Contents

1 Overview .............................................................................................................................................. 3
  1.1 Sound card schematic ................................................................................................................ 3
  1.2 Static configuration .................................................................................................................. 3
  1.3 Dynamic configuration ............................................................................................................ 4
2 STM32MP15 evaluation board sound card configuration ............................................................ 5
  2.1 Sound card overview ............................................................................................................... 5
  2.2 Static configuration .................................................................................................................. 6
  2.3 Dynamic configuration ............................................................................................................ 10
    2.3.1 Wolfson wm8994 output configuration ........................................................................... 10
    2.3.2 Wolfson wm8994 input configuration ........................................................................... 11
    2.3.3 SPDFIRX input configuration ......................................................................................... 11
3 STM32MP15 disco board sound card configuration ................................................................. 13
  3.1 Sound card overview ............................................................................................................... 13
  3.2 Static configuration .................................................................................................................. 13
  3.3 Dynamic configuration ............................................................................................................ 15
    3.3.1 Cirrus cs42l51 output configuration ............................................................................ 16
    3.3.2 Cirrus cs42l51 input configuration .............................................................................. 16
4 References ........................................................................................................................................ 17
1 Overview

This article explains how to configure ST audio peripherals, as well as STM32MP1 boards external audio components, when they are assigned to the Linux® OS. In such cases, they are controlled by the ALSA framework.

In the ASoC layer of the ALSA framework, audio hardware components are described as CPU DAIs and codec, which are linked together to create DAI links. A sound card is a software component gathering a set of DAI links.

Each of the following STM32 MPU board sections describes one or more sound cards. A schematic for each sound card is provided, as well as its means of static and dynamic configuration.

1.1 Sound card schematic

The sound card schematic gives an overview of the hardware and software components forming the sound card, and their relationships.

The example sound card schematic given below emphasizes the links between the sound card and the device-tree section.

1.2 Static configuration

- Device tree
The device tree allows the description, configuration and connection of the audio hardware components to define the sound card. The user has to follow the audio graph card bindings\(^1\) to configure the sound card and device graph bindings\(^2\) to connect audio components. The user must also refer to the audio component (codec and CPU DAI) bindings to configure these components properly. The bindings of the audio components can be found in device-tree samples in following sections and in the References chapter.

The STMicroelectronics configuration tool STM32CubeMX, allows the generation of CPU DAI device tree nodes.

1.3 **Dynamic configuration**

The codecs and CPU DAI drivers also provide ALSA controls, allowing dynamic configuration of the sound card. The controls can be changed at runtime through the `amixer` utility to modify certain settings in the audio path. For instance, such controls can be used to modify the audio volume or the mute state of a block in the codec.

A customized configuration of these controls can be saved in the asound.state configuration file using the `alsactl` utility. This configuration can be restored at boot time through `alsactl`. STM32MPU sound cards come with a dedicated asound.state configuration file providing relevant control settings.
2 STM32MP15 evaluation board sound card configuration

2.1 Sound card overview

STM32MP15 Evaluation board sound card
2.2 Static configuration

The extract below is from the STM32MP15 evaluation board device tree. Only the nodes associated to the sound card, and the most relevant properties are shown here. For example, the properties linking nodes to form the first DAI link are emphasized with green font.

```c
/ * SOUND CARD */
sound {
    compatible = "audio-graph-card[1]";
    label = "STM32MP1-EV";
    /* Sound card identified as STM32MP1EV in ALSA */
    routing =
        "AIF1CLK", "MCLK1",
        "AIF2CLK", "MCLK1",
        "IN1LN", "MICBIAS2",
        "DMIC2DAT", "MICBIAS1",
        "DMIC1DAT", "MICBIAS1";
    dais = <&sai2a_port &sai2b_port &sai4a_port &spdifrx_port
        &dfsdm0_port &dfsdm1_port &dfsdm2_port &dfsdm3_port>;
};
/ * CODECS */
spdif_out: spdif-out {
    compatible = "linux,spdif-dit[4]";
    spdif_out_port: port@0 {
        spdif_out_endpoint: endpoint {
            remote-endpoint = <&sai4a_endpoint>;
        };
    };
};
spdif_in: spdif-in {
    compatible = "linux,spdif-dir[5]";
    spdif_in_port: port@0 {
        spdif_in_endpoint: endpoint {
            remote-endpoint = <&spdifrx_endpoint>;
        };
    };
};
dmic0: dmic@0 {
    compatible = "dmic-codec";
    port {
        dmic0_endpoint: endpoint {
            remote-endpoint = <&dfsdm_endpoint0>;
        };
    };
};
dmic1: dmic@1 {
    compatible = "dmic-codec";
    port {
        dmic1_endpoint: endpoint {
            remote-endpoint = <&dfsdm_endpoint1>;
        };
    };
};
dmic2: dmic@2 {
    compatible = "dmic-codec";
};
```
port {
    dmic2_endpoint: endpoint {
        remote-endpoint = <&dfsdm_endpoint2>;
    };
};

