

Rockchip

电源 独立 DCDC

开发指南

发布版本:1.0

日期:2017.05

前言

概述

产品版本

芯片名称	内核版本
RK3399	Linux4.4
RK3328	Linux4.4
RK3368	Linux4.4
RK3288	Linux4.4
RK3036	Linux4.4
RK312X	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

修订记录

日期	版本	作者	修改说明
2017-05-`9	V1.0	ZQ	初始版本

目录

1 PWM 调压	1-1
1.1 驱动文件与 DTS 节点:	1-1
2 SYR8XX 调压	2-2
2.1 驱动文件与 DTS 节点:	2-2
3 XZ321X 调压	3-3
3.1 驱动文件与 DTS 节点:	3-3
4 DEBUG 接口	4-5
4.1 读取电源树:	4-5
4.2 手动设置电压	4-5

1 PWM 调压

1.1 驱动文件与 DTS 节点：

驱动文件所在位置：

drivers/regulator/pwm-regulator.c

Dts 节点：

```
vdd_center: vdd-center {
    compatible = "pwm-regulator";
    rockchip,pwm_id = <2>;
    rockchip,pwm_voltage = <900000>;
    pwms = <&pwm2 0 25000 1>;
    regulator-name = "vdd_center";
    regulator-min-microvolt = <800000>;
    regulator-max-microvolt = <1400000>;
    regulator-always-on;
    regulator-boot-on;
};
```

参数说明：

(1)

```
rockchip,pwm_id = <2>;//pwm2
```

```
rockchip,pwm_voltage = <900000>;//uboot 中 Init 的电压
```

这两个参数主要是给 uboot 使用，kernel 不使用。

(2)

```
pwms = <&pwm2 0 25000 1>;
```

Pwm2 是引用 pwm2 节点，25000 是 pwm 的周期，1 是 pwm 电路极性是反极性。

Pwm 电路极性：

正极性：pwm 占空比越大，输出电压越高

反极性：pwm 占空比越大，输出电压越低

(3)

```
regulator-name = "vdd_center";
```

Pwm 输出电源的 name，用于调压时引用

(4)

```
regulator-min-microvolt = <800000>;
```

```
regulator-max-microvolt = <1400000>;
```

Pwm 电路硬件支持的最大最小电压。这个一定要是实际的硬件值。（测试方法：把 pwm 口强制拉高拉低后对应输出的电压）

(5)

```
regulator-always-on;
```

电源是否是常供电的，如果需要自己管理控制开关，此属性去掉。

(6)

```
regulator-boot-on;
```

Uboot 中使用，用于判读此路是否需要在 Uboot 阶段就开启。

2 SYR8XX 调压

2.1 驱动文件与 DTS 节点：

驱动文件所在位置：

drivers/regulator/fan53555.c

Dts 节点：

```
vdd_cpu_b: syr827@40 {
    compatible = "silergy,syr827";
    reg = <0x40>;
    vin-supply = <&vcc5v0_sys>;
    regulator-compatible = "fan53555-reg";
    pinctrl-0 = <&vsel1_gpio>;
    vsel-gpios = <&gpio1 17 GPIO_ACTIVE_HIGH>;
    regulator-name = "vdd_cpu_b";
    regulator-min-microvolt = <712500>;
    regulator-max-microvolt = <1500000>;
    regulator-ramp-delay = <1000>;
    fcs,suspend-voltage-selector = <1>;
    regulator-always-on;
    regulator-boot-on;
    regulator-initial-state = <3>;
    regulator-state-mem {
        regulator-off-in-suspend;
    };
};
```

参数说明：

(1)

```
vin-supply = <&vcc5v0_sys>;
```

硬件输入电压，没有实际意义，主要是为了构建电源树使用。

(2)

```
pinctrl-0 = <&vsel1_gpio>;
vsel-gpios = <&gpio1 17 GPIO_ACTIVE_HIGH>;
fcs,suspend-voltage-selector = <1>;
```

