



SIM8500 sc2721 charge bringup

芯讯通无线科技(上海)有限公司

上海市长宁区临虹路289号3号楼芯讯通总部大楼

电话: 86-21-31575100

名称:	SIM8500内置充电芯片调试
版本:	1.00
日期:	2022.03.08
状态:	已发布

版权声明

本手册包含芯讯通无线科技（上海）有限公司（简称：芯讯通）的技术信息。除非经芯讯通书面许可，任何单位和个人不得擅自摘抄、复制本手册内容的部分或全部，并不得以任何形式传播，违反者将被追究法律责任。对技术信息涉及的专利、实用新型或者外观设计等知识产权，芯讯通保留一切权利。芯讯通有权在不通知的情况下随时更新本手册的具体内容。

本手册版权属于芯讯通，任何人未经我公司书面同意进行复制、引用或者修改本手册都将承担法律责任。

芯讯通无线科技(上海)有限公司

上海市长宁区临虹路289号3号楼芯讯通总部大楼

电话：86-21-31575100

邮箱：simcom@simcom.com

官网：www.simcom.com

了解更多资料，请点击以下链接：

<http://cn.simcom.com/download/list-230-cn.html>

技术支持，请点击以下链接：

<http://cn.simcom.com/ask/index-cn.html> 或发送邮件至 support@simcom.com

版权所有 © 芯讯通无线科技(上海)有限公司 2021，保留一切权利。

关于文档

版本历史

版本	日期	作者	备注
1.00	2022.3.8	zhangyang	第一版

适用范围

本文档适用于 SIMCom SIM8500 系列内置 pm2721 charge bringup。

目录

关于文档.....	3
版本历史.....	3
适用范围.....	3
1 介绍.....	5
1.1 本文目的.....	5
1.2 参考文档.....	5
1.3 术语和缩写.....	5
2 Uboot 充电介绍.....	6
2.1 主要驱动文件：.....	6
2.2 Uboot 中主要实现的几个功能：.....	6
2.2.1 低电量检测.....	6
2.2.2 预充电.....	6
2.2.3 关机充电模式.....	7
3 Kernel 充电介绍.....	7
3.1 Charger Manager.....	7
3.2 Charger IC.....	8
4 充电配置介绍.....	9
4.1 charger-manager 节点.....	9
4.2 Charger IC 节点.....	11
5 debug 介绍.....	12
5.1Healthd log.....	12
5.2 插拔充电器 log.....	12
5.3 调试节点.....	12

1 介绍

1.1 本文目的

介绍如何调试 SIM8500 内置 pm2721 芯片充电部分。

1.2 参考文档

[1] SC2721GDeviceSpecification_V0.3.pdf

1.3 术语和缩写

PSY : 电源

DCP: 专用充电接口

SDP: 标准下行接口

CDP: 充电下行接口

CC: 恒定电流

CV: 恒定电压

2 Uboot 充电介绍

2.1 主要驱动文件:

- u-boot15/drivers/power/battery/sprd_battery.c
- u-boot15/drivers/power/battery/sprd_chg_2721.c

2.2 Uboot 中主要实现的几个功能:

1. 低电量检测
2. 预充电
3. 关机充电模式

2.2.1 低电量检测

- 不插充电器时，电池电压大于 3500mV 时，才允许进入 Kernel 阶段。否则显示当前电池电量低并直接关机。
- 插充电器时，电池电压大于 3300mV 时，才允许进入 Kernel 阶段。否则显示电池电量低且正在充电中，进入预充电阶段。

```
286 #define LOW_BAT_VOL ..... 3500 /*phone battery voltage low than this value will not boot up*/
287 #define LOW_BAT_VOL_CHG ..... 3300 //3.3V charger connect
288
289
```

在 uboot 中搜索 关键字 LOW_BAT_VOL 即可看到。

2.2.2 预充电

在预充电阶段，根据充电器的类型设定充电电流，充电电流范围为 450mA ~ 1000mA。

```

19 #define DCP_LIMIT_CUR 3000
20 #define SDP_LIMIT_CUR 500
21 #ifndef DCP_CHG_CUR
22 #define DCP_CHG_CUR 1000
23 #endif
24 #define SDP_CHG_CUR 450
25 #define FAST_CHG_VOL 3000
26

```

2.2.3 关机充电模式

```

232 boot_mode_enum_type get_mode_from_charger(void)
233 {
234     if (charger_connected()) {
235         debugf("get mode from charger\n");
236         #ifdef CONFIG_FIBOCOM_SC665
237             return CMD_NORMAL_MODE; //CMD_CHARGE_MODE;
238         #else
239             return CMD_NORMAL_MODE; //simcom disable poweroff charge, change CMD_CHARGE_MODE to CMD_NORMAL_MODE
240         #endif
241     } else {
242         return CMD_UNDEFINED_MODE;
243     }
244 }

