

A83T

AXP813 Linux Power Driver
application document/V1.0

Revision History

Version	Date	Author	History
V1.0	2014.08.28		

Confidential

Table of Contents

A83T.....	1
AXP813 Linux Power Driver application document/V1.0.....	1
Table of Contents	2
1. Overview	3
1. 1. Purpose.....	3
1. 2. Scope	3
1. 3. Related personnel	3
2. Code structure.....	4
3. Configuration.....	5
3. 1. Menuconfig configuration:	5
3. 2. sys_config.fex configuration	5
3.2.1. Regulator tree.....	5
3.2.2. Battery supply	6
4. Regulator	14
4. 1. Regulator demo	14
4.1.1. LDO/DCDC list	14
4.1.2. demo.....	14
4.1.3. Dump file node	15
4. 2. Regulator shell.....	16
5. Battery supply.....	18
5. 1. Battery information	18
5. 2. Sysfs file node	18
5.2.1. MFD.....	18
5.2.2. Power_supply (standard):	19
5.2.3. Power_supply (non standard):	20
6. Others	22
6. 1. GPIO.....	22
6.1.1. AXP GPIO list	22
6.1.2. AXP GPIO demo.....	22
6. 2. AXP813 Register Shell	23
6. 3. Debug and Print.....	24

1. Overview

1. 1. Purpose

Introduce the method of axp813 software, to help the user to understand and debug the PMU.

1. 2. Scope

software : Linux-3.4kernel ◦

1. 3. Related personnel

Allwinnertech A83T PMU driver developers ◦

Confidential

2. Code structure

```
linux-3.4
drivers\power\axp_power
|—— axp-irq.c
|—— axp-state.c
|—— axp-filenode.c
|—— axp-regu.h
|—— axp-regu.c
|—— axp-script.c
|—— axp-regu-script.c
|—— axp-mfd.c
|—— axp-cfg.h
|—— axp-rw.h
|—— Kconfig
|—— Makefile
drivers\power\axp_power\axp81x
|—— axp81x-board.c
|—— axp81x-regu.c
|—— axp81x-init.c
|—— axp81x-sply.c
|—— virtual81x.c
|—— virtural81x-dev.c
|—— axp81x-common.h
|—— axp81x-mfd.h
|—— axp81x-regu.h
|—— axp81x-sply.h
include\linux\mfd
|—— axp-mfd.h
|—— axp-mfd-81x.h
drivers\pinctrl\
|—— pinctrl-axp.c
|—— pinctrl-axp.h
```

The directory `axp_power` contain the code of the public part of the AXP driver, which is shared by all AXP driver.

The directory `axp_power/axp81x` contain the code of the private part of the AXP813 driver.

The directory `drivers\pinctrl\` under the `pinctrl-axp.c` and `pinctrl-axp.h` are AXP813 GPIO driver.

3. Configuration

3.1. Menuconfig configuration:

Enter the Linux3.4 directory, type "make ARCH=arm menuconfig". Please select the following options :

Device Drivers

—— Voltage and Current Regulator Support

—— Power supply class support —> AXP Power drivers —> AXP PMU type (AXP81X driver)

The menu of menuconfig :

```
.config - Linux/arm 3.4.39 Kernel Configuration
AXP Power drivers
[*] AXP81X driver
[ ] AXP22 driver --->
[ ] AXP19 driver
[ ] AXP15 driver
[*] AXP initial charging environment set
[*] AXP charging current set when suspend/resume/shutdown
[ ] AXP use twi as transfer channel
<Select> <Exit> <Help>
```

AXP81X driver is axp813 compiler options.

AXP initial charging environment set is charge set compiler options.

AXP charging current set when suspend/resume/shutdown is charge current set compiler options.

