CONFIDENTIAL

编程指南

FD6818

REV.1.0.0

目录

内容

CONFIDENTIAL

目录

CONFIDENTIAL

修订记录

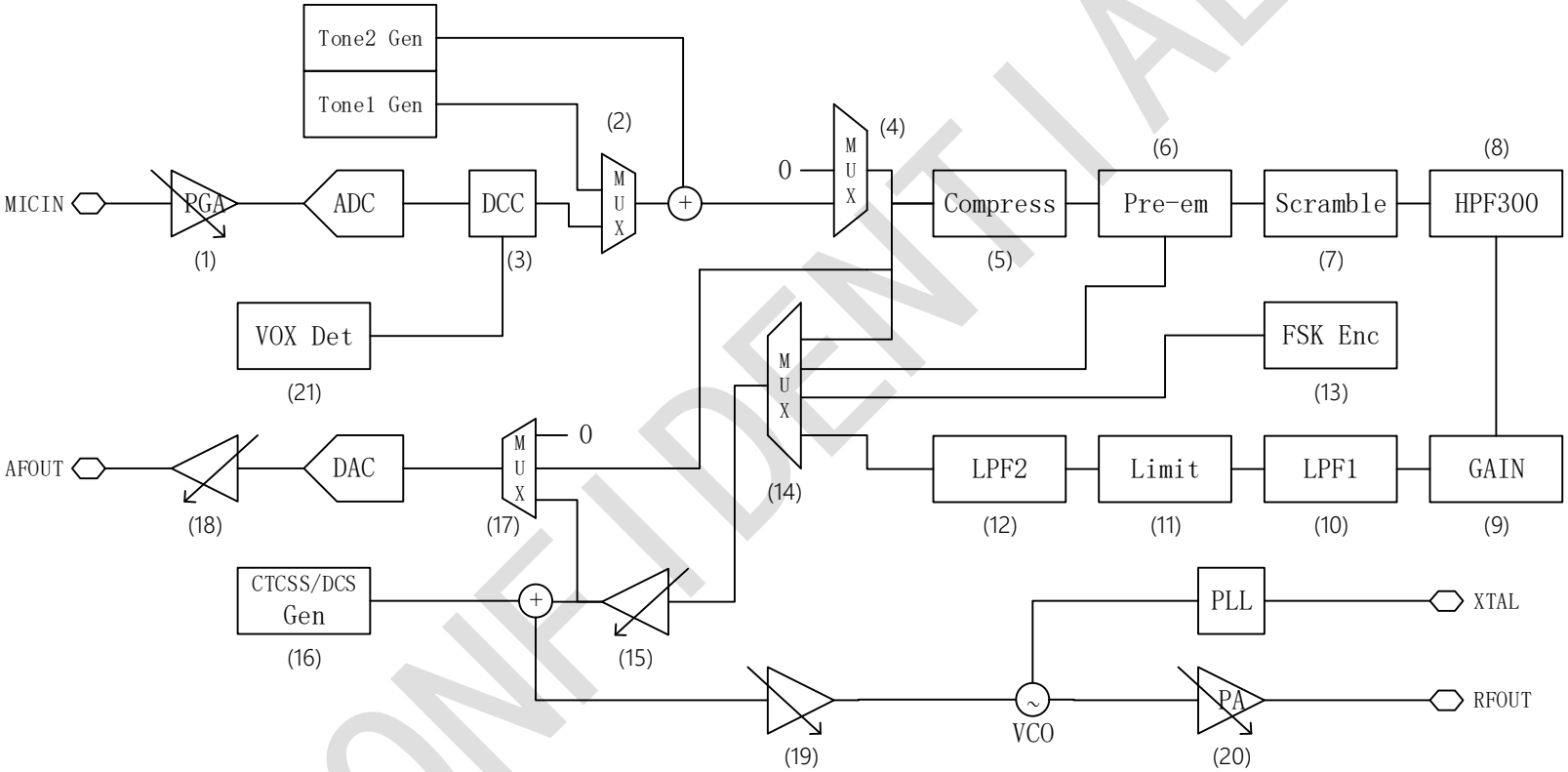
修订记录

修订日期	修订内容	修订位置
2018.07.24	宽窄带设置错误，删除对 REG_3AH 的设置	
2018.07.24	VCO 模式时，对 REG_03H 设置错误	VCO 模式设置
2018.09.06	修改 25k/12.5k 时的带宽设置	频段、频点、带宽模式设置
2018.09.07	第一次改版芯片，修改频段定义范围	频段、频点、带宽模式设置

CONFIDENTIAL

发射框图

发射框图



CONFIDENTIAL

发射框图

框图说明：

(1) MIC 通路 PGA 模拟增益

Register	Address(HEX)	Description
mic_pga_gain[4:0]	REG_18H[15:11]	Value Gain MIC_Sens(mV) 00000 = not valid 00001 = 4.6250 ~26.2 00010 = 6.1250 ~19.8 00011 = 10.6250 ~11.4 00100 = 9.1250 ~13.3 00101 = 13.6875 ~ 8.9 00110 = 15.1875 ~ 8.0 00111 = 19.6875 ~ 6.2 01000 = 12.1250 ~10.0(default) 01001 = 16.6875 ~ 7.3 01010 = 18.1875 ~ 6.7 01011 = 22.6250 ~ 5.4 01100 = 21.2500 ~ 5.7 01101 = 25.6250 ~ 4.7 01110 = 27.1250 ~ 4.5 01111 = 31.7500 ~ 3.8 10000 = 18.1875 ~ 6.7 10001 = 22.6875 ~ 5.3 10010 = 24.2500 ~ 5.0 10011 = 28.7500 ~ 4.2 10100 = 27.2500 ~ 4.4 10101 = 31.7500 ~ 3.8 10110 = 33.1250 ~ 3.7 10111 = 37.6250 ~ 3.2 11000 = 30.1875 ~ 4.0 11001 = 34.6250 ~ 3.5 11010 = 36.1250 ~ 3.4 11011 = 40.7500 ~ 3.0 11100 = 39.0000 ~ 3.1

CONFIDENTIAL

		11101 = 43.7500 ~ 2.8 11110 = 45.1250 ~ 2.7 11111 = 49.5625 ~ 2.4

上述 mic_pga_gain 的增益表的值，表征的是相对增益。修改此寄存器，将会影响 VOX 检测幅度，需要重新设置 VOX 门限。~~调节 MIC 灵敏度时，尽量不要修改此寄存器，只需修改 mic_sens_gain 寄存器，详细说明见文档“设置调制限制和 MIC 灵敏度”部分。~~

(2) TONE1、TONE2 生成

Register	Address(HEX)	Description
tone1_gen	REG_70H[15]	Enable TONE1
tone1_gain[6:0]	REG_70H[14:8]	TONE1 tuning gain
tone2_gen	REG_70H[7]	Enable TONE2
tone2_gain[6:0]	REG_70H[6:0]	TONE2 tuning gain
tone1_freq[15:0]	REG_71H[15:0]	TONE1 frequency· control word =freq(kHz)* 2^26/6500
tone2_freq[15:0]	REG_72H[15:0]	TONE2 frequency· control word =freq(kHz)* 2^26/6500

若 tone1_gen=1，MIC 输入将被旁路，同时生成单音 tone1。

若 tone1_gen=1, 且 tone2_gen=1 时，MIC 输入被旁路，同时生成双音 tone1+tone2。

(3) MIC 路 DC 滤波器

Register	Address(HEX)	Description
dcc_tx_bypass	REG_7EH[7]	1=MICIN DC Filter bypass
dcc_tx_bw[2:0]	REG_7EH[5:3]	MICIN DC Filter bandwidth(3dB) select 111=15Hz;110=30Hz;101=60Hz;100=120Hz; 011=240Hz;010=480Hz;

默认 dcc_tx_bypass=0，dc_tx_bw=111。用作 DMR/dPMR 模式时，可以将 dcc_tx_bypass=1 关掉此 DC 滤波器。修改发射频响时，可以用 dcc_tx_bw=011 或 010 对 300Hz 进行衰减。

(4) 发射通路选择 1

Register	Address(HEX)	Description
audio_tx_mute	REG_50H[15]	1=Audio Tx Mute

audio_tx_mute 可用于 DTMF 发射 symbol 间的 idle。

CONFIDENTIAL

(5) 发射语音压扩

详细说明见文档”语音模式设置”部分。

(6) 发射语音预加重

Register	Address(HEX)	Description
audio_emph_tx_bypass	REG_9BH[0]	1=PRE-EMPHASIS bypass

(7) 发射语音扰频

详细说明见文档”SCRAMBLE 模式设置”部分。

(8) 发射语音 300Hz 高通滤波器

Register	Address(HEX)	Description
audio_hpf_tx_bypass	REG_9BH[3]	1=Audio HPF 300Hz bypass for TX

~~(9) 发射语音增益 1（MIC 灵敏度调节）~~

~~Register~~	~~Address(HEX)~~	~~Description~~
~~mic_sens_gain[5:0]~~	~~REG_97H[13:8]~~	~~MIC sensitivity adjust gain 0=mute, 63=max, 0.5dB/step~~

~~**详细说明见文档“设置调制限制和 MIC 灵敏度”部分。**~~

(10) 发射语音低通滤波器 1

Register	Address(HEX)	Description
audio_lpf1_tx_bypass	REG_9BH[2]	1=Audio LPF1 bypass for TX

