pmu 接口使用说明书

2.0

2018.12.24

文档履历

版本号	日期	制/修订人	内容描述
1.0	2016.12.05	AWA1053	
2.0	2018.12.24	AWA1430	修改格式

目录

pmic 使用文档 (axp305/axp806)

pmic 使用简介

本文档适用于 axp806 和 axp305. pmic，包含 regulator，power-supply，power-key。pmic 的配置是通过设备树进行配置. pmic 在内核中的设备是多个设备同时存在，并存在层次对应关系。pmic 的对应关系为:

```
pmic主设备(MFD)
  |
  +------> regulator device
  |
  +------> power key device
  |
  +------> power-supply device
  |
  +------> gpio device
|
+------> wdt device
```

pmic 设备树配置示例:

```
pmu0: pmu{ compatible = "x-powers,axp806"; reg = <0x36>;

    standby_param: standby_param {
        vcc-dram = <0x8>;
    };

    regulators{
        reg_dcdc1: dcdca {
            regulator-name = "axp806-dcdca";
            regulator-min-microvolt = <600000>;
            regulator-max-microvolt = <1520000>;
```

```
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
    };

    reg_dcdc2: dcdcb {
        regulator-name = "axp806-dcdcb";
        regulator-min-microvolt = <1000000>;
        regulator-max-microvolt = <2550000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
    };

    reg_dcdc3: dcdcc {
        regulator-name = "axp806-dcdcc";
        regulator-min-microvolt = <600000>;
        regulator-max-microvolt = <1520000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
    };

    reg_dcdc4: dcdcd {
        regulator-name = "axp806-dcdcd";
        regulator-min-microvolt = <600000>;
        regulator-max-microvolt = <3300000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
    };

    reg_dcdc5: dcdce {
        regulator-name = "axp806-dcdce";
        regulator-min-microvolt = <1100000>;
        regulator-max-microvolt = <3400000>;
```

```
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
    };

    reg_aldo1: aldo1 {
        regulator-name = "axp806-aldo1";
        regulator-min-microvolt = <3300000>;
        regulator-max-microvolt = <3300000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
    };
    reg_aldo2: aldo2 {
        regulator-name = "axp806-aldo2";
        regulator-min-microvolt = <3300000>;
        regulator-max-microvolt = <3300000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
    };
    reg_aldo3: aldo3 {
        regulator-name = "axp806-aldo3";
        regulator-min-microvolt = <3300000>;
        regulator-max-microvolt = <3300000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
    };
    reg_bldo1: bldo1 {
        regulator-name = "axp806-bldo1";
        regulator-min-microvolt = <180000>;
        regulator-max-microvolt = <1800000>;
        regulator-step-delay-us = <25>;
        regulator-final-delay-us = <50>;
        regulator-always-on;
```

```
};
reg_bldo2: bldo2 {
    regulator-name = "axp806-bldo2";
    regulator-min-microvolt = <700000>;
    regulator-max-microvolt = <1900000>;
    regulator-step-delay-us = <25>;
    regulator-final-delay-us = <50>;
};
reg_bldo3: bldo3 {
    regulator-name = "axp806-bldo3";
    regulator-min-microvolt = <700000>;
    regulator-max-microvolt = <1900000>;
    regulator-step-delay-us = <25>;
    regulator-final-delay-us = <50>;
};
reg_bldo4: bldo4 {
    regulator-name = "axp806-bldo4";
    regulator-min-microvolt = <700000>;
    regulator-max-microvolt = <1900000>;
    regulator-step-delay-us = <25>;
    regulator-final-delay-us = <50>;
};
reg_cldo1: cldo1 {
    regulator-name = "axp806-cldo1";
    regulator-min-microvolt = <700000>;
    regulator-max-microvolt = <3300000>;
    regulator-step-delay-us = <25>;
    regulator-final-delay-us = <50>;
};
reg_cldo2: cldo2 {
    regulator-name = "axp806-cldo2";
    regulator-min-microvolt = <700000>;
    regulator-max-microvolt = <4200000>;
    regulator-step-delay-us = <25>;
    regulator-final-delay-us = <50>;
```

```
        };
        reg_cldo3: cldo3 {
            regulator-name = "axp806-cldo3";
            regulator-min-microvolt = <700000>;
            regulator-max-microvolt = <3300000>;
            regulator-step-delay-us = <25>;
            regulator-final-delay-us = <50>;
        };
        reg_sw: sw {
            regulator-name = "axp806-sw";
        };
    };
};
```