3.2. sys_config.fex configuration

3.2.1. Regulator tree

```
[pmu1_regu]
ldo_count = 23
ldo1      = "axp81x_dcdc1 none sys vcc-emmc vcc-usb0-33 vcc-io vcc-io-gps vcc-io1 vcc-sensor"
ldo2      = "axp81x_dcdc2 none sys vdd-cpua"
ldo3      = "axp81x_dcdc3 none none vdd-cpub"
ldo4      = "axp81x_dcdc4 none none vdd-gpu"
```

```

ldo5      = "axp81x_dcdc5 none sys vcc-dram"
ldo6      = "axp81x_dcdc6 none sys vdd-sys vdd-usb0-09 vdd-hdmi-09"
ldo7      = "axp81x_dcdc7 none none"
ldo8      = "axp81x_rtc none none"
ldo9      = "axp81x_aldo1 none sys vcc-dsi-18 vcc-csi2-18 vcc-lvds-18 vcc-efuse-18 vcc-hdmi-18
vcc-pd"
ldo10     = "axp81x_aldo2 none sys vdd-dramp11 vdd-lpddr-18 vcc-pll-18 vcc-adc-18 vdd-codec-cp
vdd-codec-ldoin"
ldo11     = "axp81x_aldo3 none sys vcc-avcc vcc-pl"
ldo12     = "axp81x_dldo1 none none vcc-wifi-io vcc-io2"
ldo13     = "axp81x_dldo2 none none vcc-lcd-0"
ldo14     = "axp81x_dldo3 none none afvcc-csi iovdd-csi"
ldo15     = "axp81x_dldo4 none none avdd-csi"
ldo16     = "axp81x_eldo1 1 none dvdd-csi-12"
ldo17     = "axp81x_eldo2 1 none vcc_dsi"
ldo18     = "axp81x_eldo3 1 none dvdd-csi-18"
ldo19     = "axp81x_fldo1 none none vcc-hsic-12"
ldo20     = "axp81x_fldo2 none sys vdd-cpus"
ldo21     = "axp81x_gpio0ldo none none vcc-ctp"
ldo22     = "axp81x_gpio1ldo none none vcc-card"
ldo23     = "axp81x_dc1sw none none"

```

Regulator tree configures the power dependency relationship.

The first column is ldo/dcdc name, such as "axp81x_dcdc1".

The second column is the configuration which explain the relationship between ldo/dcdc. For example , the dcdc1 power supply to the eldo. So, we configure ldo16's the second column to 1, which explain ldo16 is supplied by ldo1. If the ldo is independent, which configure to "none" .

The third column is the configuration which explain the ldo whether is the system power supply or not. When the ldo is system power, which is configured to "sys". If not which is configured to "none". The system power supply is always enable state, which can not be disabled by the regulator_disable function.

Beginning with the fourth column, all of the configuration are supply id for each module. The supply id separated by space, each supply id is 20 characters or fewer. Each ldo supports up to 17 supply id.

Notice :

- 1 Please don't modify the ldo/dcdc name, which is used by regulator core.
- 2 The relationship between ldo/dcdc, please configure according to the actual circuit.
- 3 the system power is closely related to standby, please carefully modify.

3.2.2. Battery supply

```

;-----
;pmu1 is AXP813
;-----
[pmu1_para]
pmu_used                = 1

```

pmu_id = 6
pmu_twi_addr = 0x34
pmu_twi_id = 0
pmu_irq_id = 0
pmu_battery_rdc = 100
pmu_battery_cap = 0
pmu_batdeten = 1
pmu_chg_ic_temp = 0
pmu_runtime_chgcur = 1000
pmu_earlysuspend_chgcur = 1000
pmu_suspend_chgcur = 1600
pmu_shutdown_chgcur = 1600
pmu_init_chgvol = 4200
pmu_init_chgend_rate = 15
pmu_init_chg_enabled = 1
pmu_init_bc_en = 0
pmu_init_adc_freq = 800
pmu_init_adcts_freq = 800
pmu_init_chg_pretime = 70
pmu_init_chg_csttime = 720
pmu_batt_cap_correct = 1
pmu_bat_regu_en = 0

pmu_bat_para1 = 0
pmu_bat_para2 = 0
pmu_bat_para3 = 0
pmu_bat_para4 = 0
pmu_bat_para5 = 0
pmu_bat_para6 = 0
pmu_bat_para7 = 0
pmu_bat_para8 = 0
pmu_bat_para9 = 5
pmu_bat_para10 = 8
pmu_bat_para11 = 9
pmu_bat_para12 = 10
pmu_bat_para13 = 13
pmu_bat_para14 = 16
pmu_bat_para15 = 20
pmu_bat_para16 = 33
pmu_bat_para17 = 41
pmu_bat_para18 = 46
pmu_bat_para19 = 50
pmu_bat_para20 = 53
pmu_bat_para21 = 57
pmu_bat_para22 = 61
pmu_bat_para23 = 67

pmu_bat_para24 = 73
pmu_bat_para25 = 78
pmu_bat_para26 = 84
pmu_bat_para27 = 88
pmu_bat_para28 = 92
pmu_bat_para29 = 93
pmu_bat_para30 = 94
pmu_bat_para31 = 95
pmu_bat_para32 = 100

pmu_usbvol_limit = 0
pmu_usbcur_limit = 0
pmu_usbvol = 4000
pmu_usbcur = 0
pmu_usbvol_pc = 4400
pmu_usbcur_pc = 500
pmu_pwroff_vol = 3300
pmu_pwron_vol = 2600
pmu_pekoff_time = 6000
pmu_pekoff_func = 0
pmu_pekoff_en = 1
pmu_pekoff_delay_time = 0
pmu_peklong_time = 1500
pmu_pekon_time = 1000
pmu_pwrok_time = 64
pmu_pwrok_shutdown_en = 0
pmu_reset_shutdown_en = 1
pmu_battery_warning_level1 = 15
pmu_battery_warning_level2 = 0
pmu_restvol_adjust_time = 60
pmu_ocv_cou_adjust_time = 60
pmu_chgled_func = 0
pmu_chgled_type = 0
pmu_vbusen_func = 1
pmu_reset = 0
pmu_IRQ_wakeup = 0
pmu_hot_shutdown = 1
pmu_inshort = 0
power_start = 0

