EC2x&AG35-Quecopen

I2S Interface Configuration

LTE Module Series

Rev. EC2x&AG35-Quecopen_I2S Interface Configuration_V1.2

Date: 2019-01-31

Status: Preliminary

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
7th Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://www.quectel.com/support/sales.htm

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm
Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2018-04-11	Grady QUAN	Initial
1.1	2018-08-28	Grady QUAN	Added AG35 pin
1.2	2019-1-31	Grady QUAN	Added the setup of 96k sample rate

Contents

Table Index

Figure Index

1 Introduction

This document mainly introduces how to configure I2S interface especially how to configure the pins and the interfaces from the user's development perspective to use I2S interface.

2 Hardware Design

2.1. Pins Assignment

2.1.1. The Primary PCM

Table 1: Multiplexing the Primary I2S Pins

Pin Name	Pin Number	Primary Function	Multiplexing Function 1	Multiplexing Function 2	Multiplexing Function 3	Multiplexing Function 4
SPI_CS_N	37(EC2x)/ 79(AG35)	SPI_CS_N _BLSP6	PCM_IN	I2S_IN	GPIO22	UART_RTS_BLSP6
SPI_MOSI	38(EC2x)/ 77(AG35)	SPI_MOSI _BLSP6	PCM_OUT	I2S_OUT	GPIO20	UART_TXD_BLSP6
SPI_MISO	39(EC2x)/ 78(AG35)	SPI_MISO _BLSP6	PCM_SYNC	I2S_WS	GPIO21	UART_RXD_BLSP6
SPI_CLK	40(EC2x)/ 80(AG35)	SPI_CLK_ BLSP6	PCM_CLK	I2S_CLK	GPIO23	UART_CTS_BLSP6

The primary I2S is used as SPI function by default, so it needs to configure the primary I2S pins to I2S function.

2.1.2. The Second I2S

Table 2: Multiplexing the Second I2S Pins

Pin Name	Pin Number	Primary Function	Multiplexing Function 1	Multiplexing Function 2
PCM_IN	24(EC2x)/ 66(AG35)	PCM_IN	I2S_IN	GPIO76
PCM_OUT	25(EC2x)/ 68(AG35)	PCM_OUT	I2S_OUT	GPIO77
PCM_SYNC	26(EC2x)/ 65(AG35)	PCM_SYNC	I2S_WS	GPIO79
PCM_CLK	27(EC2x)/ 67(AG35)	PCM_CLK	I2S_CLK	GPIO78

The second I2S is used as PCM function by default. I2S and PCM can be multiplexed simultaneously.

2.2. Recommended Circuit Design

The following figure is the reference design of PCM interface with an external Codec chip:

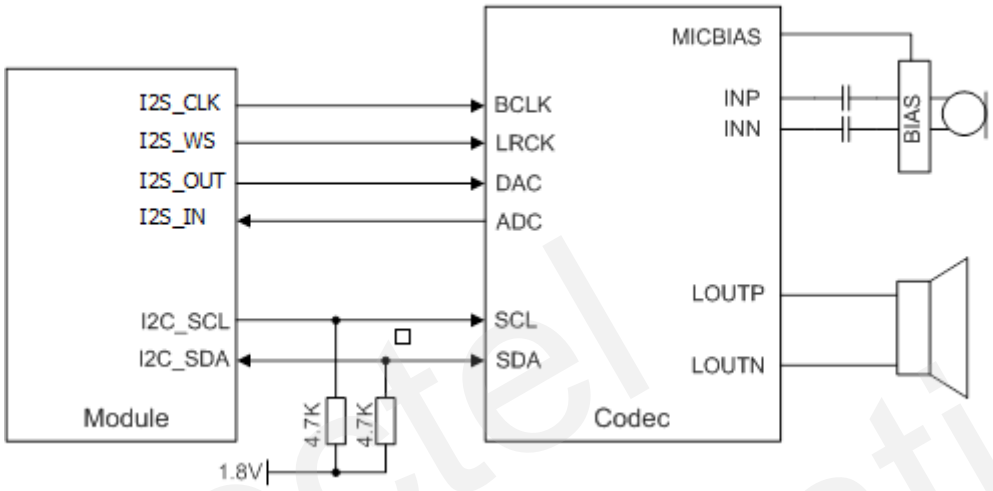

Figure 1：I2S Circuit Reference Design

It is suggested to reserve RC (R=22Ω, C=22pF) circuit on I2S signal lines, especially I2S_CLK.

