



# HCRTOS\_blt1680多点电容触摸屏使用 说明文档

---

## 1. 文档履历

---

版本号	日期	制/修订人	制/修订记录
1.0	2023.11.09	邱浩佳	新增说明
2.0	2023.11.10	邱浩佳	新增固件替换说明

---

### HCRTOS\_blt1680多点电容触摸屏使用说明文档

- 1. 文档履历
  - 2. 概述
    - 2.1 编写目的
    - 2.2 读者对象
  - 3. 模块介绍
    - 3.1 设备树的配置
    - 3.2 menuconfig的配置
      - 3.2.1 i2c的开启
      - 3.2.2 blt1680触摸屏的开启
      - 3.2.3 测试命令的开启
      - 3.2.4 编译
      - 3.2.5 测试命令的使用
  - 4. 模块测试用例与Sample Code
  - 5. 模块调试方法
    - 5.1 debug log的开启
    - 5.2 固件的替换
  - 6. 常见问题
- 

## 2. 概述

---

## 2.1 编写目的

介绍和指导如何在hcartos上使用blt1680的多点电容触摸屏；

## 2.2 读者对象

hcartos的开发者和FAE工程师；

## 3. 模块介绍

1. blt1680触摸屏需要加载对应的固件才可以产生中断及正常使用，代码目录下的固件并不一定适用，需要联系原厂调试合适的固件；
2. 由于该触摸芯片出厂并不内置固件，所以在第一次使用时需要一定时间加载固件进去；
3. hcartos上blt1680不支持电源管理，即不支持睡眠模式；
4. 触摸芯片支持在一定时间没有触摸会进入低功耗模式，有触摸时会自动唤醒；
5. blt1680触摸芯片最大支持10点触摸，需要固件支持；

### 3.1 设备树的配置

```
1  i2c@1{
2      pinmux-active = <PINPAD_B02 3 PINPAD_B03 3>;    //i2c所使用的引脚及复用
   功能
3      devpath = "/dev/i2c1";    //i2c节点所生产的路径
4      baudrate = <100000>;    //i2c波特率
5      mode = "master";    //i2c模式，此处为master
6      status = "okay";    //okay代表开启
7  };
8
9  betterlife_ts@2c{
10     i2c_devpath = "/dev/i2c1";    //触摸屏所使用的i2c节点
11     i2c_addr = <0x2c>;    //触摸屏的7位设备地址
12     reset_gpio = <PINPAD_L24 0>;//触摸屏复位所使用的引脚
13     irq_gpio = <PINPAD_L27 0>;    //触摸屏中断所使用的引脚
14     // vdd_name = "vdd28";    //触摸屏所使用的电源管理，目前不支持
15     // virtualkeys = <80 900 120 44 240 900 120 44 400 900 120 44>;    //实
   体触摸按键，需要硬件支持
16     TP_MAX_X = <480>;    //触摸屏X坐标分辨率
17     TP_MAX_Y = <800>;    //触摸屏Y坐标分辨率
18     status = "okay";    //okay代表开启
19  };
```

## 3.2 menuconfig的配置

### 3.2.1 i2c的开启

```
1 Symbol: CONFIG_I2C_SCB_MASTER [=y]
2 Type : bool
3 Prompt: I2C SCB Master
4 Location:
5     -> Components
6     -> kernel (BR2_PACKAGE_KERNEL [=y])
7     -> Drivers
8 (1)     -> I2C Driver Support (CONFIG_I2C [=y])
9 Defined at i2c:18
10 Depends on: BR2_PACKAGE_KERNEL [=y] && CONFIG_I2C [=y]
```

### 3.2.2 blt1680触摸屏的开启

```
1 There is no help available for this option.
2 Symbol: CONFIG_HC_BLT1680 [=y]
3 Type : bool
4 Prompt: blt1680
5 Location:
6     -> Components
7     -> kernel (BR2_PACKAGE_KERNEL [=y])
8     -> Drivers
9         -> input event (CONFIG_DRV_INPUT [=y])
10            -> tp menu (CONFIG_TP [=y])
11 Defined at tp:13
12 Depends on: BR2_PACKAGE_KERNEL [=y] && CONFIG_DRV_INPUT [=y] && CONFIG_TP [=y]
```

### 3.2.3 测试命令的开启

```
1 There is no help available for this option.
2 Symbol: CONFIG_CMDS_INPUT [=y]
3 Type : bool
4 Prompt: input event operations
5 Location:
6     -> Components
7     -> Cmds (BR2_PACKAGE_CMDS [=y])
8 Defined at source:46
9 Depends on: BR2_PACKAGE_CMDS [=y] && CONFIG_DRV_INPUT [=y]
```

### 3.2.4 编译

```
1 | make kernel-rebuild cmds-rebuild all
```

### 3.2.5 测试命令的使用

在串口控制终端输入：input -i1，这里的1代表event1；

```
1 | hc1512a@dbb200# input -i1
2 | ID:0, X:360, Y:717, W:1
3 | type:3, code:48, value:1
4 | type:3, code:50, value:1
5 | type:3, code:54, value:717
6 | type:1, code:330, value:1
7 | key 330 Pressed
```

## 4. 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 | #include <poll.h>
3 | #include <unistd.h>
4 | #include <stddef.h>
5 | #include <stdio.h>
6 | #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,
76 t.value);
77     }
78
79     close(fd);
80
81     return 0;
82 }
83
84     CONSOLE_CMD(input, NULL, input_test, CONSOLE_CMD_MODE_SELF, "input test,
press power to exit test")

```

## 5. 模块调试方法

### 5.1 debug log的开启

调试log宏BTL\_DEBUG\_SUPPORT的开启：  
components/kernel/source/drivers/input/tp/blt1680/bl\_chip\_custom.h；  
开启后会在初始化以及触摸时打印调试信息；

### 5.2 固件的替换

代码目录下的固件并不适用，需要将经过触摸ic原厂调试生成的固件替换bl\_fw.h里面的内容，再执行make kernel-rebuild all；

BLT7XX6\_BLT1680\_HQ-JR1785\_P3F4A1\_20231110-480X272.zip - ZIP 压缩文件, 解包大小为 1,057,620 字节

名称	大小	压缩后大小	类型	修改时间	CRC32
文件夹					
BLT7XX6_BLT1680_HQ-JR1785_P3F4A1_20231110.bin	65,492	33,755	BIN 文件	2023/11/10 1...	B7C3C5...
BLT7XX6_BLT1680_HQ-JR1785_P3F4A1_20231110.c	496,064	106,912	C Source	2023/11/10 1...	5523954B
BLT7XX6_BLT1680_HQ-JR1785_P3F4A1_20231110.h	496,064	106,912	C/C++ Header	2023/11/10 1...	5523954B

## 6. 常见问题

暂无；