pmu_temp_enable = 0
pmu_charge_ltf = 2261
pmu_charge_hrf = 388
pmu_discharge_ltf = 3200
pmu_discharge_hrf = 237
pmu_temp_para1 = 7466

```

pmu_temp_para2      = 4480
pmu_temp_para3      = 3518
pmu_temp_para4      = 2786
pmu_temp_para5      = 2223
pmu_temp_para6      = 1788
pmu_temp_para7      = 1448
pmu_temp_para8      = 969
pmu_temp_para9      = 664
pmu_temp_para10     = 466
pmu_temp_para11     = 393
pmu_temp_para12     = 333
pmu_temp_para13     = 283
pmu_temp_para14     = 242
pmu_temp_para15     = 179
pmu_temp_para16     = 134

```

	configuration	explanation
1	pmu_used	AXPxx enable feature : 0:disable, 1:enable.
2	pmu_id	0:axp19x,1:axp22x,2:axp806,3:axp808,4:axp809,5:axp803,6:axp813.
3	pmu_twi_addr	AXPxx I2C address.
4	pmu_twi_id	AXPxx is connected to which I2C interface (0 , 1 , 2 ...)
5	pmu_irq_id	irq number (0 irq0,1 irq1,.....).
6	pmu_battery_rdc	The internal resistance of batteryunit m Ω .
7	pmu_battery_cap	Battery capacity, unit mAh. If set the configuration, battery capacity measurement is coulometric mode, otherwise which is the OCV mode.
8	pmu_batdeten	Battery detection : 0:disable 1:enable.
9	pmu_chg_ic_temp	Intelligent charging , PMIC temperature configuration. This feature is currently not open, configured as 0.
10	pmu_runtime_chgcur	The running charge current feature, unit mA. Support :200-2800,200mA/steps.
11	pmu_earlysuspend_chgcur	The earlysuspend charge current feature, unit mA.Support : 200-2800,200mA/steps.
12	pmu_suspend_chgcur	The suspend charge current feature, unit mA.Support : 200-2800,200mA/steps.
13	pmu_shutdown_chgcur	The shutdown charge current feature, unit mA.Support : 200-2800,200mA/steps.
14	pmu_init_chgvol	The voltage of charge target .Support : 4100/4150/4200/4350mV.
15	pmu_init_chgend_rate	The charge end condition feature, charging current 's the percentage of constant current value. Support : 10/20.
16	pmu_init_chg_enabled	The charge enable feature : 0 : disable, 1 : enable.
17	pmu_init_adc_freq	The feature of ADC sampling frequency. Support :

		100/200/400/800 Hz.
18	pmu_init_adcts_freq	The feature of TS ADC sampling frequency. Suppor : 100/200/400/800 Hz.
19	pmu_init_chg_pretime	The time of the trickle charge timeout. Support :40/50/60/70 minutes.
20	pmu_init_chg_csttime	The time of constant current charge timeout. Support : 360/480/600/720 minutes.
21	pmu_batt_cap_correct	The battery capacity correction enable feature. 0 : disable 1 : eanble.
22	pmu_bat_regu_en	When the charge has done, the charge circuit wether is closed or not, : 0 : close 1 : open.
23	pmu_bat_para1	The battery capacity value of the battery no-load valtage 3.13V.
24	pmu_bat_para2	The battery capacity value of the battery no-load valtage 3.27V.
25	pmu_bat_para3	The battery capacity value of the battery no-load valtage 3.34V.
26	pmu_bat_para4	The battery capacity value of the battery no-load valtage 3.41V.
27	pmu_bat_para5	The battery capacity value of the battery no-load valtage 3.48V.
28	pmu_bat_para6	The battery capacity value of the battery no-load valtage 3.52V.
29	pmu_bat_para7	The battery capacity value of the battery no-load valtage 3.55V.
30	pmu_bat_para8	The battery capacity value of the battery no-load valtage 3.57V.
31	pmu_bat_para9	The battery capacity value of the battery no-load valtage 3.59V.
32	pmu_bat_para10	The battery capacity value of the battery no-load valtage 3.61V.
33	pmu_bat_para11	The battery capacity value of the battery no-load valtage 3.63V.
34	pmu_bat_para12	The battery capacity value of the battery no-load valtage 3.64V.
35	pmu_bat_para13	The battery capacity value of the battery no-load valtage 3.66V.
36	pmu_bat_para14	The battery capacity value of the battery no-load valtage 3.7V.
37	pmu_bat_para15	The battery capacity value of the battery no-load valtage 3.73V.
38	pmu_bat_para16	The battery capacity value of the battery no-load valtage 3.77V.
39	pmu_bat_para17	The battery capacity value of the battery no-load valtage 3.78V.