dmic3: dmic@3 {
    compatible = "dmic-codec";
    port {
        dmic3_endpoint: endpoint {
            remote-endpoint = <&dfsdm_endpoint3>;
        };
    };
};

&i2c2 { 
    wm8994: wm8994@1b {
        compatible = "wlf,wm8994";
        clocks = <&sai2a>;
        clock-names = "MCLK1";
        ports {
            #address-cells = <1>;
            #size-cells = <0>;
            wm8994_tx_port: port@0 {
                wm8994_tx_endpoint: endpoint {
                    remote-endpoint = <&sai2a_endpoint>;
                };
            };
            wm8994_rx_port: port@1 {
                wm8994_rx_endpoint: endpoint {
                    remote-endpoint = <&sai2b_endpoint>;
                };
            };
        }
    };
}

/* CPU DAIS */
&sai2 { 
    clocks = <&rcc SAI2>, <&rcc PLL3_Q>, <&rcc PLL4_Q>;
    pinctrl-names = "default", "sleep";
    pinctrl-0 = <&sai2a_pins_a>, <&sai2b_pins_a>;
    pinctrl-1 = <&sai2a_sleep_pins_a>, <&sai2b_sleep_pins_a>;
    clock-names = "pclk", "x8k", "x11k";
    sai2a: audio-controller@4400b004 {
        compatible = "st,stm32-sai-sub-a";
       dma-names = "tx"; /* SAI set as transmitter */
        clocks = <&rcc SAI2 K>;
        clock-names = "sai_ck";
        sai2a_port: port@0 {
            sai2a_endpoint: endpoint {
                remote-endpoint = <&wm8994_tx_endpoint>;
                format = "i2s";
                mclk-fs = <256>; /* SAI is master clock */
            };
        };
    };
}
sai2b: audio-controller@44000b024 {
    compatible = "st,stm32-sai-sub-b";
    dma-names = "rx";     /* SAI set as receiver */
    clocks = <&rcc SAI2_K>, <&sai2a>;
    clock-names = "sai_ck", "MCLK";
}

sai2b_port: port@0 {
    sai2b_endpoint: endpoint {
        remote-endpoint = <&wm8994_rx_endpoint>;
        format = "i2s";
        mclk-fs = <256>;
    } /* SAI is master clock provider */
};

&sai4 {
    clocks = <&rcc SAI4>, <&rcc PLL3 Q>, <&rcc PLL4 Q>;
    clock-names = "pclk", "x8k", "x11k";
}

sai4: audio-controller@50027004 {
    compatible = "st,stm32-sai-sub-a";
    dma-names = "tx";
    st,iec60958;         /* SAI configured for S/PDIF protocol*/
    pinctrl-names = "default", "sleep";
    pinctrl-0 = <&sai4a_pins_a>;
    pinctrl-1 = <&sai4a_sleep_pins_a>;
    clocks = <&rcc SAI4_K>;
    clock-names = "sai_ck";
}

sai4a_port: port@0 {
    sai4a_endpoint: endpoint {
        remote-endpoint = <&spdif_out_endpoint>;
    };
};

&spdifrx {
    compatible = "st,stm32h7-spdifrx";
    pinctrl-names = "default", "sleep";
    pinctrl-0 = <&spdifrx_pins_a>;
    pinctrl-1 = <&spdifrx_sleep_pins_a>;
}

spdifrx_port: port@0 {
    spdifrx_endpoint: endpoint {
        remote-endpoint = <&spdif_in_endpoint>;
    };
};

