



I2S Linux driver



A quality version of this page, approved on *21 January 2020*, was based off this revision.

Contents

1 Article purpose	3
2 Short Description	4
3 Configuration	5
3.1 Kernel Configuration	5
3.2 Device tree	5
4 How to use	6
5 How to trace and debug	7
6 Source code location	8
7 References	9



1 Article purpose

This article introduces the I2S Linux[®] driver for the SPI/I2S internal peripheral.



2 Short Description

The I2S Linux driver is an ASoC CPU DAI driver implemented in the Linux ALSA framework.

The I2S driver requests two clocks which can be used as parent clocks of the I2S peripheral kernel clock. The rates of these parent clocks must be respectively a multiple of 8kHz and 11.025kHz. One of these two clocks is selected at run-time through the I2S clock mux, depending on audio stream sampling rate.

Restriction on STM32MP15x lines

SPI/I2S2 and SPI/I2S3 internal peripherals share the same clock mux. This may result in a conflict on clock mux configuration, if I2S2 and I2S3 are used to run audio streams having incompatible rates (i.e. rates not multiple of each other). The I2S driver does not protect by itself against such conflict.



3 Configuration

3.1 Kernel Configuration

Activate the I2S Linux driver in the kernel configuration using the Linux Menuconfig tool: [Menuconfig](#) or [how to configure kernel](#).

```
[*] Device Drivers
  [*] Sound card support
    [*] Advanced Linux Sound Architecture
      [*] ALSA for SoC audio support
        STMicroelectronics STM32 SOC audio support
          [*] STM32 I2S interface (SPI/I2S block) support
```

3.2 Device tree

Refer to the [I2S_device_tree_configuration](#) article when configuring the I2S Linux kernel driver.

Information

The I2S peripheral can be configured in transmit only, receive only or full-duplex mode, via its configuration register.

The I2S driver offers the support of transmit only, receive only, or full-duplex use cases. However, the driver always configure the I2S in full-duplex mode in the configuration register, whatever the chosen use case. If a path (transmit or capture) is not used, it is simply discarded. So, **there is no property in the I2S device tree bindings to configure the mode** of the I2S peripheral.



4 How to use

The I2S Linux driver can be accessed from userland through an ALSA device. Refer to [ALSA overview](#) for information on how to list and use ALSA devices.



5 How to trace and debug

The `debugfs` and `procfs` file system can be checked to get information on the I2S driver and the resources it uses. A non-exhaustive list of these file system entries is provided below. Refer to [ALSA overview](#) for more details on debugging tools.

- `debugfs` entries:
 - **asoc**: refer to [ALSA_overview#How_to_monitor](#)
 - **clk**: refer to [Clock_overview#How_to_monitor_with_debugfs](#) to get information on clocks.
 - **pinctrl**: refer to [Pinctrl_overview#How_to_monitor](#) to get information on pins.
 - **regmap**: allow to monitor SPI/I2S peripheral registers.

```
$ cat /sys/kernel/debug/regmap/xxx.audio-controller/registers
```

- `procfs` entries:
 - **asound**: refer to [ALSA_overview#How_to_debug](#)
 - **interrupts**: allow to check interrupts.

```
$ cat /proc/interrupts
```



6 Source code location

sound/soc/stm/stm32_i2s.c : implements the I2S Linux driver.



7 References

Integrated Interchip Sound

Linux[®] is a registered trademark of Linus Torvalds.

ALSA System on Chip

Central processing unit

Digital Audio Interface

Serial Peripheral Interface

Advanced Linux sound architecture

Process File System (See <https://en.wikipedia.org/wiki/Procfs> for more details)

Debug File System (See <https://en.wikipedia.org/wiki/Debugfs> for more details)