40	pmu_bat_para18	The battery capacity value of the battery no-load valtage 3.8V.
41	pmu_bat_para19	The battery capacity value of the battery no-load valtage 3.82V.
42	pmu_bat_para20	The battery capacity value of the battery no-load valtage 3.84V.
43	pmu_bat_para21	The battery capacity value of the battery no-load valtage 3.85V.
44	pmu_bat_para22	The battery capacity value of the battery no-load valtage 3.87V.
45	pmu_bat_para23	The battery capacity value of the battery no-load valtage 3.91V.
46	pmu_bat_para24	The battery capacity value of the battery no-load valtage 3.94V.
47	pmu_bat_para25	The battery capacity value of the battery no-load valtage 3.98V.
48	pmu_bat_para26	The battery capacity value of the battery no-load valtage 4.01V.
49	pmu_bat_para27	The battery capacity value of the battery no-load valtage 4.05V.
50	pmu_bat_para28	The battery capacity value of the battery no-load valtage 4.08V.
51	pmu_bat_para29	The battery capacity value of the battery no-load valtage 4.1V.
52	pmu_bat_para30	The battery capacity value of the battery no-load valtage 4.12V.
53	pmu_bat_para31	The battery capacity value of the battery no-load valtage 4.14V.
54	pmu_bat_para32	The battery capacity value of the battery no-load valtage 4.15V.
55	pmu_usbvol_limit	The featurer USB adapter voltage limit. 0: disable 1: eanble.
56	pmu_usbcur_limit	The featurer USB adapter current limit. 0: disable 1: eanble.
57	pmu_usbvol	The value of USB adapter voltage limit. Support : 4000/4100/4200/4300/4400/4500/4600 4700 mV, 0-not limit.
58	pmu_usbcur	The value of USB adapter current limit. Support : 500/900/1500/2000/2500/3000/3500 4000 mA, 0-not limit.
59	pmu_usbvol_pc	The value of USB PC voltage limit. Support : 4000/4100/4200/4300/4400/4500/4600 4700 mV, 0-not limit.
60	pmu_usbcur_pc	The value of USB PC current limit. Support : 500/900/1500/2000/2500/3000/3500 4000 mA, 0-not limit.
61	pmu_pwroff_vol	When system shutdown, PMU low voltage protection

		value : 2600/2700/2800/2900 /3000/3100/3200/3300 mV
62	pmu_pwron_vol	When system is runing, PMU low voltage protection value : 2600/2700/2800/2900 /3000/3100/3200/3300 mV
63	pmu_pekoff_time	The time of press power key to shutdown feature, support : 4000/6000/8000/10000 ms
64	pmu_pekoff_func	The long press power key feature, 0: shutdown 1: reset.
65	pmu_pekoff_en	When long press power key, whethor shutdown PMU or not. 0: not shutdown 1: shutdown.
66	pmu_pekoff_delay_time	The active time of long press power key shutdown system, 0/10/20/30/40/50/60/70 seconds.
67	pmu_peklong_time	The time of long power key irq : 1000/1500/2000/2500 ms
68	pmu_pekcon_time	The delay time of power key Startup system : 128/1000/2000/3000 ms
69	pmu_pwrok_time	The delay time of reset key Startup system : 8/16/32/64 ms
70	pmu_pwrok_shutdown_en	This feature is currently not open, configured as 0.
71	pmu_reset_shutdown_en	When PMU reset system, whether close ldo/dcdc or not. 0: not close, 1: close.
72	pmu_battery_warning_level1	Low battery alarm level 1 threshold: 5~20, 1/steps.
73	pmu_battery_warning_level2	Low battery alarm level 2 threshold: 0~15, 1/steps.
74	pmu_restvol_time	The time of battery capacity update, 30/60/120 s
75	pmu_ocv_cou_adjust_time	The time of OCV battery capacity correcting, 30/60/120 s
76	pmu_chgled_func	The CHGLED function feature, : 0: moto, 1: charge led.
77	pmu_chgled_type	The CHGLED mode feature, : 0 : mode A, 1 : mode B.
78	pmu_vbusen_func	The N_VBUSEN mode feature : 0 : input 1 : output.
79	pmu_reset	When long press power key 16s, whethor reset pmu or not. : 0 : not reset, 1 : reset.
80	pmu_IRQ_wakeup	The pmu irq wakeup feanture, 0: not wakeup 1 : wakeup.
81	pmu_hot_shutdown	When PMU is overhot, whethor shutdown system or not. 0: not shutdown, 1: shutdown.
82	pmu_inshort	The feature of ACIN/VBUS short detection, 0 : PMU auto detection , 1 : set to short.
83	pmu_temp_enable	The feature of battery temperature detecion, : 0: disable 1: enable.
84	pmu_charge_ltf	The battery voltage value of the battery is in chargeing state 's temperature lower threshold.
85	pmu_charge_htf	The battery voltage value of the battery is in chargeing state 's upper temperature threshold.
86	pmu_discharge_ltf	The battery voltage value of system must be shutdown state 's temperature lower limit.
87	pmu_discharge_htf	The battery voltage value of system must be shutdown state 's upper temperature limit.
88	pmu_temp_para1	The battery voltage value of the battery temperature -25.
89	pmu_temp_para2	The battery voltage value of the battery temperature -15.

