

密级状态：绝密() 秘密() 内部() 公开(√)

RK817_RK809_电量计开发指南

(技术部，第二系统产品部)

文件状态： [] 正在修改 [√] 正式发布	当前版本：	V1.00
	作 者：	陈顺庆
	完成日期：	2019-2-14
	审 核：	XSF
	完成日期：	2019-2-14

福州瑞芯微电子股份有限公司

Fuzhou Rockchips Electronics Co. , Ltd

(版本所有,翻版必究)

版 本 历 史

版本号	作者	修改日期	修改说明	备注
V1.00	陈顺庆	2019.2.14	初版发布	

目 录

RK817_RK809_电量计开发指南	1
前 言	1
1 概述	1
2 驱动文件.....	2
3 DTS 配置说明	3
4 实例配置.....	6
4.1 RK817.....	6
4.2 RK809+BQ25700.....	7
4.3 RK809+BQ24133.....	8
4.4 RK809+双节电池	9
5 关机充电.....	10
6 常见问题.....	10
6.1 如何打开调试信息?	10
6.2 电池校准.....	11
6.2.1 电池校准方式.....	11
6.2.2 何时需要校准	11
6.3 为什么显示电量与 DTS 中的 ocv_table 对不上?	11
6.4 为什么关机后电池电压比 DTS 配置关机电压高?	11
6.5 拔掉电池再开机, 电量变化	11
6.6 Log 打印的电流与实际测量的差很多	11
6.7 为什么电量一直显示都是 50%, 而且一直在充电?	11
6.8 为什么电量一直显示都是 66%, 而且一直在充电?	12
6.9 RK809 为什么读到电池电压不对?	12
6.10 关闭关机充电之后, 重启机器电量跳变?	12

前 言

概述

本文档主要介绍 Rockchip 的 RK817/RK809 子模块电量计的开发指南,旨在介绍相关的概念、配置与一些常见问题的分析定位。

产品版本

芯片名称	内核版本
RK817	Linux4.4
RK809	Linux4.4

读者对象

本文档（本指南）主要适用于以下工程师：

技术支持工程师

软件开发工程师

1 概述

RK817/RK809 是一款高性能 PMIC，集成了多路大电流 DCDC，多个 LDO 和线性开关，1 个 USB 5V 及 boost 输出，高性能 codec，电量计，然而不同的是，RK817 多了充电功能，RK809 不带充电功能。所以下面提到的充电功能默认指的是 RK817。

另外一个不同的，RK809 可以支持双节电池，RK817 只支持单节电池，因为 RK817 只支持最大 5.5v 的输入电压。

2 驱动文件

drivers/power/rk817_battery.c RK817 与 RK809 共用电量计驱动

drivers/power/rk817_charger.c RK817 充电驱动

如果 defconfig 默认没有配置上，请在 menuconfig 中选上：

```
Symbol: BATTERY_RK817 [=y]
Type : boolean
Prompt: RK817 Battery driver
Location:
-> Device Drivers
(1) -> Power supply class support (POWER_SUPPLY [=y])
Defined at drivers/power/Kconfig:540
Depends on: POWER_SUPPLY [=y] && MFD_RK808 [=y]

Symbol: CHARGER_RK817 [=y]
Type : boolean
Prompt: RK817 Charger driver
Location:
-> Device Drivers
(2) -> Power supply class support (POWER_SUPPLY [=y])
Defined at drivers/power/Kconfig:547
Depends on: POWER_SUPPLY [=y] && MFD_RK808 [=y]
```

同时记住请把 test_power 驱动去掉：

```
Symbol: TEST_POWER [=y]
Type : tristate
Prompt: Test power driver
Location:
-> Device Drivers
(1) -> Power supply class support (POWER_SUPPLY [=y])
Defined at drivers/power/Kconfig:67
Depends on: POWER_SUPPLY [=y]
```