3 I2S Timing Analysis

3.1. Different Time Slot Timings

I2S supports 48kHz and 96kHz sample rates but currently I2S does not support other sample rates. Figure 2 shows the timing of 48kHz sample rate.

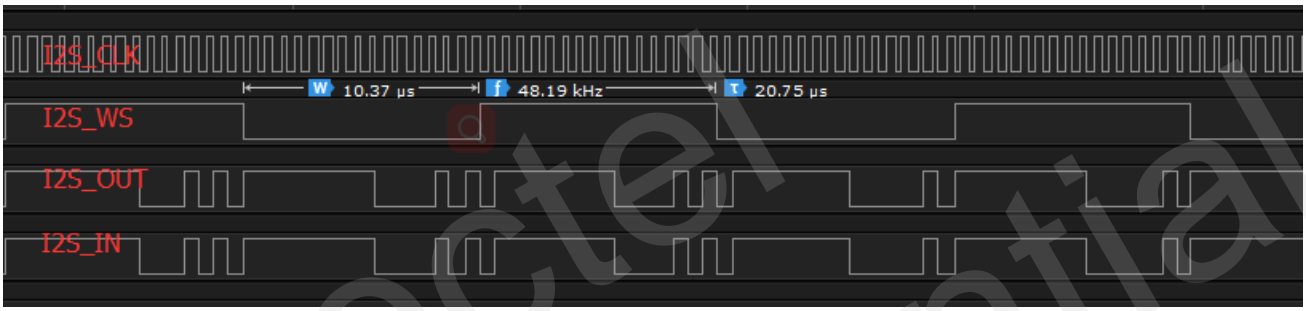

Figure 2: Timing of 48kHz Sample Rate

4 I2S Pin Configuration

4.1. mdm9607-mtp.dtsi

4.1.1. The Primary I2S

The primary I2S pins which are correspond to gpio20, gpio21, gpio22, and gpio23 are used as SPI6 when the codes are released on the platform. So please disable SPI6 in mdm9607-mtp.dtsi and enable mi2s_prim at the same time. The codes need to be modified as follows:

```
    &spi_6 {
-       status = "ok";
+       status = "disabled";
    };
    &mi2s_prim {
-       status = "disabled";
+       status = "ok";
    };
```

4.1.2. The Second I2S

The second I2S pins which are corresponded to gpio76, gpio77, gpio78 and gpio79 have been used as I2S function when the codes are released on the platform, so do not modify the codes.

4.2. mdm9607-pinctrl.dtsi

In the codes released on the platform, the pins corresponded to the primary and second I2S have been set up, so users do not need to modify anything. Here are part of the codes.

```
    pmx_pri_mi2s {
        pri_mi2s_ws_active: pri_mi2s_ws_active {
            mux {
                pins = "gpio20";
                function = "pri_mi2s_ws_a";
            };
```

```
            config {
                pins = "gpio20";
                drive-strength = <8>;     /* 8 MA */
                bias-disable;              /* No PULL */
                output-high;
            };
        };
        ……
    }
```

Here drive-strength stands for the drive strength of I/O pins, and bias-disable stands for I/O pins status and output-high means to output high level voltage.

4.3. mdm9607.dtsi

The description of some parameters of this file can be referred to qcom-audio-dev.txt in path: apps_proc/kernel/Documentation/devicetree/bindings/sound.

4.3.1. The Primary I2S

Open pinctrl corresponded to the primary I2S and modify the codes as follows:

```
    mi2s_prim: qcom,msm-dai-q6-mi2s-prim {
        ......
        qcom,msm-mi2s-tx-lines = <1>;
        pinctrl-names = "default", "idle";
        //comment out by cullen
-       /*pinctrl-0 = <&pri_mi2s_ws_active
+       pinctrl-0 = <&pri_mi2s_ws_active
                &pri_mi2s_sck_active
                &pri_mi2s_dout_active
                &pri_mi2s_din_active>;
        pinctrl-1 = <&pri_mi2s_ws_sleep
                &pri_mi2s_sck_sleep
                &pri_mi2s_dout_sleep
                &pri_mi2s_din_sleep>;
-       */
    };
```

4.3.2. The Second I2S

Open pinctrl corresponded to the second I2S and modify the codes as follows:

```
    mi2s_sec: qcom,msm-dai-q6-mi2s-sec {
        ……
        qcom,msm-dai-q6-mi2s-dev-id = <1>;
        qcom,msm-mi2s-rx-lines = <2>;
        qcom,msm-mi2s-tx-lines = <1>;
-       /*
        pinctrl-names = "default", "idle";
        pinctrl-0 = <&sec_mi2s_ws_active
                &sec_mi2s_sck_active
                &sec_mi2s_dout_active
                &sec_mi2s_din_active>;
        pinctrl-1 = <&sec_mi2s_ws_sleep
                &sec_mi2s_sck_sleep
                &sec_mi2s_dout_sleep
                &sec_mi2s_din_sleep>;
-       */
    };
```