90	pmu_temp_para3	The battery voltage value of the battery temperature -10.
91	pmu_temp_para4	The battery voltage value of the battery temperature -5.
92	pmu_temp_para5	The battery voltage value of the battery temperature 0.
93	pmu_temp_para6	The battery voltage value of the battery temperature 5.
94	pmu_temp_para7	The battery voltage value of the battery temperature 10.
95	pmu_temp_para8	The battery voltage value of the battery temperature 20.
96	pmu_temp_para9	The battery voltage value of the battery temperature 30.
97	pmu_temp_para10	The battery voltage value of the battery temperature 40.
98	pmu_temp_para11	The battery voltage value of the battery temperature 45.
99	pmu_temp_para12	The battery voltage value of the battery temperature 50.
100	pmu_temp_para13	The battery voltage value of the battery temperature 50.
101	pmu_temp_para14	The battery voltage value of the battery temperature 60.
102	pmu_temp_para15	The battery voltage value of the battery temperature 70.
103	pmu_temp_para16	The battery voltage value of the battery temperature 80.

4. Regulator

4.1. Regulator demo

4.1.1. LDO/DCDC list

AXP813 :

Axp ldo/dcdc name	ldo name	Regulator File node name
DCDC1	axp81x_dcdc1	regulator.1
DCDC2	axp81x_dcdc2	regulator.2
DCDC3	axp81x_dcdc3	regulator.3
DCDC4	axp81x_dcdc4	regulator.4
DCDC5	axp81x_dcdc5	regulator.5
DCDC6	axp81x_dcdc6	regulator.6
DCDC7	axp81x_dcdc7	regulator.7
RTC-VCC	axp81x_rtc	regulator.8
ALDO1	axp81x_aldo1	regulator.9
ALDO2	axp81x_aldo2	regulator.10
ALDO3	axp81x_aldo3	regulator.11
DLDO1	axp81x_dldo1	regulator.12
DLDO2	axp81x_dldo2	regulator.13
DLDO3	axp81x_dldo3	regulator.14
DLDO4	axp81x_dldo4	regulator.15
ELDO1	axp81x_eldo1	regulator.16
ELDO2	axp81x_eldo2	regulator.17
ELDO3	axp81x_eldo3	regulator.18
FLDO1	axp81x_fldo1	regulator.19
FLDO2	axp81x_fldo2	regulator.20
GPIO0/LDO	axp81x_ldoio0	regulator.21
GPIO1/LDO	axp81x_ldoio1	regulator.22
SWOUT	axp81x_dc1sw	regulator.23

4.1.2. demo

For example , if the dcdc2 power supply to the cpu cluster1. And add the "vdd-cpu" supply id in ldo2 of regulator tree.

The dcdc2's maximum output voltage is 1.3V. We set the dcdc2's voltage to 1V.

```
#include <linux/regulator/consumer.h>
```

```
struct regulator *regu= NULL;
```

```
int ret = 0;
```

```
regu= regulator_get(NULL, "vdd-cpu");
```

```

if (IS_ERR(regu)) {
    pr_err("%s: some error happen, fail to get regulator \n", __func__);
    goto exit;
}

//set output voltage to 1V
ret = regulator_set_voltage(regu, 1000000, 1300000);
if (0 != ret) {
    pr_err("%s: some error happen, fail to set regulator voltage!\n", __func__);
    goto exit;
}

//enalbe regulator
ret = regulator_enable(regu);
if (0 != ret) {
    pr_err("%s: some error happen, fail to enable regulator!\n", __func__);
    goto exit;
}

//disalbe regulator
ret = regulator_disable(regu);
if (0 != ret) {
    pr_err("%s: some error happen, fail to disable regulator!\n", __func__);
    goto exit;
}

//put regulator, when module exit
regulator_put(regu);