```

上方代码段是去掉 关机充电功能的，如果想要打开 关机充电，将 239 行 return CMD_CHARGE_MODE 即可。

3 Kernel 充电介绍

Kernel 驱动分为三个部分，分别是 Charge Manger、FuelGauge、Charge IC。这三部分作为独立的设备驱动均注册到 Power-supply 中，每一个设备为单独的 PSY。（FuelGauge 指电量计，本篇文档不做介绍）PSY 之间可以通过 power supply 属性相互访问，虽然每个 PSY 都是独立的，但实际上 Fuel Gauge 跟 Charge IC 是服务于 Charge Manger，Charge Manger 不需要了解硬件细节，仅通过获取相应功能的 PSY 设备实例，通过这个 PSY 的属性获取相应信息。

3.1 Charger Manager

Charger Manager 以 “battery” 名字注册至 Power Supply 架构，能够读写 Fuel Gauge 和 Charger

IC 的 PSY 属性。

源文件： bsp/kernel/kernel4.14/drivers/power/supply/charger-manager.c

PSY 描述：

```
1985 static const struct power_supply_desc psy_default = {
1986     .name = "battery",
1987     .type = POWER_SUPPLY_TYPE_BATTERY,
1988     .properties = default_charger_props,
1989     .num_properties = ARRAY_SIZE(default_charger_props),
1990     .get_property = charger_get_property,
1991     .set_property = charger_set_property,
1992     .property_is_writeable = charger_property_is_writeable,
1993     .no_thermal = true,
1994 };
1995
```

支持属性：

```
1960 static enum power_supply_property default_charger_props[] = {
1961     /* Guaranteed to provide */
1962     POWER_SUPPLY_PROP_STATUS,
1963     POWER_SUPPLY_PROP_HEALTH,
1964     POWER_SUPPLY_PROP_PRESENT,
1965     POWER_SUPPLY_PROP_VOLTAGE_NOW,
1966     POWER_SUPPLY_PROP_VOLTAGE_AVG,
1967     POWER_SUPPLY_PROP_CAPACITY,
1968     POWER_SUPPLY_PROP_ONLINE,
1969     POWER_SUPPLY_PROP_CHARGE_FULL,
1970     POWER_SUPPLY_PROP_CONSTANT_CHARGE_CURRENT,
1971     POWER_SUPPLY_PROP_INPUT_CURRENT_LIMIT,
1972     POWER_SUPPLY_PROP_CURRENT_NOW,
1973     POWER_SUPPLY_PROP_CURRENT_AVG,
1974     POWER_SUPPLY_PROP_CHARGE_COUNTER,
1975     POWER_SUPPLY_PROP_CHARGE_CONTROL_LIMIT,
1976     /*
1977     * Optional properties are:
1978     * POWER_SUPPLY_PROP_CHARGE_NOW,
1979     * POWER_SUPPLY_PROP_CURRENT_NOW,
1980     * POWER_SUPPLY_PROP_TEMP, and
1981     * POWER_SUPPLY_PROP_TEMP_AMBIENT,
1982     */
1983 };
1984
```

3.2 Charger IC

sim8500 内置 charge 模块集成在 pm2721 上。

源文件： bsp/kernel/kernel4.14/drivers/power/supply/sc2721_charger.c

PSY 描述：


```

639 static const struct power_supply_desc sc2721_charger_desc = {
640     .name = "sc2721_charger",
641     .type = POWER_SUPPLY_TYPE_USB,
642     .properties = sc2721_usb_props,
643     .num_properties = ARRAY_SIZE(sc2721_usb_props),
644     .get_property = sc2721_charger_usb_get_property,
645     .set_property = sc2721_charger_usb_set_property,
646     .property_is_writeable = sc2721_charger_property_is_writeable,
647     .usb_types = sc2721_charger_usb_types,
648     .num_usb_types = ARRAY_SIZE(sc2721_charger_usb_types)
649 };
650

```

支持的属性:

```

629 static enum power_supply_property sc2721_usb_props[] = {
630     POWER_SUPPLY_PROP_CONSTANT_CHARGE_VOLTAGE_MAX,
631     POWER_SUPPLY_PROP_CONSTANT_CHARGE_CURRENT,
632     POWER_SUPPLY_PROP_INPUT_CURRENT_LIMIT,
633     POWER_SUPPLY_PROP_USB_TYPE,
634     POWER_SUPPLY_PROP_STATUS,
635     POWER_SUPPLY_PROP_ONLINE,
636     POWER_SUPPLY_PROP_HEALTH,
637 };
638

