

iTOP-4412 开发板怎么修改电源管理芯片的输出电压

iTOP-4412 核心板使用的电源管理芯片是三星专门针对 4412 研发的 S5M8767，S5M8767 提供 9 路 BUCK 和 28 路 LDO 输出，每路电压的大小可以通过软件进行设置。

S5M8767 的驱动位于内核的 `drivers/regulator/s5m8767.c` 文件中,Exynos 4412 处理器是通过 I2C 总线来控制 S5M8767 的。S5M8767 在系统启动的过程中会注册到内核里面的 `regulator` 模块里面。

`regulator` 模块是内核用于控制系统中某些设备的电压/电流供应，在嵌入式系统（尤其是手持设备）中，控制耗电量很重要，它直接影响到电池的续航时间。所以，如果系统中某一个模块暂时不使用，就可以通过 `regulator` 关闭其电源；或者降低提供给该模块的电压、电流大小来达到降低功耗的目的。S5M8767 驱动的主要作用就是调用 `regulator_register` 函数向内核注册 `regulator_dev` 设备，每个 `regulator_dev` 代表一个 `regulator` 设备，内核可以分别控制每个 `regulator`。

为了实现 S5M8767 驱动还需要在平台相关的代码里定义 `regulator_init_data` 结构，`regulator_init_data` 用来建立父子 `regulator`、受电模块之间的树状结构，以及一些 `regulator` 的基本信息，比如电压大小等，下面我们来看下 `regulator_init_data` 结构的定义，代码在 `arch/arm/mach-exynos/mach-itop4412.c` 里面，在这个文件里使用宏 `REGULATOR_INIT` 定义了 28 个 LDO 的 `regulator_init_data` 结构，代码如下：

```
#define REGULATOR_INIT(_ldo, \
_name, _min_uV, _max_uV, _always_on, _ops_mask, \
_disabled) \
static struct regulator_init_data s5m8767_##_ldo##_init_data = { \
    .constraints = { \
        .name = _name, \
        .min_uV = _min_uV, \
        .max_uV = _max_uV, \
        .boot_on = _always_on, \
        .apply_uV = 1, \
        .valid_ops_mask = _ops_mask, \
        .state_mem = { \
            .disabled = _disabled, \
            .enabled = !(_disabled), \
        } \
    }, \
    .num_consumer_supplies = ARRAY_SIZE(s5m8767_##_ldo##_supply), \
    .consumer_supplies = &s5m8767_##_ldo##_supply[0], \
}
```

上面的宏定义中，第三个和第四个参数指定了 LDO 的电压最小值和最大值，第五个参数设置 LDO 在系统开始运行时是输出还是关闭的（1 是输出，0 是关闭）。

第六个参数是 LDO 具有哪些功能，例如可以修改电压，电流，改变状态等等，通过位掩码的方式设置，第七个参数是设置在休眠的时候是否由 `PWREN` 引脚控制它的开关（1 是由 `PWREN` 控制，0 是不受 `PWREN` 控制），休眠的时候 `PWREN` 为低电平，LDO 会关闭，系

统唤醒，PWREN 为高电平，LDO 会输出。

例如 LDO2 的定义，如下：

```
REGULATOR_INIT(ldo2, "VDDQ_M12", 1500000, 1500000, 1,  
REGULATOR_CHANGE_STATUS, 1)
```

根据定义，可以知道 LDO2 输出的电压是 1.5v，系统启动的时候会默认输出，系统休眠的时候会关闭。其他的 LDO 的设置原理与 LDO2 是一样的。

系统中 BUCK 的定义，例如 BUCK1：

```
static struct regulator_init_data s5m8767_buck1_data = {  
.constraints = {  
.name = "vdd_mif range",  
.min_uV = 900000,  
.max_uV = 1100000,  
.valid_ops_mask = REGULATOR_CHANGE_VOLTAGE |  
REGULATOR_CHANGE_STATUS,  
.state_mem = {  
.disabled = 1,  
},  
},  
.num_consumer_supplies = 1,  
.consumer_supplies = &s5m8767_buck1_consumer,  
};
```

根据上面的定义，可以知道 BUCK1 的电压范围在 0.9v 到 1.1v，它具有可以修改电压，修改状态的功能（变量 valid_ops_mask）。可以使用函数 regulator_set_voltage 修改 BUCK1 的电压。其他几个 BUCK 的定义原理和 BUCK1 是一样的

如果我们想要修改 8767 的某个 LDO 的输出电压，就可以通过修改对应 LDO 的 regulator_init_data 结构体里面的电压值来实现，修改 BUCK 的电压可以使用函数 regulator_set_voltage 来实现。

注意：因为 S5M8767 的每个 BUCK 和 LDO 都有各自规定的输出最大值，因此在修改输出电压的时候，一定要参照 S5M8767 的 datasheet，确保修改的电压在 datasheet 规定的范围内。