说明：test power 是一个向 android 固定上报电池状态的驱动，主要是为了在没有电量计驱动的情况下使用。

3 DTS 配置说明

```
battery {
    compatible = "rk817,battery";
    ocv_table = <3500 3625 3685 3697 3718 3735 3748
                3760 3774 3788 3802 3816 3834 3853
                3877 3908 3946 3975 4018 4071 4106>;
    design_capacity = <2500>;
    design_qmax = <2750>;
    bat_res = <100>;
    sleep_enter_current = <300>;
    sleep_exit_current = <300>;
    sleep_filter_current = <100>;
    power_off_thresd = <3500>;
    zero_algorithm_vol = <3850>;
    max_soc_offset = <60>;
    monitor_sec = <5>;
    sample_res = <10>;
    virtual_power = <0>;
};

charger {
    compatible = "rk817,charger";
    min_input_voltage = <4500>;
    max_input_current = <1500>;
    max_chrg_current = <2000>;
    max_chrg_voltage = <4200>;
    chrg_term_mode = <0>;
    chrg_finish_cur = <300>;
    virtual_power = <0>;
    dc_det_adc = <0>;
    extcon = <&u2phy>;
};
```

battery 是 RK817 和 RK809 电量计驱动节点。

charger 是 RK817 充电驱动节点。

battery 节点参数说明：

【ocv_table】

开路电压-电量表。即“电压对应电量”，一共 21 个电压值，分别对应 0% → 100%，电压值之间的电量步进为 5%。该数据表可以由电池原厂提供，也可以由 RK 深圳分公司进行测量，或者 RK 提供的测量工具得到，具体请咨询深圳分公司相关工程师。

【design_capacity】

实际电池容量。经实际测量后确定的实际可用容量。例如标称 4000mah，但是实测只有 3850mah，则该值请填写 3850。

【design_qmax】

最大容量值，主要用途是作为软件处理的纠错条件之一。目前请该值请填写标称容量的 1.1 倍数：即标称容量*1.1。

【bat_res】

电池内阻。主要在放电算法中会用到，非常重要！该值在测量 ocv_table 时一起获取，所以请注意这个参数的测量，切勿遗漏。

【sleep_enter_current】

进入松弛模式的条件之一。目前填写 300，不做改动。

在极低负载情况下（目前只针对于二级待机），如果系统的负载电流持续超过一定时间（软件可配）都小于某个阈值，则电量计模块进入松弛模式。

在松弛模式下电量计每隔 8 分钟会采集一组电压，我们称之为松弛电压。用途：二级待机的负载很小，我们近似地认为松弛电压近似于开路电压，因此驱动处理上，在系统从二级待机唤醒且满足一定条件时会用它查询 ocv_table 表进行库仑计的校正。

【sleep_exit_current】

退出松弛模式的条件之一。目前填写 300，不做改动。

【sleep_filter_current】

过滤无效的松弛电流。目前填写 100，不做改动。

【power_off_thresd】请仔细阅读和理解

期待的系统关机电压，单位：mV。特别注意：该值指的是 **VSYS 的瞬时电压**，而不是 vbat 端的电压（电量计采集的是 vbat 端的电压）！原理说明：Vbat 端的电压需要经过一个阻值大约 50 毫欧的 mos 管后（除此外，其实另外还有部分 PCB 走线带来的阻抗）才转换为 VSYS 供给系统，所以把 VSYS 作为关机点依据才是正确的。由此我们可知：相同的 vbat 端电压，当前的负载电流越大，则 vsys 端电压就越低；反之，相同 vsys 下，当前负载电流越大，对应的 vbat 电压也就越高。RK 的平台不建议 vsys 端的电压低于 3.4v，这样容易导致 VCC_IO（3.3v）等 dcdc/ldo 的供电不稳定。

【zero_algorithm_vol】

进入电压+库仑计放电模式的电压值，单位：mV。低于该值，进入电压+库仑计结合的软件

放电算法。建议： 4.2v 电池设置为 3850mv， 4.3v 及其以上的电池设置为 3950mv。

【max_soc_offset】

开机校正时允许的最大电量误差。如果关机至少 30 分钟，则开机时会进行一次 ocv 表的电量查询，并且对比关机前的电量，如果偏差超过 max_soc_offset，即进行强制校正，把电量设置为 ocv 表对应的真实值。例如：当前显示电量是 20%，但是根据 ocv 电压推算的实际电量为 80%，则此时显示的电量直接显示为 80%。一般在发生死机后会出现这种电量偏差极大的情况，这个值的大小依客户的可接受程度，由客户自己进行设置，不建议这个值小于 60