```

4.1.3. Dump file node

In the serial shell, enter the command.

```
cat /sys/class/regulator/dump
```

It will print all regulator information.

```

axp81x_dclsw : disabled  0  1600000  supply_name:
axp81x_gpio1ldo : disabled  0  3300000  supply_name:
axp81x_gpio0ldo : enabled  1  3000000  supply_name: vcc-ctp
axp81x_fldo2 : enabled  0  900000  supply_name:
axp81x_fldo1 : disabled  0  1250000  supply_name:
axp81x_eldo3 : disabled  0  1800000  supply_name:
axp81x_dldo4 : disabled  0  2800000  supply_name:
axp81x_eldo1 : disabled  0  1200000  supply_name:
axp81x_eldo2 : enabled  1  1800000  supply_name: vcc_dsi

```



```

axp81x_dldo3 : disabled  0  2800000  supply_name:
axp81x_dldo2 : enabled   1  4200000  supply_name: vcc-lcd-0
axp81x_dldo1 : enabled   3  2800000  supply_name: vcc-io2 vcc-wifi-io vcc-wifi-io
axp81x_aldo3 : enabled   2  3000000  supply_name: vcc-avcc
axp81x_aldo2 : enabled   3  1800000  supply_name: vdd-codec-cp vdd-codec-ldoin
axp81x_aldo1 : enabled   0  1800000  supply_name:
axp81x_rtc   : disabled  0  3000000  supply_name:
axp81x_dcdc7 : disabled  0  1000000  supply_name:
axp81x_dcdc6 : enabled   0  900000   supply_name:
axp81x_dcdc5 : enabled   0  1200000  supply_name:
axp81x_dcdc4 : enabled   1  900000   supply_name: vdd-gpu
axp81x_dcdc3 : disabled  0  840000   supply_name:
axp81x_dcdc2 : enabled   0  840000   supply_name:
axp81x_dcdc1 : enabled   6  3000000  supply_name: vcc-io1 vcc-io vcc-io vcc-io vcc-emmc

```

The first column is regulator name, such as "axp81x_dcdc1".

The second column is regulator state, : 0 : disable, 1: enable.

The third column is the value of the regulator use_count, which explain the count of enable.

The fourth column is the voltage value of regulator.

The fifth column is the supply_name, which records all the supply id in enable state.

Notice : Sometimes the use_count is more than the one the number of supply id in enable state. For example, the use_count value of axp81x_dcdc1 is 6, but the number of supply id is 5. The reason is the ldo/dcdc is the system power.

4.2. Regulator shell

AXP regulator shell commands which can enable disable the regulator and can set the output voltage. Each file node is created in the directory /sys/bus/platform/devices.

```

reg-81x-cs-ald01  reg-81x-cs-ald02  reg-81x-cs-ald03  reg-81x-cs-dcdc1  reg-81x-cs-dcdc2
reg-81x-cs-dcdc3  reg-81x-cs-dcdc4  reg-81x-cs-dcdc5  reg-81x-cs-dcdc6  reg-81x-cs-dcdc7
reg-81x-cs-dldo1  reg-81x-cs-dldo2  reg-81x-cs-dldo3  reg-81x-cs-dldo4  reg-81x-cs-eldo1
reg-81x-cs-eldo2  reg-81x-cs-eldo3  reg-81x-cs-fldo1  reg-81x-cs-fldo2  reg-81x-cs-gpio0ldo
reg-81x-cs-gpio1ldo  reg-81x-cs-rtc

```

The AXP81X ALDO1's maximum output voltage is 3.3V. We set the AXP81X ALDO1's voltage to 3V.

//enable and set output voltage to 3.0V

```
echo 3300000 > /sys/bus/platform/devices/reg-81x-cs-ald01/max_microvolts
```

```
echo 3000000 > /sys/bus/platform/devices/reg-81x-cs-ald01/min_microvolts
```

//close the regulator

```
echo 3300000 > /sys/bus/platform/devices/reg-81x-cs-ald01/max_microvolts
```

```
echo 3000000 > /sys/bus/platform/devices/reg-81x-cs-ald01/min_microvolts
```

```
echo 0 > /sys/bus/platform/devices/reg-81x-cs-ald01/min_microvolts
```

When set output voltage after, again set output voltage to 0V, which can close the regulator.

Notice : Sometimes shell commands can not set the regulator's output voltage, because the regulator voltage is set by `regulator_set_voltage` function in code.

Confidential

5. Battery supply

5.1. Battery information

Refer to section 6.3, open the debug information of the battery information management part .

System will print the battery information in serial shell.