&dfsdm {
    compatible = "st,stm32mp1-dfsdm";
    pinctrl-names = "default", "sleep";
    pinctrl-0 = <&dfsdm_clkout_pins_a>
        &dfsdm_data1_pins_a &dfsdm_data3_pins_a>;
    pinctrl-1 = <&dfsdm_clkout_sleep_pins_a>
        &dfsdm_data1_sleep_pins_a &dfsdm_data3_sleep_pins_a>;
    spi-max-frequency = <2048000>;
    clocks = <&rcc DFSDM_K>, <&rcc ADFSDM_K>;
    clock-names = "dfsdm", "audio";
}

dfsdm0: filter@0 {
    compatible = "st,stm32-dfsdm-dmic";
st,adc-channels = <3>; /* Use channel 3 fed by mic U1 signal wired to input 3 */
st,adc-channel-names = "dmic_u1"; /* Free name used to reference associated mic U1 */
st,adc-channel-types = "SPI_R"; /* mic U1 signal available on input 3 Rising edge */
st,adc-channel-clk-src = "CLKOUT"; /* CKOUT clocks the microphones */
st,filter-order = <3>;

asoc_pdm0: dfsdm-dai {
    compatible = "st,stm32h7-dfsdm-dai";
    #sound-dai-cells = <0>;
    io-channels = <&dfsdm0 0>;
    cpu_port0: port {
        dfsdm_endpoint0: endpoint {
            remote-endpoint = <&dmic0_endpoint>;
        }
    }
};

dfsdm1: filter@1 {
    compatible = "st,stm32-dfsdm-dmic";
    st,adc-channels = <0>; /* Use channel 0 fed by mic U2 signal wired to input 1 */
    st,adc-alt-channel = <1>; /* Connect channel 0 to next input (input 1) */
    st,adc-channel-names = "dmic_u2"; /* Free name used to reference associated mic U2 */
    st,adc-channel-types = "SPI_F"; /* mic U2 signal available on input 1 Falling edge */
    st,adc-channel-clk-src = "CLKOUT";
    st,filter-order = <3>;

    asoc_pdm1: dfsdm-dai {
        compatible = "st,stm32h7-dfsdm-dai";
        #sound-dai-cells = <0>;
        io-channels = <&dfsdm1 0>;
        cpu_port1: port {
            dfsdm_endpoint1: endpoint {
                remote-endpoint = <&dmic1_endpoint>;
            }
        }
    }
};

dfsdm2: filter@2 {
    compatible = "st,stm32-dfsdm-dmic";
    st,adc-channels = <2>; /* Use channel 2 fed by mic U3 signal wired to input 3 */
    st,adc-alt-channel = <1>; /* Connect channel 2 to next input (input 3) */
    st,adc-channel-names = "dmic_u3"; /* Free name used to reference associated Dmic U3 */
    st,adc-channel-types = "SPI_F"; /* mic U3 signal available on input 3 Falling edge */
    st,adc-channel-clk-src = "CLKOUT";
    st,filter-order = <3>;

    asoc_pdm2: dfsdm-dai {
        compatible = "st,stm32h7-dfsdm-dai";
        #sound-dai-cells = <0>;
        io-channels = <&dfsdm2 0>;
        cpu_port2: port {
            dfsdm_endpoint2: endpoint {
                remote-endpoint = <&dmic2_endpoint>;
            }
        }
    }
}
### 2.3 Dynamic configuration

The table below gives an overview of the controls allowing configuration of the STM32MPU evaluation board "sound" sound card.

<table>
<thead>
<tr>
<th>audio device</th>
<th>CPU DAI</th>
<th>codec</th>
</tr>
</thead>
<tbody>
<tr>
<td>playback_code</td>
<td>no controls available</td>
<td>configure codec output path</td>
</tr>
<tr>
<td>recordCodec</td>
<td>no controls available</td>
<td>configure codec input path</td>
</tr>
<tr>
<td>playback_spdif</td>
<td>configure iec958</td>
<td>no controls available</td>
</tr>
<tr>
<td>record_spdif</td>
<td>configure SPDFIRX input path</td>
<td>no controls available</td>
</tr>
</tbody>
</table>