~~(11) 发射语音限幅~~

~~Register~~	~~Address(HEX)~~	~~Description~~
~~audio_tx_limit_bypass~~	~~REG_50H[10]~~	~~1=Audio Tx Limit bypass~~
~~audio_tx_limit[9:0]~~	~~REG_50H[9:0]~~	~~Audio Tx Limit Value~~

CONFIDENTIAL

发射框图

建议不要修改此寄存器，调制限制只需调节 dev_sh 和 dev_lvl 寄存器（详细说明见文档“设置调制限制和 MIC 灵敏度”部分）。

(12) 发射语音低通滤波器 2

Register	Address(HEX)	Description
audio_lpf2_tx_bypass	REG_9BH[1]	1=Audio LPF2 bypass for TX
audio_lpf2_tx_sel[1:0]	REG_43H[8:6]	Audio LPF bandwidth (Apass=1dB) for Tx 100 = 4.5 kHz 101 = 4.25 kHz 110 = 4 kHz 111 = 3.75 kHz 000 = 3 kHz 001 = 2.5 kHz 010 = 2 kHz 011 =1.7kHz

(13) 发射 FSK 编码

详细说明见“FSK 模式设置”部分。

(14) 发射通路选择 2

Register	Address(HEX)	Description
audio_tx_path_sel[1:0]	REG_9DH[3:2]	01=select pre-emphasis output 10=reserved 11=select LPF2 output

发射通路选择，可以选择其中几个节点的输出作发射。

~~(15) 发射语音增益 2~~

~~Register~~	~~Address(HEX)~~	~~Description~~
~~audio_tx_gain_sh [2:0]~~	~~REG_53H[12:10]~~	~~111=max,000=min~~ ~~0~~ ~~42dB, 6dB/step, digital gain~~
~~audio_tx_gain[4:0]~~	~~REG_53H[4:0]~~	~~0=min,31=max~~

CONFIDENTIAL

		~~-6~3.3dB, ~~=1dB/step, digital gain

~~建议不要修改此增益，因为此后语音要和亚音频作相加发射处理，如果此增益过高，同时亚音频调制很大时，会造成波形溢出。建议使用默认值或初始化推荐值。~~

(16) 发射亚音频生成

详细说明见文档“亚音频设置”部分。

(17) AF DAC 通路选择

Register	Address(HEX)	Description
afout_invert	REG_47H[13]	1 = invert afout
afout_mode[4:0]	REG_47H[12:8]	0x10 = MUTE 0x11 = RX AFOUT 0x18 = BEEP/TX Side Tone（CTCSS/CDCSS is not include） 0x15 = RX ALARM TONE

此寄存器收发复用，可用来设置静音、接收 AF 输出和侧音 BEEP 输出。

(18) AF DAC 模拟增益

Register	Address(HEX)	Description
afout_en	REG_03H[9]	1=Enable AFOUT DAC
dac_vgain[3:0]	REG_23H[3:0]	DAC maximum output level

无论是输出接收 AF 还是 BEEP 侧音，都需要设置 afout_enable=1 来打开 AF DAC。增益 dac_vgain 设置，建议使用最大值 1111（默认）。音量调节可以选用数字增益，详细说明见文档“接收静音（MUTE）及音量设置”部分。

(19) 发射调制限制

Register	Address(HEX)	Description
dev_en	REG_40H[12]	Enable FM deviation
dev_sh[3:0]	REG_40H[11:8]	**FM deviation coarse tuning,** **0000=max, 1111=min**

CONFIDENTIAL

dev_lvl[7:0]	REG_40H[7:0]	FM deviation fine tuning, 0000=min, 1111=max, **GAIN=(256+dev_lvl[7:0])>>dev_sh[3:0]**

如果仅当 VCO 用，可以设置 dev_en=0 来关闭 FM 调制。调制限制详细说明见文档“设置调制限制和 MIC 灵敏度**”部分。**

(20) 发射功率控制

详细说明见文档“设置发射功率**”部分。**

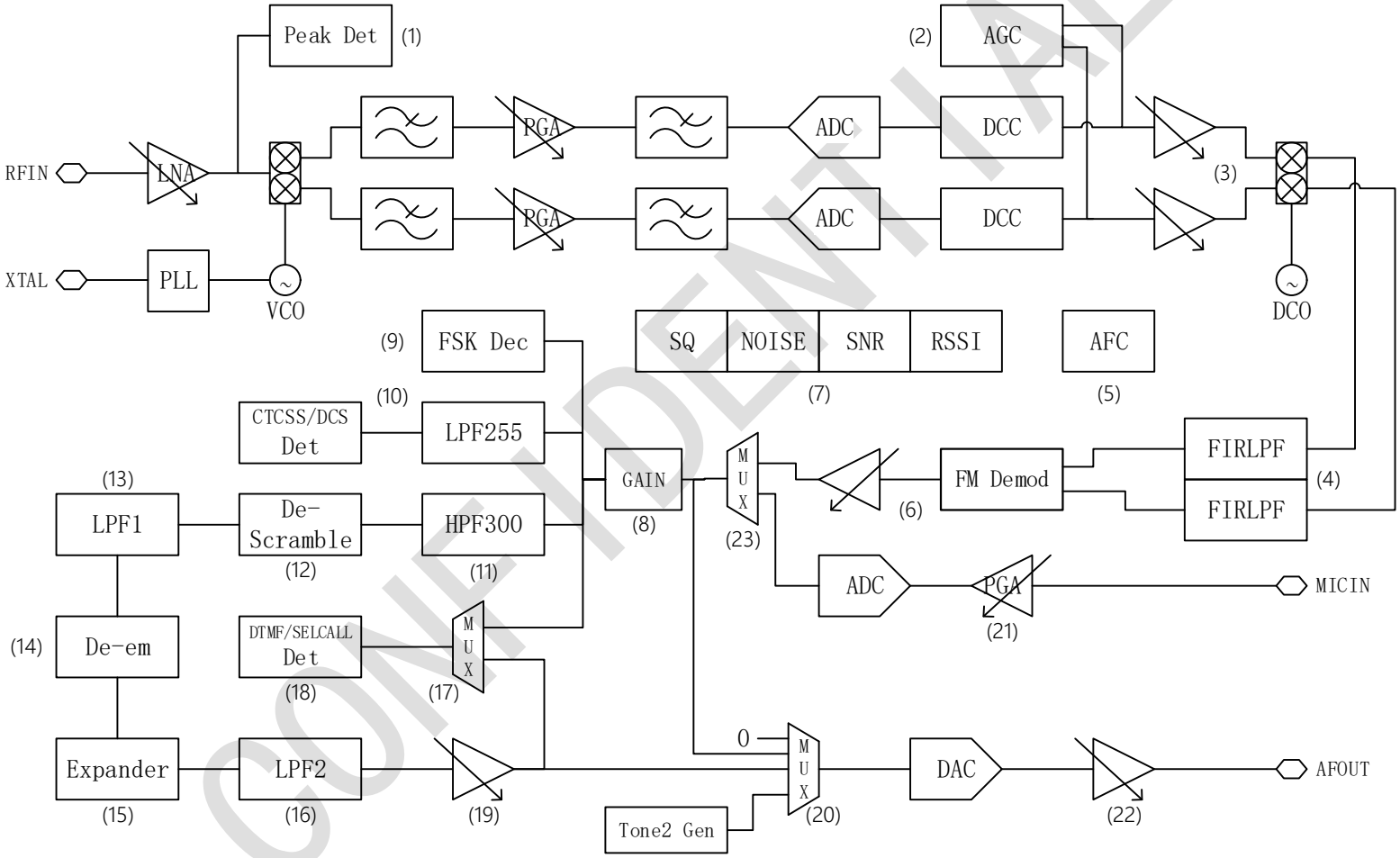
(21) 声控（VOX）检测

详细说明见文档“声控（VOX）、发射超时（TOT）设置**”部分。**

CONFIDENTIAL

接收框图

接收框图



CONFIDENTIAL

接收框图

框图说明:

(1) 模拟射频前端及增益

Register	Address(HEX)	Description
lna_peak_rssi[7:0]	REG_62H[7:0]	RSSI after LNA, 1dB/step. (Read Only)

经过 LNA 后的宽带信号强度指示。

(2) 接收 AGC

Register	Address(HEX)	Description
agc_rssi[7:0]	REG_62H[15:8]	RSSI after DCC, 1dB/step. (Read Only)

经过 ADC、DCC 后的窄带数字信号强度指示。

(3) 接收 IQ 数字增益 1

Register	Address(HEX)	Description
dig_gain_rx[4:0]	REG_7DH[12:8]	Gain after AGC, digital down conversion. 1dB/step

(4) 接收 IQ 滤波器，数字增益 2

Register	Address(HEX)	Description
firlpf_bw[2:0]	REG_43H[14:12]	RF filter bandwidth (Apass=0.1dB) 000 = 1.7 kHz 001 = 2 kHz 010 = 2.5 kHz 011 = 3 kHz 100 = 3.75 kHz 101 = 4 kHz 110 = 4.25 kHz 111 = 4.5 kHz if wb=1, firlpf_bw *=2;
firlpf_bw_for_weak[2:0]	REG_43H[11:9]	RF filter bandwidth when signal is weak.
wb	REG_43H[5]	1 = 25kHz/20kH; 0 = 12.5kHz/6.25kHz
firlpf_gain[1:0]	REG_43H[1:0]	Gain after FIR LPF