【monitor_sec】

轮询时间（秒）。电量计驱动是需要不停地轮询才能正常工作，期间需要进行不少 I2C 读写操作，但是考虑到不同平台上 I2C 的健壮程度不同，所以预留该配置选项。目前建议 5~10s 比较合适，设置为 5s 是最佳选择。

【sample_res】

电池端附近的采样电阻大小，单位：毫欧。库仑计是通过该电阻来获知当前系统的电流大小，请根据实际硬件贴的电阻大小填写。目前电阻的大小只支持 10mR 或者 20mR。

【virtual_power】

测试模式。有时候在拷机过程中不希望因为电量、充电电流等原因导致系统供电不足导致系统关机。设置该值为 1，即放开充电电流限制，系统输入电流始终为 max_input_current 来满足供电。此时驱动始终上报给 android，当前为充电状态，电量 66%。

charger 节点参数说明：

【min_input_voltage】

输入电压限制。目前有如下档位（单位： mV）：

<4000, 4100, 4200, 4300, 4400, 4500, 4600, 4700>

【max_input_current】

最大输入电流。目前有如下档位（单位： mA）：

<450, 80, 850, 1500, 1750, 2000, 2500, 3000>

注意，第二个档位是 80，不是 800！使用中一般不去设置 80ma 的档位。

如果配置不在上面这些档位里面的，比如配置<1700>，那驱动会设置输入电流为 1500MA。

【max_chrg_current】

最大充电电流。目前有如下档位（单位： mA）：

<1000, 1500, 2000, 2500, 2750, 3000, 3500, 500>

如果配置不在上面这些档位里面的，比如配置<1700>，那驱动会设置输入电流为 1500MA。

【max_chrg_voltage】

最大充电电压，即电池满充的截止电压。目前有如下档位（单位： mV）：

<4000, 4100, 4200, 4300, 4400, 4500, 4600, 4700>

【chrg_term_mode】

配置模拟/数字模式，0:模拟， 1:数字。

不要配置成 1，可能会导致电池电压充不上去。

【chrg_finish_cur】

充电截止电流。

当充电电流达到截止电流，电压达到截止电压，且没有发生输入限流的情况下产生充电截止信号，不再继续充电，我们认为此时电池已经充满。（这里需要输入限流的条件来判读，否则无法分清是真的充电电流变小了，还是因为此时的系统负载比较大导致给电池的充电电流小）

【virtual_power】

测试模式。设置该值为 1，无论有没有接充电器都会上报正在充电状态。

【dc_det_adc】

RK817 DC 充电目前还不支持通过 adc 检测。

【extcon】

RK817 通过注册 extcon 通知链来获取充电类型。

如果是 usb 充电口，extcon = <&u2phy>，如果多个 usb 口，请配置成对应的 u2phy；

如果是 type-c 充电口，extcon = <&fusb0>，如果多个 type-c 口，请配置成对应的 fusb0。

4 实例配置

目前有以下几种组合方案：

RK817(自带充电和电量计功能)

RK809+BQ25700(充电电流为软件控制)

RK809+BQ24133(充电电流为硬件控制，BQ24133 不需要驱动)