#### 2.3.1 Wolfson wm8994 output configuration

Control commands to configure aif1 interface to headset output (HPOUT1L/R) path, on wm8994 codec:
Soundcard configuration

Control commands to configure aif1 interface to speaker output (SPKOUTL/RP) path, on wm8994 codec:

```
amixer -c STM32MP1EV cset name='AIF1DAC1 Volume' '96' '96'
amixer -c STM32MP1EV cset name='DAC1L Mixer AIF1.1 Switch' 'on'
amixer -c STM32MP1EV cset name='DAC1R Mixer AIF1.1 Switch' 'on'
amixer -c STM32MP1EV cset name='DAC1 Switch' 'on','on'
amixer -c STM32MP1EV cset name='DAC1 Volume' '96','96'
amixer -c STM32MP1EV cset name='SPKL DAC1 Volume' '50' '50'
amixer -c STM32MP1EV cset name='SPKR DAC1 Volume' '50' '50'
amixer -c STM32MP1EV cset name='SPKL DAC1 Switch' 'on'
amixer -c STM32MP1EV cset name='SPKR DAC1 Switch' 'on'
amixer -c STM32MP1EV cset name='SPKL Output Switch' 'on'
amixer -c STM32MP1EV cset name='SPKR Output Switch' 'on'
amixer -c STM32MP1EV cset name='Speaker Mode' 'Class AB'
amixer -c STM32MP1EV cset name='Speaker Volume' '50' '50'
amixer -c STM32MP1EV cset name='Speaker Mixer Volume' '3'
amixer -c STM32MP1EV cset name='Speaker Reference' '0'
amixer -c STM32MP1EV cset name='Speaker Switch' 'on'
```

2.3.2 Wolfson wm8994 input configuration

Control commands to configure headset microphone input (IN1LN) to aif2 interface, on wm8994 codec:

```
amixer -c STM32MP1EV cset name='IN1L PGA IN1LN Switch' 'on'
amixer -c STM32MP1EV cset name='IN1L PGA IN1LP Switch' 'off'
amixer -c STM32MP1EV cset name='IN1L Volume' '25'
amixer -c STM32MP1EV cset name='IN1L Switch' 'on'
amixer -c STM32MP1EV cset name='MIXINL IN1L Switch' 'on'
amixer -c STM32MP1EV cset name='MIXINL IN1LN Volume' '1'
amixer -c STM32MP1EV cset name='MIXINL IN1LP Volume' '0'
amixer -c STM32MP1EV cset name='AIF1ADCL Source' 'Left'
amixer -c STM32MP1EV cset name='ADCL Mux' 'ADC'
amixer -c STM32MP1EV cset name='DAC2 Left Sidetone Volume' '12'
amixer -c STM32MP1EV cset name='DAC2 Right Sidetone Volume' '12'
amixer -c STM32MP1EV cset name='AIF2DAC2L Mixer Left Sidetone Switch' 'on'
amixer -c STM32MP1EV cset name='AIF2DAC2R Mixer Right Sidetone Switch' 'on'
amixer -c STM32MP1EV cset name='DAC2 Volume' '96' '96'
amixer -c STM32MP1EV cset name='DAC2 Switch' 'on','on'
amixer -c STM32MP1EV cset name='AIF2ADC Volume' '96' '96'
amixer -c STM32MP1EV cset name='AIF2ADC Mux' 'AIF2ADCDAT'
amixer -c STM32MP1EV cset name='AIF2 Boost Volume' '1'
amixer -c STM32MP1EV cset name='ADC OSR' 'Low Power'
```

2.3.3 SPDFIRX input configuration

Control commands to configure rx1 input path on SPDFIRX:

```
amixer -c STM32MP1EV cset name='SPDIFRX input' 1
```
3 STM32MP15 disco board sound card configuration