CONFIDENTIAL

		00=0dB; 01=6dB; 10=12dB; 11=18dB

(5) 自动频率校准（AFC）

详细说明见文档“AFC 设置”部分。

(6) FM 解调及增益

Register	Address(HEX)	Description
fmdem_gain[1:0]	REG_44H[12:11]	Gain after FM Demodulation 00=0dB; 01=6dB; 10=12dB; 11=18dB

(7) SQ、NOISE、SNR、RSSI

详细说明见文档“静噪（SQ）设置及 RSSI、NOISE 和 SNR”部分。

(8) 接收 DISC 信号增益

Register	Address(HEX)	Description
~~audio_rx_gain1[5:0]~~	~~REG_96H[13:8]~~	~~0=mute, 63=max, 0.5dB/step~~

DISC 信号包含音频和亚音频，所以此寄存器同时调节音频和亚音频增益。

(9) 接收 FSK 解码

详细说明见文档“FSK 模式设置”部分。

(10) 亚音频低通滤波器、亚音频解码

Register	Address(HEX)	Description
~~subau_rx_dcc_bw[2:0]~~	~~REG_B6H[12:10]~~	~~CTC/DCS Rx HPF BW~~ ~~000=bypass;001=60Hz;010=30Hz;011=15Hz;~~

CONFIDENTIAL

		~~100=8Hz;101=4Hz;110=2Hz;111=1Hz~~
subau_tx_atten_gain[1:0]	REG_B6H[9:8]	CTC/DCS Tx Atten Gain, -6dB/step
subau_rx_gain1[1:0]	REG_B6H[7:6]	CTC/DCS Rx Gain1, -6dB/step
subau_rx_gain2[1:0]	REG_B6H[5:4]	CTC/DCS Rx Gain2, -6dB/step
subau_rx_gain3[3:0]	REG_B6H[3:0]	CTC/DCS Rx Gain3, -6dB/step

亚音频解码详细说明见文档“亚音频设置”部分。

(11) 接收语音 300Hz 高通滤波器

Register	Address(HEX)	Description
audio_hpf_rx_bypass	REG_9BH[9]	1=Audio HPF 300Hz bypass for RX

(12) 接收语音解扰

详细说明见文档“SCRAMBLE 模式设置”部分。

(13) 接收语音低通滤波器 1

Register	Address(HEX)	Description
audio_lpf1_rx_bypass	REG_9BH[8]	1=Audio LPF1 bypass for RX

(14) 接收语音去加重

Register	Address(HEX)	Description
audio_emph_rx_bypass	REG_9BH[6]	1=DE-EMPHASIS bypass

(15) 接收语音解扩

详细说明见文档”语音模式设置”部分。

(16) 接收语音低通滤波器 2

Register	Address(HEX)	Description

CONFIDENTIAL

audio_lpf2_rx_bypass	REG_9BH[7]	1=Audio LPF2 bypass for RX

(17) 接收 DTMF/SELCALL 输入选择

Register	Address(HEX)	Description
dtmf_in_sel	REG_94H[5]	1=select gain out 0=select audio rx out

DTMF/SELCALL 接收信号选择，区别就在于进入 DTMF/SELCALL 解码的信号是否经过去加重处理。可根据设计需求自行选择。默认 dtmf_in_sel=1。

(18) 接收 DTMF/SELCALL 解码

详细说明见文档“DTMF 模式设置”和“SELCALL 模式设置”部分。

(19) 接收语音音量控制

Register	Address(HEX)	Description
audio_rx_gain_sh [2:0]	REG_53H[15:13]	000=max,111=min 0~-42dB, 6dB/step, digital gain
audio_rx_gain[4:0]	REG_53H[9:5]	0=mute,31=max 1dB/step, digital gain

(20) AF DAC 通路选择

同发射框图(17)。

(21) MIC 通路 PGA 模拟增益

同发射框图(1)。

(22) AF DAC 模拟增益

同发射框图(18)。

CONFIDENTIAL

接收框图

(23) 接收语音基带模式选择

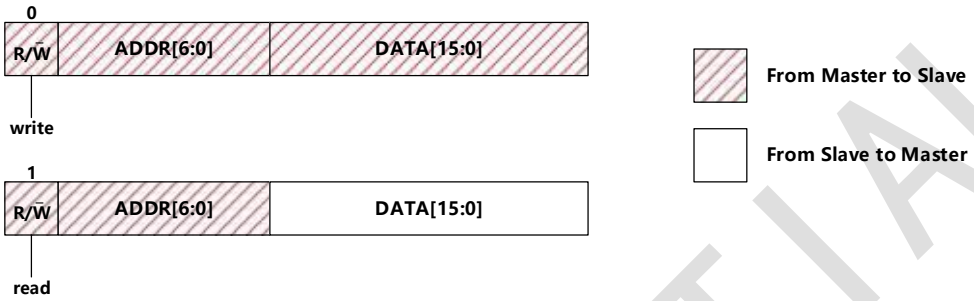
接收的 FM 解调后的 DISC 信号，通过 MICIN 送入芯片进行语音、亚音频、DTMF、5TONE 等解码。详细说明见文档“DISC 基带处理模式设置**”部分。**

CONFIDENTIAL

接口时序

3-WIRE 接口

格式如下：



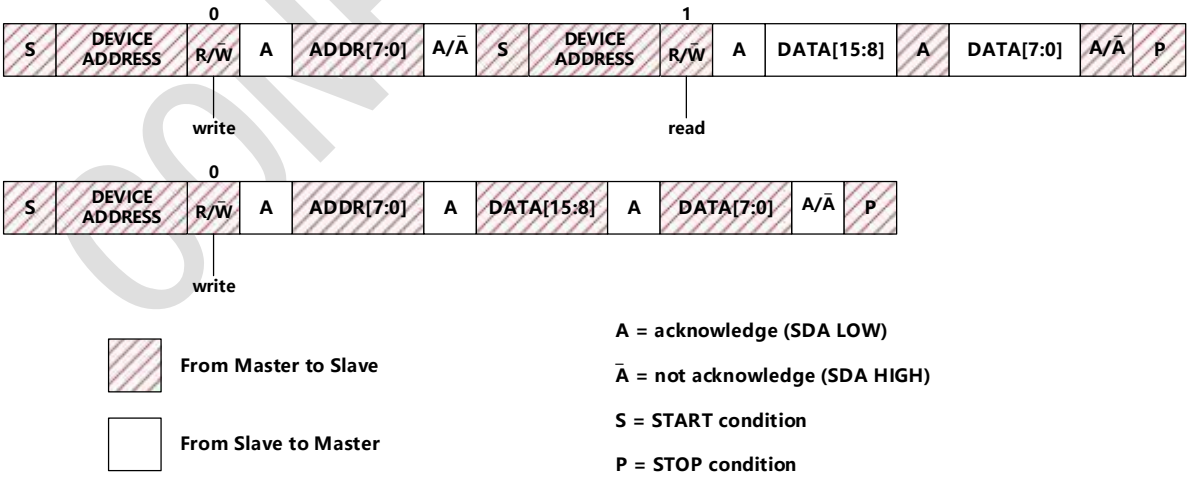
提示：3-wire 接口当读写寄存器地址大于 0x7F 时，需要翻页操作。如下：

Page0: WRITE(0x7F,0x0000); //default

Page1: WRITE(0x7F,0x0001);

2-WIRE 接口

格式如下(Device Address = ‘0101110’ when SCN is high(or no SCN pin), or Device Address = ‘1110001’ when SCN is low)：



提示：2-wire 接口读写寄存器不需要翻页操作，并“#define I2C 0x80 ”。

CONFIDENTIAL

配置流程

1. 初始化设置

WRITE(0x00,0x8000); //soft reset

WRITE(0x00,0x0000);

…

2. 设置晶体/晶振时钟频率

见“晶体/晶振时钟频率设置”部分说明。

3. 设置频段

见“频段、频点、带宽模式设置”部分说明。

4. 设置频段内频点

见“频段、频点、带宽模式设置”部分说明。

5. 设置 AGC 表

6. OFFSET 校准设置顺序

WRITE(0x03,0xBDF1); //打开接收通路，未打开 AF

WRITE(0x04,REG_04H & 0xFCFF);

WRITE(0x04,REG_04H | 0x0300); //bit[9:8]是 OFFSET 校准使能位

delay_ms(50);

WRITE(0x03,0x0000); //校准完毕进入 IDLE

提示：在每次更换频段时，都需要先设置该频段内的频点，然后重新做 OFFSET 校准。

CONFIDENTIAL

7. 设置宽窄带、亚音频和 INBAND 信号模式

8. 进入收/发状态

CONFIDENTIAL

收发状态设置

接收（RXON）设置

WRITE(0x03,0x0000);

WRITE(0x03,0xBFF1);

发射（TXON）设置

WRITE(0x03,0x0000);

WRITE(0x03,0xC1FE);

发射并伴有侧音

WRITE(0x03,0x0000);

WRITE(0x03,0xC1FE | 1<<9); //Enable AF

不发射，只产生侧音

WRITE(0x03,0x0002 | 1<<9); //Enable AF

IDLE 设置（可用于省电）

WRITE(0x03,0x0000);

