In i.MX8MQ and i.MX8M Mini, the codec used is WM8524, which only supports audio playback. Although 8M Mini does have PDM microphone interface (MICFIL), there is no support for audio record via I2S. This guide will show you how to add audio recording driver in i.MX8MQ&8MM step by step.

Hardware: i.MX8MQ/8MM Evk, I2S output digital microphone
OS: Android/Linux
Kernel version: 4.14.78

The driver code is based on wm8524, and the new driver is named "micarray". Take i.MX8MM Android for example.

Overall modification in kernel:

```
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in
working directory)

        modified:   arch/arm64/boot/dts/freescale/fsl-imx8mm-
evk.dts
        modified:   arch/arm64/configs/android_defconfig
        modified:   sound/soc/codecs/Kconfig
        modified:   sound/soc/codecs/Makefile
        modified:   sound/soc/fsl/Kconfig
        modified:   sound/soc/fsl/Makefile
        modified:   sound/soc/fsl/fsl_sai.c

Untracked files:
  (use "git add <file>..." to include in what will be
committed)

        sound/soc/codecs/micarray.c
        sound/soc/fsl/imx-micarray.c
```

1. Modify arch/arm64/boot/dts/freescale/fsl-imx8mm-evk.dts
   Add “micarray” node and comment “wm8524”:

```
@@ -107,6 +107,23 @@
                };
        };

+       micarray: micarray {
+        compatible = "fsl,micarray";
+        clocks = <&clk IMX8MM_CLK_SAI1_ROOT>;
+        clock-names = "mclk";
+        wlf,mute-gpios = <&gpio5 21 GPIO_ACTIVE_LOW>;
+    };
+
+    sound-micarray {
+        compatible = "fsl,imx-audio-micarray";
+        model = "micarray-audio";
+        audio-cpu = <&sai1>;
+        audio-codec = <&micarray>;
+        audio-routing =
+                "Line Out Jack", "LINEVOUTL",
+                "Line Out Jack", "LINEVOUTR";
+    };
+/*
        wm8524: wm8524 {
                compatible = "wlf,wm8524";
                clocks = <&clk IMX8MM_CLK_SAI3_ROOT>;
@@ -123,13 +140,14 @@
                    "Line Out Jack", "LINEVOUTL",
                    "Line Out Jack", "LINEVOUTR";
        };
-
+*/
```

Modify SAI1 configuration for recording:

```
@@ -295,6 +314,12 @@
MX8MM_IOMUXC_SAI1_TXD5_SAI1_TX_DATA5    0xd6
MX8MM_IOMUXC_SAI1_TXD6_SAI1_TX_DATA6    0xd6
MX8MM_IOMUXC_SAI1_TXD7_SAI1_TX_DATA7    0xd6
+MX8MM_IOMUXC_SAI1_RXFS_SAI1_RX_SYNC     0xd6
+MX8MM_IOMUXC_SAI1_RXC_SAI1_RX_BCLK      0xd6
+MX8MM_IOMUXC_SAI1_RXD0_SAI1_RX_DATA0        0xd6
+MX8MM_IOMUXC_SAI1_RXD1_SAI1_RX_DATA1        0xd6
+MX8MM_IOMUXC_SAI1_RXD2_SAI1_RX_DATA2        0xd6
+MX8MM_IOMUXC_SAI1_RXD3_SAI1_RX_DATA3        0xd6
                        >;
                };
@@ -868,15 +899,15 @@
        assigned-clocks = <&clk IMX8MM_CLK_SAI1_SRC>,
                          <&clk IMX8MM_CLK_SAI1_DIV>;
        assigned-clock-parents = <&clk IMX8MM_AUDIO_PLL1_OUT>;
-       assigned-clock-rates = <0>, <49152000>;
+       assigned-clock-rates = <0>, <24576000>;
        clocks = <&clk IMX8MM_CLK_SAI1_IPG>, <&clk
IMX8MM_CLK_DUMMY>,
                 <&clk IMX8MM_CLK_SAI1_ROOT>, <&clk
IMX8MM_CLK_DUMMY>,
                 <&clk IMX8MM_CLK_DUMMY>, <&clk
IMX8MM_AUDIO_PLL1_OUT>,
                 <&clk IMX8MM_AUDIO_PLL2_OUT>;
        clock-names = "bus", "mclk0", "mclk1", "mclk2", "mclk3",
"pll8k", "pll11k";
-       fsl,sai-multi-lane;
-       fsl,dataline,dsd = <0 0xff 0xff 2 0xff 0x11>;
-       dmas = <&sdma2 0 26 0>, <&sdma2 1 26 0>;
+       //fsl,sai-multi-lane;
+       //fsl,dataline,dsd = <0 0xff 0xff 2 0xff 0x11>;
+       //dmas = <&sdma2 0 26 0>, <&sdma2 1 26 0>;
        status = "okay";
 };
```