pmic 主设备

属性配置

reg <u32>
    i2c寄存器地址

regulator

regulator 为系统 regulator_dev 设备，每个 regulator_dev 代表一路电源，设备通过对 regulator_dev 的引用建立 regulator，用来实现对电源的电压设置等功能.

regulator 属性配置

regulator 设备目前暂无设备树属性可以配置。

设备引用

其他设备对regulator_dev设备的引用通过设备树配置。

<name>-supply = <&reg_dcdc1>;

设备中通过对name的获取，可以获取reg_dcdc1的regulator_dev设备，然后对此路电源进行电源开关，电压设置等功能。

具体设备引用参考内核的regulator使用文档。

调试方法

在设备进行开发过程中，难免需要对各路电源进行调试，控制电源各路电压等操作，内核中提供了对电源调试的方式.

####...1 控制各路电压

对各路电压的控制是常用的调试手段，通过对各路不同电压设置，实现功耗，性能，稳定性等信息.

内核通过对每一路电源创建一个virtual设备，通过导出virtual设备的电源控制结点，用来对每一路电源的电压进行控制。

通过进入不同的virtual设备，来控制不同的电源。

virtual设备存在于axp305主设备结点下面，因此设备路径为主设备下面的从设备:axp305的设备举例::

/sys/devices/platform/soc/twi5/i2c-5/5-0034/regulator/regulator.1/reg-virt-consomer.1-dcdca/

通过此路径下面的**max_microvolts**和**min_microvolts**设备结点进行写操作，用来完成对设备电源的控制，

此例为::

echo 3000000 > max_microvolts

echo 3000000 > min_microvolts

设置电压为3000000uv,3000mv,3v

####...2 获取各路电源获取设备

获取有哪几路电源引用了电源，进入需要查看的电源, 进入 /sys/class/regulator/regulator.1 , 查看当前目录下的目录，即可确定有哪几路引用设备。