提示：可以通过 RXON 和 IDLE 之间的切换来实现省电模式。

CONFIDENTIAL

附：寄存器说明

Register	Address(HEX)	Description
vco_cal_en	REG_03H[15]	1=Enable PLL Calibration, (0->1)
pabias_en	REG_03H[14]	1=Enable PABIAS Output
rxlink_en[3:0]	REG_03H[13:10]	1111=Enable Rx Link, (LNA,MIXER,FILTER,ADC)
afout_en	REG_03H[9]	1=Enable AFOUT DAC
pll_en[4:0]	REG_03H[8:4]	11111=Enable PLL, VCO
padrv_en	REG_03H[3]	1=Enable PADRV
micin_en	REG_03H[2]	1=Enable MICIN ADC
txon	REG_03H[1]	1=Enable Tx DSP
rxon	REG_03H[0]	1=Enable Rx DSP

CONFIDENTIAL

频段、频点、带宽模式设置

发射接收频点设置

例如设置频率为 409.75MHz，则 DEC2HEX(409.75*100000)=0x2713A98

WRITE(0x38,0x3A98);

WRITE(0x39,0x0271);

提示：每次设置频率后，需要重新 RXON 或 TXON，频率才会切换；频率最小分辨率为 10Hz。

宽带（25KHZ）模式设置

~~WRITE(0x3A,0x00E0);~~

WRITE(0x43,0x3028);

提示：可以选择合适的发射和接收带宽，以上只是推荐值。

窄带（12.5KHZ）模式设置

~~WRITE(0x3A,0x0040);~~

WRITE(0x43,0x4048);

提示：可以选择合适的发射和接收带宽，以上只是推荐值。

附：寄存器说明

Register	Address(HEX)	Description
freq [15:0]	REG_38H[15:0]	Frequency =(REG_39H<<16 + REG_38H)*10 Hz
freq [31:16]	REG_39H[15:0]	
firlpf_bw[2:0]	REG_43H[14:12]	RF filter bandwidth (Apass=0.1dB) 000 = 1.7 kHz 001 = 2 kHz 010 = 2.5 kHz 011 = 3 kHz 100 = 3.75 kHz 101 = 4 kHz

CONFIDENTIAL

		110 = 4.25 kHz 111 = 4.5 kHz if wb=1, firlpf_bw *=2;
firlpf_bw_for_weak[2:0]	REG_43H[11:9]	RF filter bandwidth when signal is weak.
audio_lpf2_tx_sel[2:0]	REG_43H[8:6]	Audio LPF bandwidth (Apass=1dB) for Tx 100 = 4.5 kHz 101 = 4.25 kHz 110 = 4 kHz 111 = 3.75 kHz 000 = 3 kHz 001 = 2.5 kHz 010 = 2 kHz 011 =1.7kHz
wb	REG_43H[5]	Rx filter band width mode select 1=25khz/20khz wide band mode, 0=12.5khz narrow band mode

附：频段设置表

BAND	REG_1AH	REG_40H 24M~26M	REG_40H 12M~13M	REG_40H 18M~20M
Band_740M_1120M	*0x1f80*	*0x3ad0*	*0x37d0*	*0x373a*
Band_370M_560M	*0x2f50*	*0x39d0*	*0x38d0*	*0x383a*
Band_247M_373M	*0x9f30*	*0x385c*	*0x375c*	*0x38d7*
Band_185M_280M	*0x3f48*	*0x38d0*	*0x37d0*	*0x373a*
Band_124M_186M	*0xaf28*	*0x375c*	*0x365c*	*0x37d7*
Band_93M_140M	*0x4f44*	*0x37d0*	*0x36d0*	*0x363a*
Band_62M_93M	*0xbf24*	*0x365c*	*0x355c*	*0x36d7*
Band_46M_70M	*0x5f42*	*0x36d0*	*0x35d0*	*0x353a*
Band_31M_46M	*0xcf22*	*0x355c*	*0x345c*	*0x35d7*
Band_23M_35M	*0x6f41*	*0x35d0*	*0x34d0*	*0x343a*
Band_16M_23M	*0xdf21*	*0x345c*	*0x335c*	*0x34d7*

CONFIDENTIAL

设置调制限制和 MIC 灵敏度

在仅将芯片做 VCO 使用时，可以设置 dev_en=0 来关闭 FM 调制，以免 MICIN 信号影响本振输出。

dev_sh 的值是 2 倍递减关系，如 dev_lvl 设置相同的值，dev_sh=7 的调制就是 dev_sh=8 的调制的 2 倍。

调制限制、MIC 灵敏度和频响的调试顺序建议为：

1. 调制限制（输入较大 MICIN 信号，如 200mv，设置 dev_sh 和 dev_lvl 使最大调制符合设计要求）
2. 发射频响（按照发射频响 300Hz 和 3KHz 的设置说明进行相应设置。因为改变频响会影响 MIC 灵敏度，所以要放在前面）
3. MIC 灵敏度（调试 mic_sens_gain，使 MIC 灵敏度符合设计要求）

提示：不同频段调制寄存器（REG_40H）需要设置不同的值。另外，由于会影响亚音频的调制大小，所以要先确定 REG_40H 的值后，再调试亚音频发射增益（REG_51H）。

附：调制限制和 MIC 灵敏度寄存器

Register	Address(HEX)	Description
dev_en	REG_40H[12]	Enable FM deviation
dev_sh[3:0]	REG_40H[11:8]	**FM deviation coarse tuning,** **0000=max, 1111=min**
dev_lvl[7:0]	REG_40H[7:0]	FM deviation fine tuning ① 0000=min, 1111=max, **GAIN=(256+dev_lvl[7:0])>>dev_sh[3:0]**
~~mic_sens_gain[5:0]~~	~~REG_97H[13:8]~~	~~MIC sensitivity adjust gain~~ ~~0=mute, 63=max, 0.5dB/step~~

~~① 例如，当前 dev_lvl=0xA3，调制大小为 2.1kHz。如果想把调制修改为 2.2khz，那么计算方法如下：~~

~~dev_lvl= (0xA3+256)/2.1*2.2-256 ~~=0xB7;~~

如果调整幅度很大，就需要修改 dev_sh[3:0]，是 2 倍单调递减关系。

CONFIDENTIAL

设置发射功率

发射功率由寄存器 padrv_gain 和 pa_gain_vreg 进行控制，同时可以输出 PA Bias 来控制外部 PA 功率。

附：发射功率寄存器

Register	Address(HEX)	Description
pabias_en	REG_03H[14]	1=Enable PAbias output
pabias_out[3:0]	REG_19H[3:0]	PAbias output 1.3~2.8V, 100mv/step 0000=1.3V … 1111=2.8V
padrv_gain[2:0]	REG_28H[5:3]	111(max)->000(min)
pa_gain_vreg[2:0]	REG_28H[2:0]	111(max)->000(min)

功率(dB)	Padrv_gain[2:0]							
Pa_gain_vreg[2:0]	111	110	101	100	011	010	001	000
111	8.33	7.96	7.53	7.11	6.62	5.69	4.12	1.59
110	7.96	7.11	6.62	6.19	5.69	4.68	2.16	-2.16
101	7.11	6.19	6.19	5.23	4.68	3.10	0.84	-5.54
100	6.40	5.69	5.23	4.68	3.44	2.16	-0.67	-8.61
011	5.70	4.68	4.12	3.44	2.16	0.85	-3.62	-12.78
010	4.10	3.44	2.16	1.59	0.15	-2.84	-8.2	-19.53
001	2.80	1.59	0.84	-0.67	-2.16	-5.5	-11.4	-22.65
000	1.60	0.84	-0.67	-2.16	-4.28	-8.2	-13.7	-24.00

CONFIDENTIAL

接收静音（MUTE）及音量设置

设置静音：

WRITE(0x47,(REG_47H & 0xE0FF) | 0x1000);

设置接收 AF 输出：

WRITE(0x47,(REG_47H & 0xF0FF) | 0x1100);

设置发射侧音输出：

WRITE(0x47,(REG_47H & 0xE0FF) | 0x1800);

附：静音、关静音、BEEP 侧音选择寄存器

Register	Address(HEX)	Description
afout_mode[4:0]	REG_47H[12:8]	0x10 = MUTE 0x11 = RX AFOUT 0x18 = BEEP/ TX Side Tone（CTCSS/CDCSS is not include） 0x15 = RX ALARM TONE

附：接收音量寄存器

Register	Address(HEX)	Description
audio_rx_gain_sh[2:0]	REG_53H[15:13]	000=max,111=min 0~-42dB, 6dB/step, digital gain
audio_rx_gain[4:0]	REG_53H[9:5]	0=mute,31=max, 1dB/step, digital gain
dac_vgain[3:0]	REG_23H[3:0]	DAC maximum output level

~~**提示：可以用寄存器 audio_rx_gain_sh 和 audio_rx_gain 组合出音量控制档位，而 dac_vgain 则固定在一个档位（一般固定在最大档）。**~~

CONFIDENTIAL

音频响应调节

低频音频响应

增加发射 300Hz 幅度：

```
WRITE(0x7F,0x0001); //page=1
WRITE(I2C | 0x30,0x8942);
WRITE(I2C | 0x31,0x3751);
WRITE(0x7F,0x0000); //page=0
```