2. Add codec driver

Make a copy of sound/soc/codec/wm8524.c and rename to micarray.c, add capture in dai driver:

```
-      #define WM8524_RATES SNDRV_PCM_RATE_8000_192000
-      #define WM8524_FORMATS (SNDRV_PCM_FMTBIT_S16_LE |
    SNDRV_PCM_FMTBIT_S24_LE |\
-      SNDRV_PCM_FMTBIT_S32_LE)
+        #define WM8524_RATES SNDRV_PCM_RATE_8000_192000
+        #define WM8524_FORMATS (SNDRV_PCM_FMTBIT_S16_LE |
SNDRV_PCM_FMTBIT_S24_LE |\
+        SNDRV_PCM_FMTBIT_S32_LE)

static struct snd_soc_dai_driver wm8524_dai = {
    .name = "micarray-hifi",
    .playback = {
        .stream_name = "Playback",
        .channels_min = 2,
        .channels_max = 8,
        .rates = MICARRAY_RATES,
        .formats = MICARRAY_FORMATS,
    },
+    .capture = {
+        .stream_name = "Capture",
+        .channels_min = 1,
+        .channels_max = 8,
+        .rates = SNDRV_PCM_RATE_KNOT,
+        .formats = MICARRAY_FORMATS,
+    },
    .ops = &wm8524_dai_ops,
};
```

Change driver name in micarray.c:

```
static const struct of_device_id wm8524_of_match[] = {
-        { .compatible = "wlf,wm8524" },
+        { .compatible = "fsl,micarray" },
{ /* sentinel*/ }
};
static struct platform_driver wm8524_codec_driver = {
    .probe      = wm8524_codec_probe,
    .remove     = wm8524_codec_remove,
    .driver     = {
-        .name   = "wm8524-codec",
+        .name   = "micarray-codec",
        .of_match_table = wm8524_of_match,
    },
};
```

3. Modify codec driver Kconfig and Makefile:

```
--- a/sound/soc/codecs/Kconfig
+++ b/sound/soc/codecs/Kconfig
@@ -181,6 +181,7 @@ config SND_SOC_ALL_CODECS
       select SND_SOC_WM8510 if SND_SOC_I2C_AND_SPI
       select SND_SOC_WM8523 if I2C
       select SND_SOC_WM8524 if GPIOLIB
+      select SND_SOC_MICARRAY if GPIOLIB
       select SND_SOC_WM8580 if I2C
       select SND_SOC_WM8711 if SND_SOC_I2C_AND_SPI
       select SND_SOC_WM8727
@@ -1022,6 +1023,10 @@ config SND_SOC_WM8524
       tristate "Wolfson Microelectronics WM8524 DAC"
       depends on GPIOLIB

+config SND_SOC_MICARRAY
+    tristate "MICARRAY CODEC"
+    depends on GPIOLIB
+
--- a/sound/soc/codecs/Makefile
+++ b/sound/soc/codecs/Makefile
@@ -192,6 +192,7 @@ snd-soc-wm8400-objs := wm8400.o
 snd-soc-wm8510-objs := wm8510.o
 snd-soc-wm8523-objs := wm8523.o
 snd-soc-wm8524-objs := wm8524.o
+snd-soc-micarray-objs := micarray.o
 snd-soc-wm8580-objs := wm8580.o
 snd-soc-wm8711-objs := wm8711.o
 snd-soc-wm8727-objs := wm8727.o
@@ -440,6 +441,7 @@ obj-$(CONFIG_SND_SOC_WM8400)       += snd-
soc-wm8400.o
 obj-$(CONFIG_SND_SOC_WM8510)   += snd-soc-wm8510.o
 obj-$(CONFIG_SND_SOC_WM8523)   += snd-soc-wm8523.o
 obj-$(CONFIG_SND_SOC_WM8524)   += snd-soc-wm8524.o
+obj-$(CONFIG_SND_SOC_MICARRAY)  += snd-soc-micarray.o
 obj-$(CONFIG_SND_SOC_WM8580)   += snd-soc-wm8580.o
```