这个重点说明一下。

这个 IO 是用来切换两组不同电压的，但是当前的做法是用来快速切换开关的。

```
fcs,suspend-voltage-selector = <1>;
```

vsel 脚为低的时候输出电压，拉高时关闭电压。默认是下拉的 IO。

```
fcs,suspend-voltage-selector = <0>;
```

vsel 脚为高的时候输出电压，拉低时关闭电压。默认是上拉的 IO。

这个要以实际的硬件填写。

备注：

VSEL 脚的功能也可以不做快速切换开关，用于休眠唤醒切换电压，只要去掉：

```
pinctrl-0 = <&vsel1_gpio>;
```

```
vsel-gpios = <&gpio1 17 GPIO_ACTIVE_HIGH>;
```

此时 vsel 脚接到 pmic_sleep 上，功能：

```
fcs,suspend-voltage-selector = <1>;
```

vsel 脚为低的时候输出运行电压，拉高时输出待机电压（也可以设置成待机关闭）。默认是下拉的 IO。

```
fcs,suspend-voltage-selector = <0>;
```

vsel 脚为高的时候输出运行电压，拉低时输出待机电压也可以设置成待机关闭）。默认是上拉的 IO。

(3)

```
regulator-name = "vdd_cpu_b";
```

Pwm 输出电源的 name，用于调压时引用

(4)

```
regulator-min-microvolt = <712500>;
```

```
regulator-max-microvolt = <1500000>;
```

软件限定的最大和最小值，超出范围后不允许设置。

(5)

```
regulator-always-on;
```

电源是否是常供电的，如果需要自己管理控制开关，此属性去掉。

(6)

```
regulator-boot-on;
```

Uboot 中使用，用于判读此路是否需要在 Uboot 阶段就开启。

(7)

```
regulator-ramp-delay = <1000>;
```

这个是控制调整电压的上升速度的。一般不需要修改，已经配置成最优了。

3 XZ321X 调压

3.1 驱动文件与 DTS 节点：

驱动文件所在位置：

drivers/regulator/xz3216.c

Dts 节点：

```
xz3216: xz3216@60 {
    compatible = "xz3216";
    reg = <0x60>;
    status = "okay";
    regulators {
        #address-cells = <1>;
        #size-cells = <0>;
        xz3216_dc1: regulator@0 {
            reg = <0>;
            regulator-compatible = "xz_dcdc1";
            regulator-name = "vdd_cpu_l";
            regulator-min-microvolt = <712500>;
```

```

        regulator-max-microvolt = <1400000>;
        regulator-always-on;
        regulator-boot-on;
        //regulator-initial-mode = <0x1>;
        regulator-initial-state = <3>;
        regulator-state-mem {
            regulator-off-in-suspend;
            regulator-suspend-microvolt = <1100000>;
        };
    };
};
};

```

参数说明：

备注：

(1)

```
regulator-name = "vdd_cpu_l";
```

输出电源的 name，用于调压时引用

(2)

```
regulator-min-microvolt = <712500>;
```

```
regulator-max-microvolt = <1500000>;
```

软件限定的最大和最小值，超出范围后不允许设置。

(3)

```
regulator-always-on;
```

电源是否是常供电的，如果需要自己管理控制开关，此属性去掉。

(4)

注意调频调压的改动：

如果此路是给 cpu 小核使用，还要修改：

```

&cpu_l0 {
    cpu-supply = <&xz3216_dc1>;
};
&cpu_l1 {
    cpu-supply = <&xz3216_dc1>;
};
&cpu_l2 {
    cpu-supply = <&xz3216_dc1>;
};
&cpu_l3 {
    cpu-supply = <&xz3216_dc1>;
};

```

如果此路是给 cpu 大核使用，还要修改：

```

&cpu_b0 {
    cpu-supply = <&xz3216_dc1>;
};
&cpu_b1 {
    cpu-supply = <&xz3216_dc1>;
};