增加接收 300Hz 幅度：

```
WRITE(0x2C,0x8942);
WRITE(0x2D,0x3751);
```

高频音频响应

增加发射 3 KHz 幅度：

```
WRITE(0x7F,0x0001); //page=1
WRITE(I2C | 0x23,0xC1BB);
WRITE(I2C | 0x24,0x2226);
WRITE(0x7F,0x0000); //page=0
```

增加接收 3 KHz 幅度：

```
WRITE(0x7F,0x0001); //page=1
WRITE(I2C | 0x21,0xBBC0);
WRITE(I2C | 0x22,0x2616);
WRITE(0x7F,0x0000); //page=0
```

CONFIDENTIAL

语音模式设置

附：语音相关寄存器

Register	Address(HEX)	Description
audio_emph_rx_bypass	REG_9BH[6]	DE-EMPHASIS bypass
audio_emph_tx_bypass	REG_9BH[0]	PRE-EMPHASIS bypass
audio_hpf_rx_bypass	REG_9BH[9]	Audio HPF 300Hz bypass for RX
audio_hpf_tx_bypass	REG_9BH[3]	Audio HPF 300Hz bypass for TX
audio_lpf1_rx_bypass	REG_9BH[8]	Audio LPF1 bypass for RX
audio_lpf1_tx_bypass	REG_9BH[2]	Audio LPF1 bypass for TX
audio_lpf2_rx_bypass	REG_9BH[7]	Audio LPF2 bypass for RX
audio_lpf2_tx_bypass	REG_9BH[1]	Audio LPF2 bypass for TX
cmpd_rx_factor[2:0]	REG_98H[14:12]	Expanding Factor. 100=1:3;011=1:2.5;010=1:2;001=1:1.5
cmpd_rx_th_high[5:0]	REG_98H[11:6]	Above this amplitude point, audio will be expanded according to the expanding factor. The unit of this threshold is 2dB.
cmpd_rx_th_low[5:0]	REG_98H[5:0]	Under this amplitude point, audio will be attenuated. The unit of this threshold is 2dB.
cmpd_rx_gain[4:0]	REG_96H[7:3]	Audio CMPD Gain, 1dB/step
cmpd_rx_bypass[2:0]	REG_96H[2:0]	Audio Compander bypass for RX 110=bypass with gain 001=bypass without gain 000=CMPD ON for RX Others=reserved
cmpd_tx_factor[2:0]	REG_99H[14:12]	Compressing Factor. 111=8:1;110=4:1;100=2:1;000=1:1
cmpd_tx_th_high[5:0]	REG_99H[11:6]	Above this amplitude point, audio will be compressed according to the compressing factor. The unit of this threshold is 2dB.
cmpd_tx_th_low[5:0]	REG_99H[5:0]	Under this amplitude point, audio will be attenuated. The unit of this threshold is 2dB.
cmpd_tx_gain[4:0]	REG_97H[7:3]	Audio CMPD Gain, 1dB/step
cmpd_tx_bypass[2:0]	REG_97H[2:0]	Audio CMPD bypass for TX

CONFIDENTIAL

		110=bypass with gain 001=bypass without gain 000=CMPD ON for TX Others=reserved
cmpd_ct_intvl[5:0]	REG_9AH[13:8]	Companding Amplitude Detect Interval, 0.64ms/step
cmpd_atk_step[3:0]	REG_9AH[7:4]	Companding Gain Attack Speed. 0000=most fast … 1111=most slow
cmpd_rls_step[3:0]	REG_9AH[3:0]	Companding Gain Release Speed. 0000=most fast … 1111=most slow
cmpd_db_out[6:0]	REG_87H[6:0]	Audio amplitude output, 1dB/step. (Read Only)

压扩点设置方法

发射压扩点：将 micin 信号幅度加至压扩点幅度，读 **cmpd_db_out（REG_87H[6:0]），**所得值除以 2，设置到 **cmpd_tx_th_high（REG_99H[11:6]）**即可。

接收解扩点：将信号源调制频偏设置为压扩点频偏，读 **cmpd_db_out（REG_87H[6:0]），**所得值除以 2，设置到 **cmpd_rx_th_high（REG_98H[11:6]）**即可。

CONFIDENTIAL

SCRAMBLE 模式设置

附：扰频及频率设置寄存器

Register	Address(HEX)	Description
scramb_en	REG_04H[5]	Enable SCRAMBLER
scramb_freq[15:0]	REG_54H[15:0]	SCRAMBLE frequency control word =3.3(KHz)*2^26/6500 **- The scrambler inversion mixing frequency should be kept between 2.6kHz and 3.5kHz**

CONFIDENTIAL

DTMF 模式设置

附：DTMF/ SELCALL 接收寄存器

Register	Address(HEX)	Description
dtmf_code_ready	REG_0BH[12]	1=DTMF/ SELCALL symbol received ready. (Read Only)
dtmf_code[3:0]	REG_0BH[11:8]	DTMF/ SELCALL symbol received. (Read Only)
dtmf_det_th[5:0]	REG_94H[12:7]	DTMF/ SELCALL detect threshold, 1dB/step
dtmf_symbol_mode	REG_94H[6]	DTMF/ SELCALL symbol mode 1=" symbol" +" idle" +" symbol" +... (DTMF) 0=" symbol" +" symbol" +" symbol" +...(5TONE)
dtmf_in_sel	REG_94H[5]	1=select gain out 0=select audio rx out
dtmf_mode	REG_94H[4]	Dual/Single Tone Detect mode 1=dual tone 0=single tone
dtmf_symbol_max[3:0]	REG_94H[3:0]	SELCALL maximum symbol number 15=symbol "0" ~" F" (DTMF) 14=symbol "0" ~" E" (EIA) ...

标准 DTMF 符号表格

DTMF Symbol		High Frequency (Hz)			
		1209	**1336**	**1477**	**1633**
Low Frequency (Hz)	**697**	'1'	'2'	'3'	'A'
	770	'4'	'5'	'6'	'B'
	852	'7'	'8'	'9'	'C'
	941	'E'	'0'	'F'	'D'

发射流程：

1. 设置发射静音：WRITE(0x50,REG_50H | 0x8000);

CONFIDENTIAL

2. 设置 TONE1 和 TONE2 的频率和增益

3. 延时 T1

4. 设置发射开启：WRITE(0x50,REG_50H);

5. 延时 T2 后，若 DTMF 发射继续跳转到 1，发射下一个 DTMF 符号

0	TONE1+TONE2	0	TONE1+TONE2	…
T1	T2	T1	T2	…

接收流程：

1. 等中断，或读 dtmf_code_ready 寄存器，直到=1

2. 读 dtmf_code[3:0]得到接收到的 DTMF 符号，若接收未结束跳转到 1，继续等待下一个 DTMF 符号

CONFIDENTIAL

SELCALL 模式设置

发射流程：

1. 设置发射静音：WRITE(0x50,REG_50H | 0x8000);
2. 延时 T1
3. 设置发射开启：WRITE(0x50,REG_50H);
4. 设置 TONE1 的频率和增益
5. 延时 T2 后，若 SELCALL 发射未结束跳转到 4，发射下一个 SELCALL 符号

0	**TONE1**	**TONE1**	**TONE1**	**…**
T1	**T2**	**T2**	**T2**	**…**

接收流程：

1. 等中断，或读 dtmf_code_ready 寄存器，直到=1
2. 读 dtmf_code[3:0]得到接收到的 SELCALL 符号，若接收未结束跳转到 1，继续等待下一个 SELCALL 符号

CONFIDENTIAL

1050Hz 单音解码

利用 DTMF 进行 1050 Hz 报警解码，则配置如下：

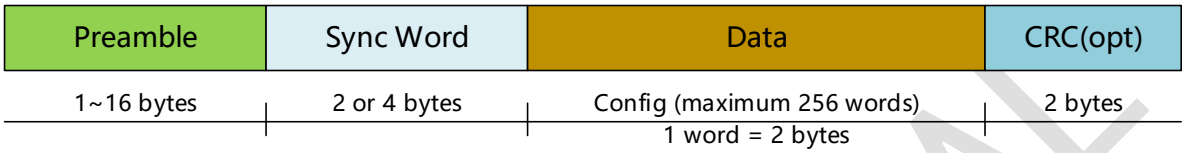
```
WRITE(0x04,0x0302);

WRITE(0x09,0x0002);

WRITE(0x09,0x105A);

WRITE(0x09,0x204B);

WRITE(0x09,0x3066);

WRITE(0x09,0x403B);

WRITE(0x07,0x1021);

WRITE(0x51,0x9400);

WRITE(0x52,0x1cb2);

WRITE(0x54,0x3269);

WRITE(0x7F,0x0001); //page1 for SPI

WRITE(0x11 | I2C,0x06FD);

WRITE(0x14 | I2C,0x8024 | 10<<7); //[12:8] decode threshold, change @2019.04.22

WRITE(0x7F,0x0000); //page0 for SPI


//polling REG_0B and REG_60,

//if REG_0B[12:8]==0x11 && REG_60[0]==1

//1050Hz is detectd.
```