4. Add machine driver

   Make a copy of sound/soc/fsl/imx-wm8524.c and rename to imx-micarray.c, add micarray rx hw params:

```
static int imx_aif_hw_params(struct snd_pcm_substream
*substream,
             struct snd_pcm_hw_params *params)
{
    struct snd_soc_pcm_runtime *rtd = substream->private_data;
    struct snd_soc_dai *cpu_dai = rtd->cpu_dai;
    struct snd_soc_dai *codec_dai = rtd->codec_dai;
    struct snd_soc_card *card = rtd->card;
    struct imx_priv *priv = snd_soc_card_get_drvdata(card);
    unsigned int channels = params_channels(params);
    struct device *dev = card->dev;
    unsigned int fmt;
    int ret = 0;

    fmt = SND_SOC_DAIFMT_I2S |
          SND_SOC_DAIFMT_NB_NF |
          SND_SOC_DAIFMT_CBS_CFS;

    ret = snd_soc_dai_set_fmt(cpu_dai, fmt);
    if (ret) {
       dev_err(dev, "failed to set cpu dai fmt: %d\n", ret);
       return ret;
    }
    priv->slots = 2;
    priv->slot_width = params_physical_width(params);

    ret = snd_soc_dai_set_tdm_slot(cpu_dai,
                BIT(channels) - 1, BIT(channels) - 1,
                priv->slots, priv->slot_width);
    if (ret) {
       dev_err(dev, "failed to set cpu dai tdm slot: %d\n",
ret);
       return ret;
    }

    return ret;
}
```

Add micarray rx startup check:

```
static int imx_aif_startup(struct snd_pcm_substream *substream)
{
    struct snd_pcm_runtime *runtime = substream->runtime;
    struct snd_soc_pcm_runtime *rtd = substream->private_data;
    struct snd_soc_card *card = rtd->card;
    struct imx_priv *priv = snd_soc_card_get_drvdata(card);

    static struct snd_pcm_hw_constraint_list constraint_rates;
    static struct snd_pcm_hw_constraint_list
constraint_channels;
    int ret;

    constraint_rates.list = micarray_rates;
    constraint_rates.count = ARRAY_SIZE(micarray_rates);

    ret = snd_pcm_hw_constraint_list(runtime, 0,
SNDRV_PCM_HW_PARAM_RATE,
                     &constraint_rates);
    if (ret)
        return ret;

    constraint_channels.list = micarray_channels;
    constraint_channels.count = ARRAY_SIZE(micarray_channels);
    ret = snd_pcm_hw_constraint_list(runtime, 0,
SNDRV_PCM_HW_PARAM_CHANNELS,
                         &constraint_channels);
    if (ret)
        return ret;

    return 0;
}

static struct snd_soc_ops imx_aif_ops = {
    .hw_params = imx_aif_hw_params,
    .startup = imx_aif_startup,
};
```

```
static struct snd_soc_dai_link imx_wm8524_dai[] = {
    {
        .name = "HiFi-tx",
        .stream_name = "HiFi-tx",
-       .codec_dai_name = "wm8524-hifi",
+       .codec_dai_name = "micarray-hifi",
        .ops = &imx_hifi_ops,
    },
+    {
+        .name = "HiFi-rx",
+        .stream_name = "HiFi-rx",
+        .codec_name = "snd-soc-dummy",
+        .codec_dai_name = "snd-soc-dummy-dai",
+        .ops = &imx_aif_ops,
+    },
};

static int imx_wm8524_probe(struct platform_device *pdev)
{
    imx_wm8524_dai[0].codec_of_node = codec_np;
    imx_wm8524_dai[0].cpu_dai_name = dev_name(&cpu_pdev->dev);
    imx_wm8524_dai[0].platform_of_node = cpu_np;
    imx_wm8524_dai[0].playback_only = 1;

+    imx_wm8524_dai[1].cpu_dai_name = dev_name(&cpu_pdev->dev);
+    imx_wm8524_dai[1].platform_of_node = cpu_np;
+    imx_wm8524_dai[1].capture_only  = 1;

    priv->card.late_probe = imx_wm8524_late_probe;
-    priv->card.num_links = 1;
+    priv->card.num_links = 2;
    priv->card.dev = &pdev->dev;
    priv->card.owner = THIS_MODULE;
}

static const struct of_device_id imx_wm8524_dt_ids[] = {
-     { .compatible = "fsl,imx-audio-wm8524", },
+     { .compatible = "fsl,imx-audio-micarray", },
};

static struct platform_driver imx_wm8524_driver = {
       .driver = {
-     .name = "imx-wm8524",
+     .name = "imx-micarray",
     .pm = &snd_soc_pm_ops,
}
```