另外一种方法就是进入 regulator 的 debugfs 结点，用来查看 regulator 的 map 信息。参考读取各路电源状态

####...3 读取各路电源状态

kernel 提供调试结点供电源进行调试进行，我们可以通过 kernel 的调试结点获取各路电源的各个详细状态。首先需要 mount debugfs 文件系统

```
mount -t debugfs none /sys/kernel/debug
cat /sys/kernel/debug/regulator/regulator_summary

regulator                use open bypass voltage current     min     max
---------------------------------------------------------------------------
regulator-dummy             0    1      0    0mV     0mA     0mV     0mV
  uart0                                          0mV     0mV
axp806-dcdca                0    7      0 3300mV     0mA  1500mV  3400mV
  spi0                                           0mV     0mV
  sdc0                                           0mV     0mV
  sdc0                                           0mV     0mV
  sdc0                                           0mV     0mV
  sdc0                                           0mV     0mV
  sdc0                                           0mV     0mV
```

```
   reg-virt-consumer.1                                0mV     0mV
axp806-dcdcb          0    1      0   900mV     0mA   500mV  1540mV
   reg-virt-consumer.2                                0mV     0mV
axp806-dcdcc          0    2      0  1000mV     0mA   500mV  3400mV
   cpu0                                        1000mV  1000mV
   reg-virt-consumer.3                                0mV     0mV
axp806-dcdcd          0    1      0  1500mV     0mA   500mV  1840mV
   reg-virt-consumer.4                                0mV     0mV
axp806-dcdce          0    1      0  1400mV     0mA  1400mV  3700mV
   reg-virt-consumer.5                                0mV     0mV
axp806-rtcldo         0    0      0  1800mV     0mA  1800mV  1800mV
axp806-rtcldo1        0    0      0  1800mV     0mA  1800mV  1800mV
axp806-aldo1          0    1      0  1800mV     0mA   500mV  3500mV
   reg-virt-consumer.8                                0mV     0mV
axp806-aldo2          0    2      0  3300mV     0mA   500mV  3500mV
   spi2                                         0mV     0mV
   reg-virt-consumer.9                                0mV     0mV
axp806-aldo3          0    1      0  3300mV     0mA   500mV  3500mV
   reg-virt-consumer.10                               0mV     0mV
axp806-aldo4          0    1      0  3300mV     0mA   500mV  3500mV
   reg-virt-consumer.11                               0mV     0mV
axp806-bldo1          0    1      0  1800mV     0mA   500mV  3500mV
   reg-virt-consumer.12                               0mV     0mV
axp806-bldo2          0    1      0  3300mV     0mA   500mV  3500mV
   reg-virt-consumer.13                               0mV     0mV
axp806-dldo1          1    1      0  1200mV     0mA   500mV  3500mV
   reg-virt-consumer.14                               0mV     0mV
axp806-dldo2          0    1      0  1200mV     0mA   500mV  1400mV
   reg-virt-consumer.15                               0mV     0mV
axp806-cpusldo        0    0      0   900mV     0mA   500mV  1400mV
```

通过上例可以看出各路电源有哪几路设备请求，已经请求的值和目前各路电源的状态

watchdog device

pmic 会创建一个 hw_timeout 为 4s 的一个看门狗，默认 timeout 为 5s。

使用方法

创建的 watchdog 会在/dev/watchdog*, 如果使能 sunxi-dev 的 soc 内部 watchdog，pmic 的 watchdog 会抢先使用/dev/watchdog -> /dev/watchdog0, 这样保证直接使用/dev/watchdog 为使用 pmic 的 watchdog。在某些情况下面，soc 内部的 watchdog 会存在不能复位 pmic 的情况，这时需要使用 pmic 的 watchdog。1. pmic 的 watchdog 打开后即开启，关闭文件不能关闭看门狗。2. 如需要关闭看门狗，需要发送'V' 字符即可关闭看门狗。3. 看门狗使用标准的 set_timeout 方法设置看门狗时间。4. 通过写 watchdog 可以喂狗，或者使用标准的 ioctl 方法.

设备树

暂未添加设备树支持

寄存器级别调试

寄存器调试是指直接对 pmic 的寄存器进行读写操作，此操作应该对寄存器有了解的情况下进行操作，不正确的操作方式将会导致芯片烧毁。在终端中，对抛出的调试结点进行读写操作，即可对寄存器进行读写操作。无论是读还是写寄存器，都应该首先挂载 debugfs 文件系统。

由于 pmic 是通过 regmap 进行读写操作，应该可以使用 regmap 的调试结点进行对 pmic 的读写访问操作。regmap 的调试结点在 debugfs 文件系统下面，通过对 regmap 调试结点的操作可以对 pmic 的寄存器进行读写访问操作。

写操作

寄存器调试挂载在debugfs文件系统。
mount -t debugfs none /sys/kernel/debug
echo ${reg} ${value} > /sys/kernel/debug/regmap/${dev-name}/registers

实例：

echo 0xff 0x01 > /sys/kernel/debug/regmap/4-0034/registers
写0xff寄存器值为0x01

读操作

寄存器调试挂载在debugfs文件系统。
mount -t debugfs none /sys/kernel/debug
cat /sys/kernel/debug/regmap/${dev-name}/registers

实例：

cat /sys/kernel/debug/regmap/4-0034/registers
读取pmic所有寄存器

1. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner. The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This document neither states nor implies warranty of any kind, including fitness for any particular application.tates nor implies warranty of any kind, including fitness for any particular application.