Battery information is as follows.

```

charger->ic_temp = 38 //PMIC temperature
charger->bat_temp = 30 //battery temperature
charger->vbat = 3801 //battery voltage, unit mV
charger->ibat = 553 //battery current, unit mA
charger->ocv = 3874 //OCV voltage, unit mV
charger->disvbat = 3801
charger->disibat = 553
power_sply = 2101 mW //battery power supply
charger->rest_vol = 59 //battery capacity
Axp Rdc = 131
Axp batt_max_cap = 2961
Axp coulumb_counter = 1751
Axp REG_B8 = e0
Axp OCV_percentage = 61
Axp Coulumb_percentage = 59
charger->is_on = 0 //battery charge statge , 1 : charge , 0 : not charge
charger->bat_current_direction = 0 //battery current direction , 1 : in , 0 : out
charger->charge_on = 1 //charge enable state , 1 : enable , 0 : disable
charger->ext_valid = 0
pmu_runtime_chgcur = 1000000 //the charge current when system is running, unit uA
pmu_earlysuspend_chgcur = 1000000 //the charge current when system is earlysuspend, unit uA
pmu_suspend_chgcur = 1600000 //the charge current when system is suspend, unit uA
pmu_shutdown_chgcur = 1600000 //the charge current when system is shutdown unit uA

```

5.2. Sysfs file node

5.2.1. MFD

functon	attribute	path	value
Modify one Register	rw	/sys/bus/platform/devices/axp81x_board/axp81x_reg	none
View multiple registers	r	/sys/bus/platform/devices/axp81x_board/axp81x_regs	none

Refer to section 6.3, access to the two file nodes, we can read and write axp813 register.

5.2.2. Power_supply (standard):

function	attribute	path	value
Battery capacity	r	/sys/class/power_supply/battery/capacity	the percentage , 0\1\2\.....\100
Battery current	r	/sys/class/power_supply/battery/current_now	Unit uA
Battery health state	r	/sys/class/power_supply/battery/health	Unknown, "Good", "Overheat", "Dead", "Over voltage", "Unspecified failure", "Cold"
Battery online	r	/sys/class/power_supply/battery/online	0\1 : not supply\supply
Battery present	r	/sys/class/power_supply/battery/present	0\1 : present\not present
Battery charge status	r	/sys/class/power_supply/battery/status	Unknown, "Charging", "Discharging", "Not charging", "Full"
Battery type	r	/sys/class/power_supply/battery/technology	Unknown, "NiMH", "Li-ion", "Li-poly", "LiFe", "NiCd", "LiMn"
The time of battery online	r	/sys/class/power_supply/battery/time_to_empty_now	Unit min
The time of battery charge	r	/sys/class/power_supply/battery/time_to_full_now	Unit min
Device type	r	/sys/class/power_supply/battery/type	Battery, "Mains", "USB"
Battery max voltage	r	/sys/class/power_supply/battery/voltage_max_design	Unit uV
Battery min voltage	r	/sys/class/power_supply/battery/voltage_min_design	Unit uV
Battery voltage	r	/sys/class/power_supply/battery/voltage_now	Unit uV
Battery max capacity	r	/sys/class/power_supply/battery/charge_full_design	Unit mAh
DC present	r	/sys/class/power_supply/ac/present	0\1 : plug out\plug in
DC online	r	/sys/class/power_supply/ac/online	0\1 : not supply\supply
Device type	r	/sys/class/power_supply/ac/	"battery", "Mains", "USB"

		type	
DC current	r	/sys/class/power_supply/ac/ current_now	Unit uA
DC voltage	r	/sys/class/power_supply/ac/ voltage_now	Unit uV
USB present	r	/sys/class/power_supply/us b/present	0\1 : plug out\plug in
USB online	r	/sys/class/power_supply/us b/online	0\1 : not supply\supply
Device type	r	/sys/class/power_supply/us b/type	"battery", "Mains", "USB"
USB current	r	/sys/class/power_supply/us b/current_now	Unit uA
USB voltage	r	/sys/class/power_supply/us b/voltage_now	Unit uV

5.2.3. Power_supply (non standard):

functon	attrib ute	path	value
ADC sampling frequency	rw	/sys/class/power_supply/bat tery/adcfreq	100\200\400\800,Unit Hz
The time of constant current charge timeout	rw	/sys/class/power_supply/bat tery/chgcsttimemin	360\480\600\720,Unit min
The time of the trickle charge timeout	rw	/sys/class/power_supply/bat tery/chgpretimemin	40\50\60\70,Unit min
The limit of charge current	rw	/sys/class/power_supply/bat tery/chgintmicrocur	300000\450000\..... 2550000,Unit uA
Charge enable	rw	/sys/class/power_supply/bat tery/chgen	0\1 : close\open
charge end condition	rw	/sys/class/power_supply/bat tery/chgendcur	charging current 's the percentage of constant current value. Support : 10/20
The voltage of charge target	rw	/sys/class/power_supply/bat tery/chgmicrovol	4100000\4220000\4200000\4400 000 , Unit uV
VBUS current limit enable	rw	/sys/class/power_supply/bat tery/iholden	0\1 : close\open
VBUS VHOLD voltage limit enable	rw	/sys/class/power_supply/bat tery/vholden	0\1 : close\open
VBUS current limit value	rw	/sys/class/power_supply/bat tery/ihold	500000\900000\not limit , Unit uA