```

```

如果此路给 GPU 使用，还需要修改: &gpu {
    status = "okay";
    mali-supply = <&xz3216_dc1>;
};

```

这个的配置，要依据硬件中。Xz3126 实际的供电情况决定。（默认是按照发布的硬件电路配置的）

4 DEBUG 接口

4.1 读取电源树：

```
cat d/regulator/regulator_summary
```

```
rk3399_mid:/ #
rk3399_mid:/ # cat d/regulator/regulator_summary[ 8.867809] capability: warning: 'main' uses 32-bit d
regulator      use open bypass voltage current  min  max
-----
regulator-dummy 0  3  0  0mV  0mA  0mV  0mV
ff100000.saradc  0  0  0  0mV  0mA  0mV  0mV
ff960000.mipi.0  0  0  0  0mV  0mA  0mV  0mV
backlight       0  0  0  0mV  0mA  0mV  0mV
vcc3v3_sys      0 14  0 3300mV 0mA 3300mV 3300mV
vdd_log         0  0  0  900mV 0mA  750mV 1350mV
vdd_cpu_1       0  1  0  925mV 0mA  750mV 1350mV
cpu0            0  0  0  925mV 0mA  925mV  925mV
vcc_ddr         0  0  0 3300mV 0mA  0mV  0mV
vcc_1v8         0  0  0 1800mV 0mA 1800mV 1800mV
vcc1v8_dvp      0  1  0 1800mV 0mA 1800mV 1800mV
ff770000.syscon:io-domains 0  0  0  0mV  0mA  0mV  0mV
vcc3v0_tp       0  0  0 3000mV 0mA 3000mV 3000mV
vcc1v8_pmu      0  1  0 1800mV 0mA 1800mV 1800mV
ff320000.syscon:io-domains 0  0  0  0mV  0mA  0mV  0mV
vcc_sd          0  2  0 3300mV 0mA 1800mV 3300mV
fe320000.dwmmc  0  0  0 3300mV 0mA 3300mV 3300mV
ff770000.syscon:io-domains 0  0  0  0mV  0mA  0mV  0mV
vcca3v0_codec   0  0  0 3000mV 0mA 3000mV 3000mV
vcc_1v5         0  0  0 1500mV 0mA 1500mV 1500mV
vcca1v8_codec   0  1  0 1800mV 0mA 1800mV 1800mV
ff770000.syscon:io-domains 0  0  0  0mV  0mA  0mV  0mV
vcc_3v0         0  1  0 3000mV 0mA 3000mV 3000mV
ff770000.syscon:io-domains 0  0  0  0mV  0mA  0mV  0mV
vcc3v3_s3       0  0  0 3300mV 0mA  0mV  0mV
vcc3v3_s0       0  0  0 3300mV 0mA  0mV  0mV
vcc5v0_host     0  2  0  0mV  0mA  0mV  0mV
phy-ff770000.syscon:usb2-phy@e460.4 0  0  0  0mV  0mA  0mV  0mV
phy-ff770000.syscon:usb2-phy@e450.2 0  0  0  0mV  0mA  0mV  0mV
vcc_phy         0  1  0  0mV  0mA  0mV  0mV
fe300000.ethernet 0  0  0  0mV  0mA  0mV  0mV
vcc5v0_sys      0  2  0 5000mV 0mA 5000mV 5000mV
vdd_cpu_b       0  1  0 1200mV 0mA  712mV 1500mV
cpu4            0  0  0 1200mV 0mA 1200mV 1200mV
vdd_gpu         0  1  0  800mV 0mA  712mV 1500mV
ff9a0000.gpu    0  0  0  800mV 0mA  800mV 1500mV
vdd_center      0  0  0 896mV 0mA  800mV 1400mV
rk3399_mid:/ #
```

4.2 手动设置电压

开启宏：

Device Drivers ->

SOC (System On Chip) specific Drivers ->

选择 Rockchip pm_test support

设置电压接口：

```
echo vdd_center 1000000 > sys/pm_tests/clk_volt
```