4.1 RK817

RK817 有充电功能，所以可以参考如下配置：

```
battery {  
    compatible = "rk817,battery";
```

```

ocv_table = <3500 3625 3685 3697 3718 3735 3748
            3760 3774 3788 3802 3816 3834 3853
            3877 3908 3946 3975 4018 4071 4106>;
design_capacity = <2500>;
design_qmax = <2750>;
bat_res = <100>;
sleep_enter_current = <300>;
sleep_exit_current = <300>;
sleep_filter_current = <100>;
power_off_thresd = <3500>;
zero_algorithm_vol = <3850>;
max_soc_offset = <60>;
monitor_sec = <5>;
sample_res = <10>;
virtual_power = <0>;
};

charger {
    compatible = "rk817,charger";
    min_input_voltage = <4500>;
    max_input_current = <1500>;
    max_chrg_current = <2000>;
    max_chrg_voltage = <4200>;
    chrg_term_mode = <0>;
    chrg_finish_cur = <300>;
    virtual_power = <0>;
    dc_det_adc = <0>;
    extcon = <&u2phy>;
};

```

4.2 RK809+BQ25700

因为 RK809 没有充电功能，所以 **charger** 节点不用配置，但需要把 BQ25700 的节点打开。

BQ25700 我们这边用来举例，不对参数进行说明。

```

battery {
    compatible = "rk817,battery";
    ocv_table = <3500 3625 3685 3697 3718 3735 3748
                3760 3774 3788 3802 3816 3834 3853
                3877 3908 3946 3975 4018 4071 4106>;
    design_capacity = <2500>;
    design_qmax = <2750>;
    bat_res = <100>;
    sleep_enter_current = <300>;
    sleep_exit_current = <300>;
};

```

```
sleep_filter_current = <100>;
power_off_thresd = <3500>;
zero_algorithm_vol = <3850>;
max_soc_offset = <60>;
monitor_sec = <5>;
sample_res = <10>;
virtual_power = <0>;

};

.....

bq25700: bq25700@6b {
    compatible = "ti,bq25703";
    reg = <0x6b>;
    extcon = <&fusb0>;
    interrupt-parent = <&gpio1>;
    interrupts = <RK_PA1 IRQ_TYPE_LEVEL_LOW>;
    pinctrl-names = "default";
    pinctrl-0 = <&charger_ok_int>;
    ti,charge-current = <1500000>;
    ti,max-charge-voltage = <8704000>;
    ti,max-input-voltage = <20000000>;
    ti,max-input-current = <6000000>;
    ti,input-current-sdp = <500000>;
    ti,input-current-dcp = <2000000>;
    ti,input-current-cdp = <2000000>;
    ti,input-current-dc = <2000000>;
    ti,minimum-sys-voltage = <6700000>;
    ti,otg-voltage = <5000000>;
    ti,otg-current = <500000>;
    ti,input-current = <500000>;
    pd-charge-only = <0>;
    status = "disabled";

};
```

4.3 RK809+BQ24133

因为 BQ24133 不需要软件驱动，并且我们的参考图没有留 DC 充电检测的 io 脚，但是可以通过 RK809 的寄存器来判断充电状态。

虽然 BQ24133 不需要驱动，但是系统需要驱动上报状态状态，所以为了兼容硬件，我们把充电 psy 放到 rk817_battery.c 驱动里面上报。

```
battery {
```

```
compatible = "rk817,battery";
ocv_table = <3500 3625 3685 3697 3718 3735 3748
             3760 3774 3788 3802 3816 3834 3853
             3877 3908 3946 3975 4018 4071 4106>;
design_capacity = <2500>;
design_qmax = <2750>;
bat_res = <100>;
sleep_enter_current = <300>;
sleep_exit_current = <300>;
sleep_filter_current = <100>;
power_off_thresd = <3500>;
zero_algorithm_vol = <3850>;
max_soc_offset = <60>;
monitor_sec = <5>;
sample_res = <10>;
virtual_power = <0>;
register_chg_psy = <1>;
};
```

