

AW881XX Android Driver

INFORMATION

Driver File	aw881xx.c, aw881xx.h, aw881xx_reg.h, aw881xx_monitor.c, aw881xx_monitor.h, aw881xx_cali.c, aw881xx_cali.h
Smart PA	aw88164/aw88194/aw88194a/aw88195
I ² C Address	0x34/0x35/0x36/0x37
ADB Debug	yes
Platform	sp9820e

PROJECT CONFIG

```
#add aw881xx smartpa
```

在 kernel/sprd-diffconfig/sharkle/arm/sp9820e_5c10_kaios_diff_config 添加

```
ADD:CONFIG_SND_SMARTPA_AW881XX
```

AUIDO DEVICE

kcontrol

在

device/sprd/sharkle/sp9820e_5c10/rootdir/system/etc/audio_params/audio_hw.xml 中添加

```
<!-- scene: 0 - music; 1 - call. -->
<ext_codec name="aw881xx_smartpa">
  <config scene="0">
    <ctl name="aw881xx_mode_switch" val="spk"/>
  </config>
  <config scene="1">
    <ctl name="aw881xx_mode_switch" val="spk"/>
  </config>

  <!-- if the ext codec shared the i2s with bt sco -->
  <ap_i2s_shared shared="1"/>
</ext_codec>
```

Board Config

在 device/sprd/sharkle/sp9820e_5c10/BoardConfig.mk 中添加

```
# AW smart pa
AUDIO_SMART_PA_TYPE_AWINIC := AWINIC
```

Firmware update

aw881xx bin 文件命名格式如下：aw881xx_pid_num_scene_type.bin，其中 scene 代表场景，awinic 提供了 spk/rcv/fw/fm 四种场景；type 为 bin 文件的类型，共有 reg/fm/cfg 三种类型；num 为 product id，共有 01 和 03 两种，具体参考下图：（注意，type 类型为 reg 的 bin 文件 num 为 xx）

product	Pid
AW88194	01
AW88194A	03
AW88195	03

在 device/sprd/sharkle/sp9820e_5c10/sp9820e_5c10_base.mk 中添加 bin 文件的路径和名字：这里只以 product id 为 01，场景为 spk 的 3 个对应 bin 文件为例：

```
ifeq ($(strip $(AUDIO_SMART_PA_TYPE_AWINIC)), AWINIC)
PRODUCT_COPY_FILES += \

$(ROOTDIR)/system/etc/audio_params/aw881xx_pid_xx_spk_reg.bin:system/vendor/firmware/aw881xx_pid_xx_spk_reg.bin \

$(ROOTDIR)/system/etc/audio_params/aw881xx_pid_01_spk_fw.bin:system/vendor/firmware/aw881xx_pid_01_spk_fw.bin \

$(ROOTDIR)/system/etc/audio_params/aw881xx_pid_01_spk_cfg.bin:system/vendor/firmware/aw881xx_pid_01_spk_cfg.bin

endif
```

KERNEL DRIVER

AW881 XX Smart PA Driver

1. 修改 dts

打开 kernel/arch/arm/boot/dts/sp9820e-5c10-native.dts 文件，添加 aw881xx 的配置

```
&sound_sprd_ap_alliis {
    status = "okay";
    sprd-audio-card,dai-link@6 {
        plat {
            sound-dai = <&sprd_pcm_iis>;
        };
        cpu {
            sound-dai = <&i2s0>;
        };
        codec {
            sound-dai = <&aw881xx-aif 0>;
        };
    };
    sprd-audio-card,dai-link@7 {
        plat {
            sound-dai = <0>;
        };
    };
}
```

```
};
cpu {
    // Voice through aw smart pa(a dummy dai)
    sound-dai = <&vaudio_iis 2>;
};
codec {
    sound-dai = <&aw881xx-aif 1>;
};
};
};
```

2. 添加驱动文件

在 kernel/sound/soc/codecs/awinic 目录下添加 aw881xx 驱动文件 aw881xx.c, aw881xx.h, aw881xx_reg.h, aw881xx_monitor.c, aw881xx_monitor.h, aw881xx_cali.c, aw881xx_cali.h

3. 更新 Kconfig 和 Makefile

1) 在 kernel/sound/soc/codecs/Kconfig 中添加

```
config SND_SMARTPA_AW881XX
    tristate "SoC Audio for awinic aw881xx series"
    depends on I2C
    help
        This option enables support for aw881xx series Smart PA.
```