5. Modify machine driver Kconfig and Makefile:

```
--- a/sound/soc/fsl/Kconfig
+++ b/sound/soc/fsl/Kconfig
@@ -350,6 +350,20 @@ config SND_SOC_IMX_WM8524
         Say Y if you want to add support for SoC audio on an
i.MX board with
         a wm8524 codec.

+config SND_SOC_IMX_MICARRAY
+    tristate "SoC Audio support for i.MX boards with micarray"
+    depends on OF && I2C
+    select SND_SOC_MICARRAY
+    select SND_SOC_IMX_PCM_DMA
+    select SND_SOC_FSL_SAI
+    select SND_SOC_FSL_UTILS
+    select SND_KCTL_JACK
+    help
+      SoC Audio support for i.MX boards with MICARRAY
+      Say Y if you want to add support for SoC audio on an i.MX
board with
+      a micarray codec.
+
 config SND_SOC_IMX_SII902X
       tristate "SoC Audio support for i.MX boards with
sii902x"
       depends on OF && I2C
--- a/sound/soc/fsl/Makefile
+++ b/sound/soc/fsl/Makefile
@@ -77,6 +77,7 @@ snd-soc-imx-sgtl5000-objs := imx-sgtl5000.o
 snd-soc-imx-wm8958-objs := imx-wm8958.o
 snd-soc-imx-wm8960-objs := imx-wm8960.o
 snd-soc-imx-wm8524-objs := imx-wm8524.o
+snd-soc-imx-micarray-objs := imx-micarray.o
 snd-soc-imx-wm8962-objs := imx-wm8962.o
 snd-soc-imx-xtor-objs := imx-xtor.o
 snd-soc-imx-sii902x-objs := imx-sii902x.o
@@ -105,6 +106,7 @@ obj-$(CONFIG_SND_SOC_IMX_SGTL5000) += snd-
soc-imx-sgtl5000.o
 obj-${CONFIG_SND_SOC_IMX_WM8958} += snd-soc-imx-wm8958.o
 obj-$(CONFIG_SND_SOC_IMX_WM8960) += snd-soc-imx-wm8960.o
 obj-$(CONFIG_SND_SOC_IMX_WM8524) += snd-soc-imx-wm8524.o
+obj-$(CONFIG_SND_SOC_IMX_MICARRAY) += snd-soc-imx-micarray.o
 obj-$(CONFIG_SND_SOC_IMX_WM8962) += snd-soc-imx-wm8962.o
 obj-$(CONFIG_SND_SOC_IMX_XTOR) += snd-soc-imx-xtor.o
 obj-$(CONFIG_SND_SOC_IMX_RPMSG) += snd-soc-imx-rpmsg.o
```

6. Modify platform driver

```
--- a/sound/soc/fsl/fsl_sai.c
+++ b/sound/soc/fsl/fsl_sai.c
@@ -1466,7 +1466,7 @@ static int fsl_sai_probe(struct
platform_device *pdev)
        sai->synchronous[RX] = true;
        sai->synchronous[TX] = false;
        fsl_sai_dai.symmetric_rates = 1;
-       fsl_sai_dai.symmetric_channels = 1;
+       fsl_sai_dai.symmetric_channels = 0;
        fsl_sai_dai.symmetric_samplebits = 1;

        if (of_find_property(np, "fsl,sai-synchronous-rx", NULL)
&&
```

7. Modify android defconfig: (only apply for Android)

```
--- a/arch/arm64/configs/android_defconfig
+++ b/arch/arm64/configs/android_defconfig
@@ -656,6 +656,7 @@ CONFIG_SND_SOC_IMX_AK5558=y
 CONFIG_SND_SOC_IMX_AK4497=y
 CONFIG_SND_SOC_IMX_WM8960=y
 CONFIG_SND_SOC_IMX_WM8524=y
+CONFIG_SND_SOC_IMX_MICARRAY=y
 CONFIG_SND_SOC_IMX_CS42888=y
 CONFIG_SND_SOC_IMX_WM8962=y
 CONFIG_SND_SOC_IMX_MICFIL=y
```

8. Rebuild and flash the image, you will have a sound card name “micarrayaudio” with both playback and recording capability.

```
# cat /proc/asound/card0/id
Micarrayaudio
# cat /proc/asound/card0/
id     pcm0p/  pcm1c/
```

For playback, the path is same with wm8524; for capture, you’ll need to connect I2S digital microphone to SAI1 via audio connector.

To test the audio record:

```
# tinycap test.wav -D 0 -d 1 -c 2 -r 48000 -b 32 -T 5
Capturing sample: 2 ch, 48000 hz, 32 bit
Captured 241664 frames
```