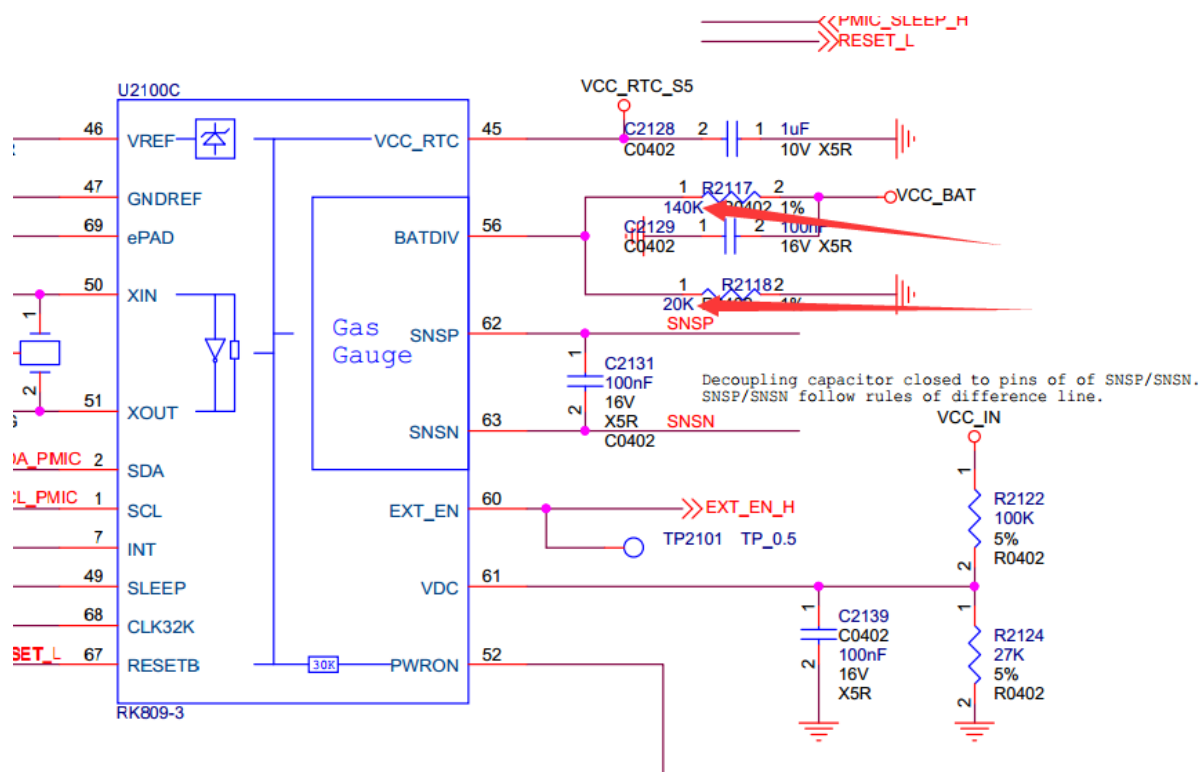
register_chg_psy = <1>; 这个需要配置成 1，目的是 RK809 驱动上报充电状态。

4.4 RK809+双节电池

```
battery {
    compatible = "rk817,battery";
    ocv_table = <7000 7250 7370 7384 7436 7470 7496
                7520 7548 7576 7604 7632 7668 7706
                7754 7816 7892 7950 8036 8142 8212>;
    design_capacity = <2500>;
    design_qmax = <2750>;
    bat_res = <100>;
    sleep_enter_current = <300>;
    sleep_exit_current = <300>;
    sleep_filter_current = <100>;
    power_off_thresd = <7000>;
    zero_algorithm_vol = <7700>;
    max_soc_offset = <60>;
    monitor_sec = <5>;
    sample_res = <10>;
    virtual_power = <0>;
    bat_res_up = <140>;
    bat_res_down = <20>;
};
```

ocv_table、ower_off_thresd、zero_algorithm_vol 需要把对应的电压配置成双节电池的电压。

bat_res_up/bat_res_down 分压电阻的值，请看硬件原理图：



5 关机充电

参考《Rockchip-Developer-Guide-UBoot-nextdev.pdf》的<5.8 充电驱动>章节。

6 常见问题

6.1 如何打开调试信息？

- 1、编译前把驱动第一行的 `static int dbg_enable = 0;`改为 1。
- 2、如果固件没有打开 `dbg_enable`，运行是也可以串口输入如下命令进行开关：

```
./pack_resource.sh <input resource.ing>
```

```
打开: echo 1 > /sys/module/rk817_battery/parameters/dbg_llevel
```

```
echo 1 > /sys/module/rk817_charger/parameters/dbg_llevel
```

```
关闭: echo 0 > /sys/module/rk817_battery/parameters/dbg_llevel
```

```
echo 0 > /sys/module/rk817_charger/parameters/dbg_llevel
```