2) 在 kernel/sound/soc/codecs/Makefile 中添加

```
#for AWINIC AW881XX Smart PA
obj-$(CONFIG_SND_SMARTPA_AW881XX) += awinic/aw881xx.o awinic/aw881xx_monitor.o
awinic/cali.o
```

4. 添加 aw881xx fw&cfg 文件

1) 在 kernel/drivers/base/firmware_class.c 中添加 bin 文件目录，目录由系统决定，一般目录为

/system/vendor/firmware 或 /system/etc/firmware

```
static const char * const fw_path[] = {
    fw_path_para,
    "/system/vendor/firmware",
    "/system/etc/firmware",
    "/lib/firmware/updates/" UTS_RELEASE,
    "/lib/firmware/updates",
    "/lib/firmware/" UTS_RELEASE,
    "/lib/firmware"
};
```

2) 使用 adb 将 config 文件 push 到手机 vendor/firmware 中

SmartPA 选择

在 kernel/sound/soc/sprd/Kconfig 添加 AW881XX 选择

```
config SND_SOC_SPRD_I2S_SMARTPA
    tristate "ASoC Spreadtrum sound card with AP I2S and smartpa"
    select SND_SOC_SPRD_I2S
    select SND_SOC_SPRD_AUDIO_TWO_STAGE_DMAENGINE_SUPPORT
    select SPRD_AUDIO_MEMORY_MANAGER
    select SND_SMARTPA_AW881XX
    help
```

Say Y or M here if you want to add support for ASoC audio on Spreadtrum boards using the AP I2S and awinic smartpa

SPEAKER CALI

AW881XX 需要产线 speaker 校准，在 speaker 的测试符合要求后，需要将值写入到 AP 的 persist 分区中，然后在开机初次配置 dsp 的时候，将校准值写入到 aw881xx 中，完成 speaker 校准。

AW881XX 工作是需要 I2S 信号的，因此产线的 speaker 校准可以在工厂模式测试 speaker 声音的时候操作。

AW881XX 的校准是通过 HAL 层的命令直接控制的，因此需要使用 aw881xx_cali 可执行文件。

参数：

File	Cmd	Device_name	I2C bus	I2C Addr	Param	Description
aw881xx_cali	start_cali	aw881xx_smartpa	6	34		Re/F0 校准
aw881xx_cali	re_cali	aw881xx_smartpa	6	34		Re 校准
aw881xx_cali	f0_cali	aw881xx_smartpa	6	34		F0 获取
aw881xx_cali	store_re	aw881xx_smartpa	6	34	8.000	存储 Re
aw881xx_cali	store_fixed_re	aw881xx_smartpa	6	34	8000	存储定点化 Re

操作步骤：

1. 正常播放静音文件；

2. 执行 AW881XX 的校准，参考命令

```
./system/bin/aw881xx_cali start_cali <device name> <i2c bus> <i2c addr>
```

示例：

```
./system/bin/aw881xx_cali start_cali aw881xx_smartpa 6 34
[AWINIC] · [INF] · main: · firmware · version: · v1.0.0 ·
[AWINIC] · [INF] · aw881xx_cali_get_re: · re=8.212402 ·
[AWINIC] · [INF] · aw881xx_cali_get_f0: · f0=875.965820 ·
```

3. 由 AP 读取 re 和 f0，校准后的 Re 和 f0 存储 Node 中，可以用于判别是否符合范围。（为了保证 Re 的精度需求，Node 的 Re 放大了 1000 倍）。

Node	Path
Re	sys/bus/i2c/drivers/<device name>/<i2c bus>-<i2c addr>/re
F0	sys/bus/i2c/drivers/<device name>/<i2c bus>-<i2c addr>/f0

4. AP 判断 re 和 f0 是否符合要求，如果 re 和 f0 在设置范围内，表示 speaker 测试正常，需要将 re 的数据写到 AP 的 NVRAM 或者 persist，参考命令

```
./system/bin/aw881xx_cali store_re <device name> <i2c bus> <i2c addr> <re>
./system/bin/aw881xx_cali store_re aw881xx_smartpa 6 34 8.000

./system/bin/aw881xx_cali start_fixed_re <device name> <i2c bus> <i2c addr> <re_ste>
./system/bin/aw881xx_cali start_fixed_re aw881xx_smartpa 6 34 8000
```