```

4 充电配置介绍

4.1 charger-manager 节点

```

charger-manager {

    compatible = "charger-manager";

    cm-name = "battery";

    cm-poll-mode = <2>; // _cm_monitor 轮询模式

    cm-poll-interval = <15000>;// 轮询时间间隔

    cm-battery-stat = <2>; // 电池在位检测方法，电压法

    cm-fullbatt-vchkdirp-ms = <30000>; // 充满电后，检查复充条件的周期

    cm-fullbatt-vchkdirp-volt = <60000>; // 满电后复充电压条件

    cm-fullbatt-voltage = <4210000>; // 软件满电电压判断阈值，必须配置

    cm-fullbatt-current = <100000>; // 软件满电电流判断阈值，必须配置

    cm-fullbatt-capacity = <100>; // 电池满电时百分比
}

```

```
cm-num-chargers = <1>; // charger ic 数量

cm-chargers = "sc2721_charger"; // charger ic 名字

cm-fuel-gauge = "sc27xx-fgu";

/* in deci centigrade */

cm-battery-cold = <200>; // 电池温度低于此温度时，停充

cm-battery-cold-in-minus;

cm-battery-hot = <800>; // 电池温度高于此温度时，停充

cm-battery-temp-diff = <100>; // 电池温度回落复充阈值

/* Allow charging for 6hr */

cm-charging-max = <18000000>; // 允许连续充电的最长时间

/* recovery charging after stop charging 45min */

cm-discharging-max = <2700000>; // 停充后，如果插着充电器，允许复充时间

/* the interval to feed charger watchdog */

cm-wdt-interval = <0>; // 开门狗周期

/* drop voltage in microVolts to allow shutdown */

cm-shutdown-voltage = <3100000>;

/* when 99% of the time is exceeded, it will be forced to 100% */

cm-tickle-time-out = <1500>; // 电量维持在 99% 15 分钟后跳到 100%

/* how much time to allow capacity change */

cm-one-cap-time = <30>; // 允许电量增加 1%最快时间

/* when the safe charging voltage is exceeded, stop charging */

cm-charge-voltage-max = <6500000>; // 充电器过压保护电压阈值
```

```
/* drop voltage in microVolts to restart charging */

cm-charge-voltage-drop = <700000>; // 复充电压条件

cm-jeita-temp-table = <1000 1030 0 4350000>, // jeita 配置

    <1150 1180 700000 4350000>,

    <1450 1420 900000 4350000>,

    <1600 1570 700000 4100000>;

regulator@0 {

    cm-regulator-name = "vddgen0";

    cable@0 {

        cm-cable-name = "USB";

        extcon = <&extcon_gpio>;

    };

};

};
```

源文件: bsp/kernel/kernel4.14/arch/arm/boot/dts/sl8541e-1h10_32b.dts

4.2 Charger IC 节点

```
pmic_charger: charger@ec0 {

    compatible = "sprd,sc27xx-charger", "sprd,sc2721-charger";

    reg = <0xec0>;

    nvmem-cell-names = "cc_cal";

    nvmem-cells = <&cc_cal>;

    status = "okay";

    phys = <&hsphy>;

    monitored-battery = <&bat>;

};
```

源文件: kernel/kernel4.14/arch/arm/boot/dts/sc2721.dtsi

5 debug 介绍

5.1 Healthd log

```
healthd: battery l=23 v=3329 t=37.0 h=2 st=2 c=-419000 fc=2780000 chg=u
```

- battery l: 当前 UI 显示的电池电量
- v: 电池端电压
- h: 当前 health 状态
- st: 当前充电状态
- c: 当前充电电流 μA
- fc: 电池容量 μAh
- chg: 充电器类型, AC 或者 USB

5.2 插拔充电器 log

插充电器:

```
android_work: sent uevent USB_STATE=CONNECTED
```

```
sc2721-charger sc27xx-charger: battery present = 1, charger type = 1
```

```
charger-manager charger-manager: CHARGING
```

```
musb-sprd 20200000.usb: device connection detected from VBUS GPIO.
```

拔充电器:

```
android_work: sent uevent USB_STATE=DISCONNECTED
```

```
sc2720-charger sc27xx-charger: battery present = 1, charger type = 0
```

```
musb-sprd 20200000.usb: device disconnect detected from VBUS GPIO
```

5.3 调试节点

```
Cat sys/class/power_supply/ battery /status //查询充电状态 (正在充电/未充电)
```

```
Cat sys/class/power_supply/battery/capacity // 查询当前手机电量百分比
```

Cat sys/class/power_supply/battery/current_now // 查询当前充电电流

Cat sys/class/power_supply/battery/voltage_now // 查询当前充电电压

Cat sys/class/power_supply/sc2721_charger/usb_type // 查询当前充电器类型 (DCP, CDP, SDP, unknown...)

SIMCom
Confidential