CONFIDENTIAL

FSK 模式设置

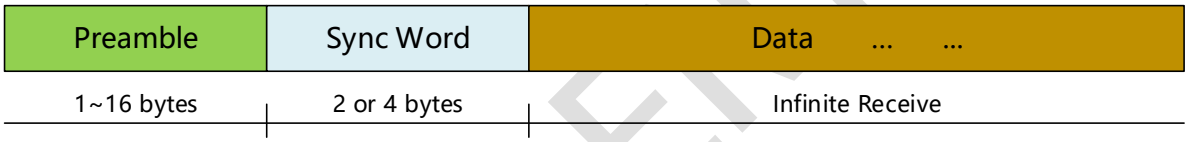
FSK 帧格式 1（固定长度）

此模式支持固定长度的数据帧传输，CRC 运算可选。



FSK 帧格式 2（无限接收）

此模式支持较大数据的传输，这种情况下，数据包长度未知，设备会将同步字后的数据永久导入 FIFO。但不支持 CRC 运算，因为数据包终端对应设备是未知的。该模式比较灵活，适合用户自定义帧结构，帧长度。

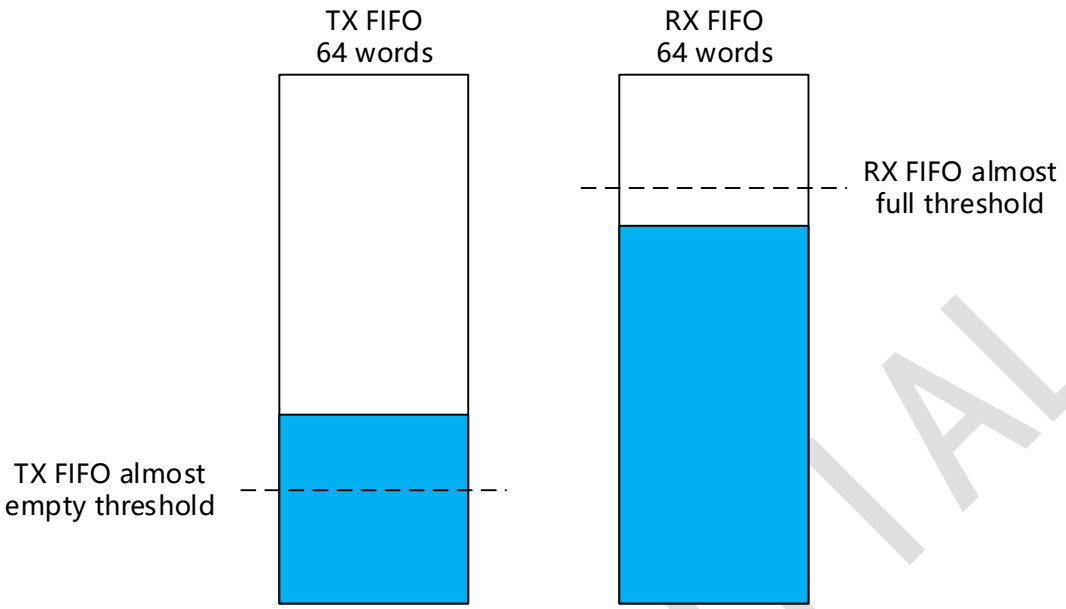


TX 和 RX FIFO

芯片内部集成 2 个 64 words（=128 bytes）的 FIFO，一个用于 TX，一个用于 RX（如下图）。通过写 REG_5FH 寄存器来对 TX FIFO 进行数据写入，通过读 REG_5FH 寄存器对 RX FIFO 数据进行读取。TX FIFO 有 fsk_tx_fifo_ae_th[5:0]（TX FIFO Almost Empty）可以设置，通过产生中断来提示对 TX FIFO 进行数据写入；RX FIFO 有 fsk_rx_fifo_af_th[5:0]（RX FIFO Almost Full）可以设置，通过产生中断来提示对 RX FIFO 进行数据读取。同时 TX FIFO 和 RX FIFO 各有一个 clear 寄存器可以用来清除或重置 TX FIFO 和 RX FIFO。

CONFIDENTIAL

FSK 模式设置



附：FSK 设置寄存器

Register	Address(HEX)	Description
fsk_en	REG_04H[4]	Enable FSK mode
fsk_rate[15:0]	REG_54H[15:0]	FSK/Scramble use the same rate register, =freq(kHz)*2^26/6500
fsk_crc_polyn[15:0]	REG_56H[15:0]	CRC polynomial coefficient, CCIT-16 is default
fsk_tx_fifo_clear	REG_59H[15]	Clear TX FIFO, 1=clear
fsk_rx_fifo_clear	REG_59H[14]	Clear RX FIFO, 1=clear
fsk_scramble_en	REG_59H[13]	1=Enable FSK Scramble
fsk_rx_en	REG_59H[12]	1=Enable FSK TX
fsk_tx_en	REG_59H[11]	1=Enable FSK RX
fsk_rx_data_inv	REG_59H[10]	1=invert FSK data after RX
fsk_tx_data_inv	REG_59H[9]	1=invert FSK data before TX
fsk_prmb_size[3:0]	REG_59H[7:4]	FSK preamble length select, length=(fsk_prmb_size[3:0]+1) bytes
fsk_sync_size	REG_59H[3]	FSK sync length select, 1=4 bytes (fsk_sync_byte0,1,2,3)

CONFIDENTIAL

		0=2 bytes (fsk_sync_byte0,1)
fsk_sync_byte0[7:0]	REG_5AH[15:8]	FSK sync byte0
fsk_sync_byte1[7:0]	REG_5AH[7:0]	FSK sync byte1
fsk_sync_byte2[7:0]	REG_5BH[15:8]	FSK sync byte2
fsk_sync_byte3[7:0]	REG_5BH[7:0]	FSK sync byte3
fsk_tx_type_byte[7:0]	REG_5CH[7:0]	FSK type byte to be transmitted. “0x01 “=FSK Format2 “0x21” =FSK Format1 without CRC “0x61” =FSK Format1 with CRC
fsk_length[7:0]	REG_5DH[15:8]	FSK data length when use FSK format1 length=(fsk_length[7:0]+1) words (1 word = 2 bytes)
fsk_tx_fifo_ae_th[5:0]	REG_5EH[11:6]	FSK TX FIFO almost empty threshold(unit is 1word=2 bytes)
fsk_rx_fifo_af_th[5:0]	REG_5EH[5:0]	FSK RX FIFO almost full threshold(unit is 1word=2 bytes)
fsk_data[15:0]	REG_5FH[15:0]	TX/RX FIFO Data

发射流程：

1. 清 TX FIFO，然后写数据到 TX FIFO
2. 发射 TXON
3. 等中断提示 FSK 发射完成
4. 结束发射 IDLE

接收流程：

1. 接收 RXON
2. 等中断提示接收完成
3. 读取 RX FIFO 数据，同时读取 fsk_crc_ok 标志位进行差错控制
4. 结束接收 IDLE，（延迟一定间隔时间）重新 RXON 等待下一 FSK 帧到来

CONFIDENTIAL

DISC 基带处理模式设置

1. 该模式仅作语音、亚音频收发基带处理。接收 DISC 和发射 MIC 的基带信号均由 MICIN 口（MICIN 的输入接收 DISC 信号幅度需要控制在 200mV 以下）送入设置如下：

   WRITE(0x44,0x53EC | 0x1); //[0] DISC mode enable

   WRITE(0x7E,0x341E | 3<<6); //DCC bypass

   WRITE(0x7F,0x0001); //page1

   WRITE(I2C | 0x1C,0x076F);

   WRITE(I2C | 0x1D,0xB9FD);

   WRITE(I2C | 0x27,0x18DA);

   WRITE(0x7F,0x0000); //page0

   RXON:

   WRITE(0x03,0x0000); WRITE(0x03,0x0005);

   TXON:

   WRITE(0x03,0x0000); WRITE(0x03,0x0006 | 1<<9); //打开 AF

   或

   WRITE(0x03,0x0000); WRITE(0x03,0xC1FE); //直接调制到 RFO 发射出去

2. 做语音、亚音频接收解码（MICIN 的输入信号幅度需要控制在 200mV 以下），同时做 VCO 使用。设置如下：

   WRITE(0x13,0x03FF);

   WRITE(0x35,0xF108);

   WRITE(0x3C,REG_3CH & 0xFFC0);

   WRITE(0x3D,0x0000);

CONFIDENTIAL

DISC 基带处理模式设置

WRITE(0x44,0x53EC | 0x1); //[0] DISC mode enable

WRITE(0x7E,0x341E | 3<<6); //DCC bypass

WRITE(0x7F,0x0001); //page1

WRITE(I2C | 0x1C,0x076F);

WRITE(I2C | 0x1D,0xB9FD);

WRITE(I2C | 0x27,0x18DA);

WRITE(0x7F,0x0000); //page0

RXON:

WRITE(0x04,0x0800);

WRITE(0x03,0x0000); WRITE(0x03,0x81FD); //RFO 输出 VCO，AF 输出音频，内部做亚音频解码

TXON:

WRITE(0x03,0x0000); WRITE(0x03,0x0006 | 1<<9); //只做发射音频滤波，从 AF 输出

或

WRITE(0x04,0x0400);

WRITE(0x03,0x0000); WRITE(0x03,0xC1FE)//正常调制并从 RFO 发射出去

CONFIDENTIAL

BYPASS 模式设置（用于 DMR/dPMR）

该模式可作 DMR/dPMR 等产品的收发器，MICIN 送入已成型滤波的 4FSK 信号进行调制发射，AFOUT 送出解调后的 4FSK 信号。设置如下（芯片要设置成窄带 12.5k 模式）：