注：Re 的值需要存储在手机的 NV 或者 persist，因此，需要在 Driver 中实现函数：

```
aw881xx_set_re_to_phone();
aw881xx_get_re_from_phone();
```

可执行文件使用：

除了直接在命令行调用 aw881xx_cali 可执行文件外，Awinic 也提供了调用 aw881xx_cali 可执行文件的示例代码 aw881xx_call_exe.c，在该文件中，需要设置或获取的变量如下：

变量名	意义	模式
-----	----	----

exe_name	可执行文件的名称，默认值为 aw881xx_cali	任何命令均需设置
cmd_type	命令类型，如 start_cali 等	
i2c_bus	i2c 总线	
i2c_addr	i2c 地址	
dev_name	设备名称，默认为 aw881xx_smartpa	
re	电阻值	store_re 命令需要设置
fixed_re	定点化电阻值	store_fixed_re 命令需要设置
f0		任何命令均不需要设置

DEBUG INTERFACE

Node

AW881XX Driver 会创建 cali/ update/dsp_rw/dsp/dsp_fw_ver/reg/rw/monitor 8 个设备节点文件，路径是 sys/bus/i2c/driver/aw881xx_smartpa/*-00xx，其中*为 i2c bus number，xx 为 i2c address。
可以使用 adb 配置 cali/ update/dsp_rw/dsp/dsp_fw_ver/reg/rw/monitor 参数调试 aw881xx。

reg

用于读写 AW881XX 的所有寄存器。

节点使用：

读寄存器值：cat reg

写寄存器值：echo reg_addr reg_data > reg （16 进制操作）

参考例程：

cat reg

echo 0x04 0x0241 > reg （向 0x04 寄存器写值 0x0241）

rw

用于读写 AW881XX 的单个寄存器。

节点使用：

读寄存器值：

echo reg_addr > rw （16 进制操作）

cat rw

写寄存器值：

echo reg_addr reg_data > rw （16 进制操作）

参考例程：

echo 0x04 > rw （读取 0x04 寄存器值）

cat rw

echo 0x04 0x0241 > rw (向 0x04 寄存器写值 0x0241)

update

用于读写 AW881XX 的 register 和 dsp 的配置。

节点使用:

更新 register 和 dsp 寄存器值:

echo 1 > update

dsp

用于读写 AW881XX 的 dsp 的配置。

节点使用:

读 dsp 寄存器值:

cat dsp

更新 dsp 寄存器值:

echo 1 > dsp

dsp_rw

用于读写 AW881XX 的 dsp 的单个寄存器。

节点使用:

读寄存器值:

echo reg_addr > dsp_rw

(16 进制操作)

cat rw

写寄存器值:

echo reg_addr reg_data > dsp_rw (16 进制操作)

参考例程:

echo 0x8601 > dsp_rw

(读取 dsp 的 0x8604 寄存器值)

cat dsp_rw

echo 0x8604 0x4011 > dsp_rw

(向 dsp 的 0x8604 寄存器写值 0x4011)

cali

用于配置 AW881XX 的 re 校准。

节点使用:

更新校准 re

echo re > cali

读取校准 re

cat cali

monitor

用于低温低压保护的开关

节点使用:

开关 monitor
echo 1 > monitor
echo 0 > monitor
读取 monitor 状态
cat monitor

dsp_fw_ver

用于查看 dsp fw 版本号

节点使用:

cat dsp_fw_ver

re

校准结束后查看校准的 re 值

节点使用:

cat re

f0

校准结束后查看校准的 f0 值

节点使用:

cat f0

Kcontrol

aw881xx_mode_switch

模式选择

tinymix aw881xx_mode_switch spk	选择 speaker 模式
tinymix aw881xx_mode_switch voice	选择 voice 模式
tinymix aw881xx_mode_switch fm	选择 fm 模式
tinymix aw881xx_mode_switch rcv	选择 receiver 模式
tinymix aw881xx_mode_switch off	选择 off 模式

aw881xx_rx_volume

音量配置，每步为 0.5dB，范围：[0~-96dB]

tinymix aw881xx_rx_volume 0 选择 0dB

tinymix aw881xx_rx_volume 6 选择-3dB