s = "/dev/input/event";
28
29     long tmp;
30     int x = 0, y = 0;
31     int event_num = -1;
32     char ch;
33     opterr = 0;
34     optind = 0;
35
36     while((ch = getopt(argc, argv, "hi:")) != EOF){
37         switch (ch) {
38             case 'h':
39                 print_help();
40                 return 0;
41             case 'i':
42                 tmp = strtoll(optarg, NULL, 10);
43                 event_num = tmp;
44                 break;
45             default:
46                 printf("Invalid parameter %c\r\n", ch);
47                 print_help();
48                 return -1;
49         }
50     }
51     if(event_num == -1)
52     {
53         print_help();
54         return -1;
55     }
56
57     sprintf(input_buf, "/dev/input/event%d", event_num);
58
59     fd = open(input_buf, O_RDONLY);
60     pfd.fd = fd;
61     pfd.events = POLLIN | POLLRDNORM;
62
63     if(fd < 0){
64         printf("can't open %s\n", input_buf);

```

```
65         return -1;
66     }
67
68     while (1) {
69         if (poll(&pfd, 1, -1) <= 0)
70             continue;
71
72         if (read(fd, &t, sizeof(t)) != sizeof(t))
73             continue;
74
75         printf("type:%d, code:%d, value:%ld\n", t.type, t.code,
t.value);
76     }
77
78     close(fd);
79
80     return 0;
81 }
82
83
84     CONSOLE_CMD(input, NULL, input_test, CONSOLE_CMD_MODE_SELF, "input test,
press power to exit test")
```

7. 模块调试方法

暂无

8. 常见问题

暂无