WRITE(0x04,0x0300);

~~WRITE(0x3a,0x0040); // 12.5k~~

~~WRITE(0x43,0x7e09);~~ -> WRITE(0x43,0x7e08); // 设置 DMR 带宽 bit14:12

WRITE(0x48,0x6c5c); // cic mode

WRITE(0x4b,0xe442); // cic mode

WRITE(0x73,0x569a); //关闭 AFC

WRITE(0x7E,0x341E | 1<<7); //dcc_tx_bypass=1

如果发射通过调外部 TCXO 实现，还需要设置

WRITE(0x40,0x0000); //关闭内部调制

除上述设置外，收发控制

TXON 改为

WRITE(0x47,0x6740); //MIC ADC 后直接发射

WRITE(0x03,0x0000);

WRITE(0x03,0xC1FE);

RXON 改为

WRITE(0x47,0x6140); //FM 解调后直接送 AFOUT

WRITE(0x03,0x0000);

~~WRITE(0x03,0xC1FE);~~ ->WRITE(0x03,0xBFF1);

CONFIDENTIAL

VCO 模式设置

接收和发射都作 VCO 使用

1. 修改初始化里相应寄存器

   ~~WRITE(0x34,0x606F);~~ WRITE(0x35,0xF608); //关闭相应的时钟，减小 VCO 干扰

   WRITE(0x40,0x0000); //关闭调制

2. TXON 和 RXON 都用

   WRITE(0x03,0x0000);

   ~~WRITE(0x03,0x01FA);~~ WRITE(0x03,0x81FA);

3. 根据需求修改收发频率频段

仅接收作 VCO 使用，发射链路正常使用

1. 修改初始化里相应寄存器

   ~~WRITE(0x34,0x206F); //关闭接收部分时钟，减少 VCO 干扰~~

2. TXON 为

   WRITE(0x03,0x0000);

   WRITE(0x40,????); //正常调制的值

   WRITE(0x03,0xC1FE);

3. RXON 为

   WRITE(0x03,0x0000);

   WRITE(0x40,0x0000); //关闭调制

   ~~WRITE(0x03,0x01FA);~~ WRITE(0x03,0x81FA);

4. 根据需求修改收发频率频段

CONFIDENTIAL

按键音侧音 BEEP 等设置

按键音侧音开启：

WRITE(0x03,0x0202); //enable AF and Tx(PLL and PA are not include)

WRITE(0x47,(REG_47H & 0xE0FF) | 0x1800);

按键音侧音 BEEP 和 CALL TONE 频率通过寄存器 REG_71H, REG_72H 设置：

双音设置 tone1_gen=1, tone2_gen=1

单音设置 tone1_gen=1

发射频率为 freq(Hz)，计算公式为=freq(kHz)* 2^26/6500

发射 CALL TONE 同时播放侧音需要置 AFOUT(REG_03[9])=1，并且设置

WRITE(0x47,(REG_47H & 0xE0FF) | 0x1800);

发射 MUTE：

WRITE(0x50,REG_50H | 0x8000);

发射 NOMUTE：

WRITE(0x50,REG_50H);

接收报警音（TONE2）

WRITE(0x47,(REG_47H & 0xE0FF) | 0x1500);

附：AFOUT 侧音控制寄存器

Register	Address(HEX)	Description
afout_mode[4:0]	REG_47H[12:8]	0x10 = MUTE 0x11 = RX AFOUT 0x18 = BEEP/TX Side Tone（CTCSS/CDCSS is not include）

CONFIDENTIAL

		0x15 = RX ALARM TONE

附：TONE1,TONE2 设置寄存器

Register	Address(HEX)	Description
tone1_gen	REG_70H[15]	Enable TONE1
tone1_gain[6:0]	REG_70H[14:8]	TONE1 tuning gain
tone2_gen	REG_70H[7]	Enable TONE2
tone2_gain[6:0]	REG_70H[6:0]	TONE2 tuning gain
tone1_freq[15:0]	REG_71H[15:0]	TONE1 frequency· control word =freq(kHz)* 2^26/6500
tone2_freq[15:0]	REG_72H[15:0]	TONE2 frequency· control word =freq(kHz)* 2^26/6500

CONFIDENTIAL

亚音频设置

关闭亚音频：

```
WRITE(0x51,0x0000);
```

设置 CTCSS 发射和接收：

```
WRITE(0x51,0x90C5); //[6:0] Gain

WRITE(0x07,0x0811); //100Hz

WRITE(0x07,0x0470 | 1<<13); //55Hz tail
```

设置 CDCSS 发射和接收：

```
WRITE(0x51,0x80B0); //[6:0] Gain

WRITE(0x07,0x0AD7); //134.4Hz DCS rate

WRITE(0x07,0x0AD7 | 1<<13); //134.4Hz CTCSS tail

WRITE(0x08,0x0813); //set ‘023’ low 12-bit

WRITE(0x08,0x0763 | 1<<15); //set ‘023’ high 12-bit
```

附：亚音频设置寄存器

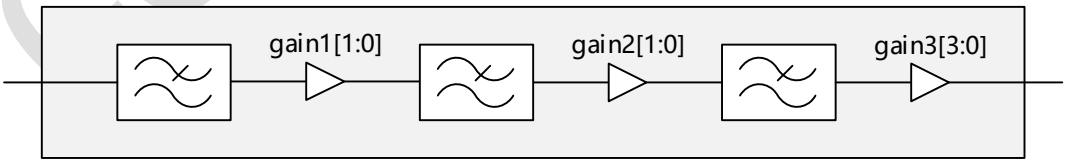
Register	Address(HEX)	Description
subau_en	REG_51H[15]	1=Enable CTCSS/CDCSS
dcs_invert	REG_51H[13]	1=Transmit negative CDCSS code 0=Transmit positive CDCSS code
ctc_dcs_sel	REG_51H[12]	CTCSS/CDCSS mode select, 1=CTCSS, 0=CDCSS
dcs_24b_mode	REG_51H[11]	24/23bit CDCSS select,

CONFIDENTIAL

		1=24bit, 0=23bit
subau_gain[6:0]	REG_51H[6:0]	Gain ① = (1+subau_gain[4:0]/32)<<subau_gain[6:5] [6:5]: CTCSS/CDCSS coarse tuning gain [4:0]: CTCSS/CDCSS fine tuning gain
ctc_freq[15:0]	REG_07H[15:0]	[13]=1 for CTCSS1/CDCSS [13]=0 for CTCSS0/CDCSS [12:0]: CTCSS frequency control word = freq(Hz)*2^27/6500000
dcs_code[15:0]	REG_08H[15:0]	[15]=1 for CDCSS high 12bit [15]=0 for CDCSS low 12bit [11:0] CDCSS 23/24bit code
~~subau_rx_dcc_bw[2:0]~~	~~REG_B6H[12:10]~~	~~CTC/DCS Rx HPF Bw 000=bypass;001=60Hz;010=30Hz;011=15Hz; 100=8Hz;101=4Hz;110=2Hz;111=1Hz~~
subau_tx_atten_gain[1:0]	REG_B6H[9:8]	CTC/DCS Tx Atten Gain ②, -6dB/step
subau_rx_gain1[1:0]	REG_B6H[7:6]	CTC/DCS Rx Gain1 ③, -6dB/step
subau_rx_gain2[1:0]	REG_B6H[5:4]	CTC/DCS Rx Gain2 ③, -6dB/step
subau_rx_gain3[3:0]	REG_B6H[3:0]	CTC/DCS Rx Gain3 ③, -6dB/step

①② 亚音频发射增益除了 subau_gain[6:0]可调之外，还可以调整 subau_tx_atten_gain[1:0]。另外要注意的是，亚音频发射调制频偏和 REG_40H 寄存器相关联，所以要先确定 REG_40H 的值（即最大调制）后再来调整 subau_gain[6:0]和 subau_tx_atten_gain[1:0]的值。

③ 亚音频接收增益按先后顺序依次为 subau_rx_gain1[1:0], subau_rx_gain2[1:0], subau_rx_gain3[3:0]。如需调整，应优先从第一级开始。



254Hz Low Pass Filer

CONFIDENTIAL

亚音频设置

附：CTCSS/CDCSS 尾音生成寄存器

Register	Address(HEX)	Description
subau_tail_gen	REG_52H[15]	sub-audio tail generate 0=normal, 1=tail generate
ctc_tail_offs[1:0]	REG_52H[14:13]	CTCSS tail mode and phase change select, 00=No phase shift（generate CTCSS1） 01=CTCSS0 120°phase shift, 10= CTCSS0 180°phase shift 11= CTCSS0 240°phase shift CTCSS Phase Decode is not supported.