6.2 电池校准

6.2.1 电池校准方式

- 1、卸下电池 10s 左右，再重新接上。
- 2、做一次完整的充放电。

6.2.2 何时需要校准

- 1、当 DTS 配置的电池容量有改变时；
- 2、很明显电量已经不准（原因可能是机器死机、某些特别的非电量计压力测试等）；
- 3、电量计专项拷机前校正一次，保证电池是在准确的情况下开始的测试，这样才有意义（只需要所有测试项的最开始校正一次即可，不用每个 case 测试前都校

6.3 为什么显示电量与 DTS 中的 ocv_table 对不上？

ocv_table 是开路不带负载情况状态下的【电压-电量】的比值，并且我们只是在开机校正、休眠校正时用到这个表。

6.4 为什么关机后电池电压比 DTS 配置关机电压高？

关机电压以最后 Log 打印的实时电压为准，而且这个关机电压是 vsys 电压（具体参考 power_off_thresd 说明）我们要保证的是实时电压不低于预设的关机点。并且关机后系统下电，锂电池极化慢慢消失，会有一个电压回升的过程，这是锂电池的特性。

6.5 拔掉电池再开机，电量变化

- 1、拔掉电池后 PMIC 完全掉电，此时再开机只能 ocv 电压查询 ocv_table 反推电量，所以是正常的，是一次电池的重新校正；
- 2、拆卸后希望电池电量不跳变？几乎不可能，除非软件做规避：把关机前的电量写到文件里，上电后再去读。客户有需求的话，请客户自己增加这部分规避处理的代码。

6.6 Log 打印的电流与实际测量的差很多

请确认选用的是 20/10 毫欧的采样电阻且电阻精度够高；其次请确认焊点焊接干净，采样电阻应该位于 BAT-和 GND 之间。

6.7 为什么电量一直显示都是 50%，而且一直在充电？

开了 test_power 驱动，请 DTS 中把 test_power disabled 掉。

6.8 为什么电量一直显示都是 66%，而且一直在充电？

当前没有接入电池；或者 DTS 的 virtual_power 被配置成了 1，请配置成 0。

6.9 RK809 为什么读到电池电压不对？

根据 4.4 节定位，应该是 dts 中 bat_res_up/bat_res_down 配置错了。

6.10 关闭关机充电之后，重启机器电量跳变？

确认下 dts 中 charge-animation 是不是被 disabled 掉了？如下：

```
charge-animation {  
    compatible = "rockchip,uboot-charge";  
    rockchip,uboot-charge-on = <1>;  
    rockchip,android-charge-on = <0>;  
    rockchip,uboot-low-power-voltage = <3450>;  
    rockchip,screen-on-voltage = <3500>;  
    rockchip,uboot-exit-charge-level = <2>;  
    rockchip,uboot-exit-charge-voltage = <3500>;  
    status = "disabled";  
};
```

这么修改是不正确的，因为 charge-animation 关掉之后 uboot 的电量计驱动就不会跑了，很多初始化在 uboot 电量计驱动里面做，所以这个必须要开。正确的修改如下：

```
charge-animation {  
    compatible = "rockchip,uboot-charge";  
    rockchip,uboot-charge-on = <0>;  
    rockchip,android-charge-on = <0>;  
    rockchip,uboot-low-power-voltage = <3450>;  
    rockchip,screen-on-voltage = <3500>;  
    rockchip,uboot-exit-charge-level = <2>;  
    rockchip,uboot-exit-charge-voltage = <3500>;  
    status = "okay";  
};
```

把 charge-animation 打开，rockchip,uboot-charge-on 和 rockchip,android-charge-on 设置成 0。