4.4. I2S Pin Mode Verification

After modifying the device tree files, users can execute the command cat /sys/kernel/debug/gpio to verify whether the pins are in I2S mode.

```
~ # cat /sys/kernel/debug/gpio
GPIOs 0-79, platform/1000000.pinctrl, 1000000.pinctrl:
 gpio0   : out 1 2mA no pull
 gpio1   : out 1 2mA no pull
 gpio2   : out 1 2mA no pull
 gpio3   : out 1 2mA no pull
 gpio4   : out 2 2mA pull down
 gpio5   : in  2 2mA pull down
 gpio6   : in  0 2mA pull down
 gpio7   : in  0 2mA pull down
 gpio8   : in  0 2mA pull down
 gpio9   : in  0 2mA pull down
 gpio10  : in  0 2mA pull down
 gpio11  : in  0 2mA pull up
 gpio12  : in  0 2mA pull down
 gpio13  : in  0 2mA pull down
 gpio14  : in  0 2mA pull down
 gpio15  : in  0 2mA pull down
 gpio16  : in  0 2mA no pull
 gpio17  : in  0 2mA pull down
 gpio18  : in  0 2mA pull down
 gpio19  : in  0 2mA pull down
 gpio20  : out 3 8mA no pull
 gpio21  : in  3 8mA no pull
 gpio22  : out 3 8mA no pull
 gpio23  : out 3 8mA no pull
```

Figure 3：Pin Status 1

```
gpio54  : in  0 2mA pull down
gpio55  : out 0 2mA no pull
gpio56  : in  0 2mA no pull
gpio57  : out 0 2mA no pull
gpio58  : in  0 2mA pull down
gpio59  : in  0 2mA pull down
gpio60  : in  0 2mA pull down
gpio61  : in  0 2mA pull down
gpio62  : in  0 2mA pull down
gpio63  : in  0 2mA pull down
gpio64  : in  0 2mA pull down
gpio65  : in  0 2mA pull down
gpio66  : in  0 2mA pull down
gpio67  : in  0 2mA pull down
gpio68  : in  0 2mA pull down
gpio69  : in  0 2mA pull down
gpio70  : in  0 2mA pull down
gpio71  : in  0 2mA pull down
gpio72  : in  0 2mA pull down
gpio73  : in  0 2mA pull down
gpio74  : in  0 2mA pull down
gpio75  : out 0 2mA no pull
gpio76  : in  2 8mA no pull
gpio77  : out 2 8mA no pull
gpio78  : out 2 8mA no pull
gpio79  : out 1 8mA no pull
```

Figure 4：Pin Status 2

5 I2S Interface Configuration

5.1. mdm9607.dtsi

5.1.1. Master and Slave Modes

The primary and the second I2S interfaces default to be in master mode. If it is necessary to modify them into slave mode, please do as follows:

```
    sound{
            compatible = "qcom,mdm9607-audio-tomtom";
            ……
-           qcom,mi2s-interface-mode = "pri_mi2s_master", "sec_mi2s_master";
+           qcom,mi2s-interface-mode = "pri_mi2s_slave", "sec_mi2s_slave";
            qcom,auxpcm-interface-mode = "pri_pcm_master", "sec_pcm_master";
            ……
    }
```

“pri_mi2s_slave” shows that the primary I2S is in slave mode, and “sec_mi2s_slave” shows that the second I2S is in slave mode.

5.2. mdm9607.c

5.2.1. Sample Rate

I2S supports 48k and 96k sample rates. If it is necessary to change the sample rate, modify the codes as follows:

```
    static int mdm_mi2s_rx_ch = 1;
    static int mdm_mi2s_tx_ch = 1;
-   static int mdm_mi2s_rate = SAMPLE_RATE_48KHZ;
+   static int mdm_mi2s_rate = 96000;
    static int mdm_sec_mi2s_rx_ch = 1;
    static int mdm_sec_mi2s_tx_ch = 1;
-   static int mdm_sec_mi2s_rate = SAMPLE_RATE_48KHZ;
+   static int mdm_sec_mi2s_rate = 96000
```

In the codes “mdm_mi2s_rate” represents the sample rate of the primary I2S, and “mdm_sec_mi2s_rate” represents the sample rate of the second I2S.