VBUS voltage limit value	rw	/sys/class/power_supply/bat tery/vhold	4000000\4100000\..... 4700000 , Unit mV
-----------------------------	----	---	--

Confidential

6. Others

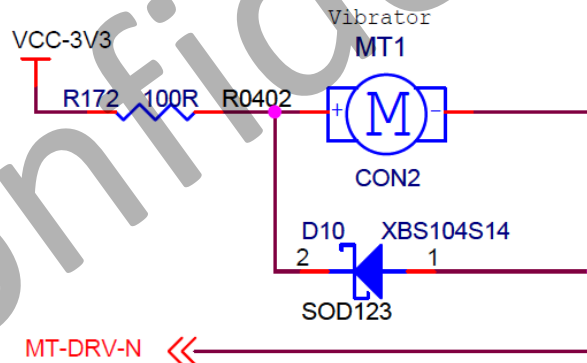
6.1. GPIO

6.1.1. AXP GPIO list

name	Configuration name	GPIO group name	axp81x PIN No.	IO status
axp81x GPIO0	power0	GPIO_AXP(0)		IO
axp81x GPIO1	power1	GPIO_AXP(1)		IO
axp81x DC1SW	power2	GPIO_AXP(2)		O
axp81x CHGLED	power3	GPIO_AXP(3)		O
axp81x N_VBUSEN	power4	GPIO_AXP(4)		O

6.1.2. AXP GPIO demo

For example, the power3 CHGLED gpio is the control gpio of the moto.



MT-DRV-N connects to the power3 CHGLED gpio. When the power3 CHGLED gpio is low level, the moto will vibrate. When the power3 CHGLED gpio is high level, the moto will close.

First, config the moto's gpio in sysconfig.

```

;-----
;motor configuration
;-----
[motor_para]
motor_used          = 1
motor_shake         = port:power3<1><default><default><1>

```

The first column, power3 is CHGLED gpio's configuration name.

The second column, 1/0: output/input.

The fifth column, 1/0: high level output/ low level output °

The moto code, as follows.

```
struct gpio_config motor_gpio;
int vibe_off;
script_item_u val;
script_item_value_type_e type;

/* sysconfig script analysis */
type = script_get_item("motor_para", "motor_shake", &val);
if(SCIRPT_ITEM_VALUE_TYPE_PIO != type) {
    printk(KERN_ERR "no motor_shake, ignore it!");
} else {
    motor_gpio = val.gpio;
    vibe_off = val.gpio.data;
}

/* request GPIO , and set default value*/
if (0 != motor_gpio.gpio) {
    if(0 != gpio_request(motor_gpio.gpio, "vibe")) {
        printk(KERN_ERR "ERROR: vibe Gpio_request is failed\n");
    }
    gpio_direction_output(motor_gpio.gpio, vibe_off);
}

/* output high level or low level */
if (0 != motor_gpio.gpio) {
    if(on) {
        __gpio_set_value(motor_gpio.gpio, !vibe_off);
    } else {
        __gpio_set_value(motor_gpio.gpio, vibe_off);
    }
}

/* free GPIO */
if (0 != motor_gpio.gpio) {
    gpio_free(motor_gpio.gpio);
}
```

6.2. AXP813 Register Shell

Write the value 0xff to the axp813 0x0f register:

```
echo 0f55 > /sys/bus/platform/devices/axp81x_board/axp81x_reg
```

Read the value from the axp813 0x0f register:

```
echo 0f > /sys/bus/platform/devices/axp81x_board/axp81x_reg
cat /sys/bus/platform/devices/axp81x_board/axp81x_reg
```

Read a plurality of registers:


```
echo 0f > /sys/bus/platform/devices/axp81x_board/axp81x_regs  
cat /sys/bus/platform/devices/axp81x_board/axp81x_regs
```

One can read 20 consecutive register from the beginning 0x0f register.

6.3. Debug and Print

Debug information control.

```
echo 1 > /sys/class/axppower/axpdebug //open the debug of the battery management.  
echo 2 > /sys/class/axppower/axpdebug //open the debug of the regulator.  
echo 4 > /sys/class/axppower/axpdebug //open the debug of the irq.  
echo 8 > /sys/class/axppower/axpdebug //open the debug of the charge.  
echo 0 > /sys/class/axppower/axpdebug //close all the debug information.
```

Confidential