3.1 Sound card overview

The extract below is from the STM32MP15 disco board device tree. Only the nodes associated to the sound card, and the most relevant properties are shown here. As an example, the properties linking nodes to form the first DAI link are emphasized with green font.

```c
/ {
   /* SOUND_CARD */
   sound {
      compatible = "audio-graph-card";
      label = "STM32MP1-DK";
      /* Sound card identified as STM32MP1DK in ALSA */
      routing =
         "Playback", "MCLK",
         "Capture", "MCLK",
         "MICL", "Mic Bias";
      dais = <&sai2a_port &sai2b_port &i2s2_port>;
      status = "okay";
   };
}
/ *
   CODECS */
&i2c1 {
   cs42l51: cs42l51@4a {
   ...
```
compatible = "cirrus,cs42l51";
...
clocks = <&sai2a>;
clock-names = "MCLK";

cs42l51_port: port {
    #address-cells = <1>;
    #size-cells = <0>;

cs42l51_tx_endpoint: endpoint@0 {
    reg = <0>;
    remote-endpoint = <&sai2a_endpoint>;
    frame-master;
    /* codec is
    master */
    bitclock-master;
};
cs42l51_rx_endpoint: endpoint@1 {
    reg = <1>;
    remote-endpoint = <&sai2b_endpoint>;
    frame-master;
    /* codec is
    master */
    bitclock-master;
};
}

hdmi-transmitter@39 {
    compatible = "sil,sii9022";
    ...
    ports {
        #address-cells = <1>;
        #size-cells = <0>;
        port@0 {
            reg = <0>;
            sii9022_in: endpoint {
                remote-endpoint = <&ltdc_ep0_out>;
            };
        };
        port@1 {
            reg = <1>;
            sii9022_tx_endpoint: endpoint {
                remote-endpoint = <&i2s2_endpoint>;
            };
        };
    }
};

/* CPU DAIS */
&sai2 {
clocks = <&rcc SAI2>, <&rcc PLL3_Q>, <&rcc PLL3_Q>;
clock-names = "pclk", "x8k", "x11k";
pinctrl-names = "default", "sleep";
pinctrl-0 = <&sai2a_pins_a>, <&sai2b_pins_b>;
pinctrl-1 = <&sai2a_sleep_pins_a>, <&sai2b_sleep_pins_b>;
status = "okay";
sai2a: audio-controller@4400b004 {
    compatible = "st,stm32-sai-sub-a";
    #clock-cells = <0>;}
The card-specific alsa-lib configuration file for STMP32MP15 Disco board is /usr/share/alsa/cards/STM32MP1DK.conf.

### Dynamic configuration

The table below gives an overview of the controls allowing the configuration of the STM32MPU disco board sound card.

<table>
<thead>
<tr>
<th>audio device</th>
<th>CPU DAI</th>
<th>codec</th>
</tr>
</thead>
<tbody>
<tr>
<td>playback_code</td>
<td>no controls available</td>
<td>configure codec output path</td>
</tr>
<tr>
<td></td>
<td>no controls</td>
<td></td>
</tr>
</tbody>
</table>
### Soundcard configuration

<table>
<thead>
<tr>
<th>audio device</th>
<th>CPU DAI</th>
<th>codec</th>
</tr>
</thead>
<tbody>
<tr>
<td>record_codec</td>
<td>available</td>
<td>configure codec input path</td>
</tr>
<tr>
<td>playback_hdmi</td>
<td>no controls</td>
<td>no controls available</td>
</tr>
</tbody>
</table>

#### 3.3.1 Cirrus cs42l51 output configuration

Control commands to configure the aif interface to headset output (AOUTA/B) path, on the cs42l51 codec:

```bash
amixer -c STM32MP1DK cset name='PCM Playback Switch' 'on','on'
amixer -c STM32MP1DK cset name='PCM Playback Volume' '63','63'
amixer -c STM32MP1DK cset name='Analog Playback Volume' '204','204'
amixer -c STM32MP1DK cset name='PCM channel mixer' 'L R'
```

#### 3.3.2 Cirrus cs42l51 input configuration

Control commands to configure headset microphone input (MICIN1/AIN3A) to the aif interface, on the cs42l51 codec:

```bash
amixer -c STM32MP1DK cset name='PGA-ADC Mux Left' '3'
amixer -c STM32MP1DK cset name='Mic Boost Volume' '1','1'
```
4 References

- 1.01.1 Documentation/devicetree/bindings/sound/audio-graph-card.txt
- Documentation/devicetree/bindings/graph.txt
- 3.03.13.2 asound.conf
- Documentation/devicetree/bindings/sound/spdif-transmitter.txt
- Documentation/devicetree/bindings/sound/spdif-receiver.txt