发射 CTCSS 时，设置 subau_tail_gen=1，若 ctc_tail_offs=0，则自动发射 CTCSS1 作为尾音；若 ctc_tail_offs 为其他值，则自动发射相应相位移位的 CTCSS0 作为尾音。

发射 CDCSS 时，设置 subau_tail_gen=1 后，自动发射 CTCSS1 作为尾音。

附：亚音频检测寄存器

Register	Address(HEX)	Description
ctc_th_mode	REG_52H[12]	ctcss detect threshold mode, 1=~0.1%; 0=0.1 Hz
ctc_th_in[5:0]	REG_52H[11:6]	CTCSS found detect threshold
ctc_th_out[5:0]	REG_52H[5:0]	CTCSS lost detect threshold
dcs_detect[1:0]	REG_60H[11:10]	[0]:CDCSS positive code received [1]:CDCSS negative code received (Read Only)
ctc_detect[1:0]	REG_60H[1:0]	[1]:CTCSS1 received [0]:CTCSS0 received (Read Only)

CTCSS 检测阈值有两种模式，1 为频率百分比（单位为~0.1%）模式，0 为频率(单位为 0.1Hz)模式（默认）。

//mode=0 时，ctc_th_in/ ctc_th_out 对应的单位约为 0.1Hz

CONFIDENTIAL

亚音频设置

//mode=1 时，ctc_th_in/ ctc_th_out 对应的单位为约 0.1%

//mode=0 时，若 ctc_th_in/ ctc_th_out=15，那么对应阈值就约为 15*0.1=1.5hz

//mode=1 时，若 ctc_th_in/ ctc_th_out=10，那么当解 67hz 时候，对应的阈值为 10*67*0.1%=0.67hz

可以通过定时读 dcs_detect 和 ctc_detect 的值来检测亚音频，也可以通过中断来获取。

CONFIDENTIAL

声控（VOX）、发射超时（TOT）设置

声控（VOX）的判断采用双阈值算法，其判断逻辑如下：

```
if (vox_amp > vox_amp_th_in)
        VOX = 1;
else
if (vox_amp < vox_amp_th_out) && delay_out
        VOX = 0;
```

附：VOX 相关寄存器

Register	Address(HEX)	Description
vox_en	REG_04H[2]	Enable VOX detection
vox_delay[3:0]	REG_7AH[15:12]	VOX=0 delay, *128ms
vox_rssi_th [5:0]	REG_46H[15:10]	RSSI threshold (2dB/step) for VOX. VOX works only when RSSI is lower than this threshold.
vox_amp_th_in[9:0]	REG_46H[9:0]	voice amp threshold for VOX=1 detect
vox_amp_th_out[9:0]	REG_79H[9:0]	voice amp threshold for VOX=0 detect
vox_amp[9:0]	REG_64H[9:0]	voice amp out. (Read Only)
vox_out	REG_0CH[2]	VOX result output. (Read Only)
tot_en	REG_04H[6]	Enable TOT detection
tot_timer[7:0]	REG_31H[7:0]	tot timer, *327ms

CONFIDENTIAL

静噪（SQ）设置及 RSSI、NOISE 和 SNR

静噪判断采用 rssi_sq、noise_sq 双阈值算法，其判断逻辑如下：

```
if (rssi_sq > rssi_sq_th_in  && noise_sq < noise_sq_th_in)
        SQ = 1;
else
if (rssi_sq < rssi_sq_th_out || noise_sq > noise_sq_th_out)
        SQ = 0;
```

不同的静噪等级用 rssi_sq 的阈值来进行区分，noise_sq 的阈值可以设置为固定值，这样静噪表格比较简单。

提示：不同方案可能会有不同增益的外置 LNA，为了 RSSI 归一化，所以设置了 ext_lna_gain 寄存器进行补偿。如外置 LNA 增益为 10dB，那么需要设置 ext_lna_gain=10。

附：SQ 相关寄存器

Register	Address(HEX)	Description
rssi_sq_th_in[7:0]	REG_78H[15:8]	RSSI threshold for SQ=1, 0.5dB/step
rssi_sq_th_out[7:0]	REG_78H[7:0]	RSSI threshold for SQ=0, 0.5dB/step
noise_sq_th_out[6:0]	REG_4FH[14:8]	noise threshold for SQ=0
noise_sq_th_in[6:0]	REG_4FH[6:0]	noise threshold for SQ=1
ext_lna_gain[4:0]	REG_2CH[4:0]	External LNA gain RSSI, 1dB/step
rssi_sq[8:0]	REG_67H[8:0]	0.5dB/step, RSSI (dB) = rssi_sq/2 – 160. (Read Only)
snr_out[7:0]	REG_61H[15:8]	SNR indicator, dB/step. (Read Only)
rssi_rel[7:0]	REG_65H[15:8]	RSSI relative, dB/step. (Read Only)
noise_sq[7:0]	REG_65H[7:0]	NOISE indicator, dB/step. (Read Only)
sq_out	REG_0CH[1]	SQ result output. (Read Only)
weak_rssi	REG_0CH[7]	1=signal is too weak. (Read Only)

CONFIDENTIAL

SOFT MUTE 设置

附：SOFT MUTE 寄存器

Register	Address(HEX)	Description
soft_mute_en	REG_90H[12]	1=Enable Soft Mute
soft_mute_atten[1:0]	REG_90H[9:8]	Soft Mute Atten Level 00=-16dB; 01=-12dB; 10=-8dB; 11=-4dB
soft_mute_rate[1:0]	REG_90H[7:6]	Soft Mute Rate (SNR/GAIN) 00=1/4; 01=1/2; 10=1; 11=2
snr_th_for_sm[5:0]	REG_90H[5:0]	SNR Threshold for Soft Mute. If SNR little than this value, Soft Mute begin.
snr_out[7:0]	REG_61H[15:8]	SNR indicator, dB/step. (Read Only)

当 snr< snr_th_for_sm && weak_rssi && soft_mute_en 时，芯片内部才开起 soft_mute 功能。此功能有利于降低弱信号下接收音频底噪。

CONFIDENTIAL

AFC 设置

如果使用的是温补晶振 TCXO，频率比较准，可以关闭 AFC 功能（设置 afc_disable=1）；如果使用的是晶体，频率校准后仍有一定的偏差，可以打开 AFC（设置 afc_disable=0）并设置合适的 afc_range。

附：AFC 寄存器

Register	Address(HEX)	Description
afc_range[2:0]	RegW_73H[13:11]	AFC Range: 000: ~=2.2 kHz 001: ~=1.5 kHz 010: ~=1.1 kHz 011: ~=750 Hz 100: ~=550 Hz 101: ~=375 Hz 110: ~=275 Hz 111: ~=188 Hz if wb, afc range*=2
afc_disable	RegW_73H[4]	1=disable AFC
afc_rail	RegW_0CH[8]	0 = AFC not railed 1 = AFC railed (Read Only)

CONFIDENTIAL

GPIO、中断设置

附：INTERRUPT 相关寄存器

Register	Address(HEX)	Description
int_out	REG_0CH[0]	Interrupt output
irq_mask[15:0]	REG_3FH[15:0]	[15]: Enable FSK Tx finished interrupt [14]: Enable FSK FIFO almost empty interrupt [13]: Enable FSK Rx succeed interrupt [12]: Enable FSK FIFO almost full interrupt [11]: Enable FSK Header received interrupt [10]: Enable FSK SyncP succeed interrupt [9]: Enable FSK SyncN succeed interrupt [8]: Enable DTMF/SELCALL code received interrupt [7]: Enable PLL lock lost interrupt [6]: Enable CDCSS receive/lost interrupt [5]: Enable CTCSS receive/lost interrupt [4:3]: reserved [2]: Enable TOT time out interrupt [1]: Enable VOX receive/lost interrupt [0]: Enable SQ receive/lost interrupt
irq_vector[15:0]	REG_02H[15:12]	dtmf/selcall code index[3:0]
	REG_02H[11:10]	10=CDCSS positive code receive 11=CDCSS negative code receive 00=CDCSS positive code lost 01=CDCSS negative code lost
	REG_02H[9:8]	reserved
	REG_02H[7:6]	10=CTCSS0 receive interrupt 00=CTCSS0 lost interrupt 11=CTCSS1 receive interrupt 01=CTCSS1 lost interrupt
	REG_02H[5]	0=VOX lost 1=VOX receive
	REG_02H[4]	0=SQ lost

CONFIDENTIAL

		1=SQ receive
	REG_02H[3:0]	15=FSK Tx Finished 14=FSK FIFO(almost empty) need to WRITE 13=FSK Rx Succeed 12=FSK FIFO(almost full) need to read 11=FSK Header received interrupt 10=FSK SyncP found interrupt 9=FSK SyncN found interrupt 8=DTMF/SELCALL receive interrupt 7=PLL lock lost interrupt 6=CDCSS receive/lost interrupt 5=CTCSS receive/lost interrupt 4,3=reserved 2=TOT time out interrupt 1=VOX receive/lost interrupt 0=SQ receive/lost interrupt

中断可由芯片任意 GPIO 口输出，中断通过对 02H 寄存器写任意值来清除，如

```
WRITE (0x02,0x0000); //clear interrupt
```

提示：中断高电平有效，得到中断时，要先清除中断，才能去读取中断向量表。读到向量表，先根据 irq_vector[3:0]来判断中断源，然后再找该中断源里的具体中断事件。处理如下：

```
int rdata = 0;
if (PIN_INT) { // MCU 读取芯片的中断输出 PIN_INT
    WRITE(0x02,0x0000);
    Read(0x02,rdata);
    switch (rdata & 0xF) {
        case 0: SQ = (rdata >> 4) & 1; break;
        case 1: VOX = (rdata >> 5) & 1; break;
        case 2: break; //TOT 发射超时
```

CONFIDENTIAL

```
                case 3: break; //reserved
                case 5:
                        switch ((rdata >> 6) & 1) {
                                case 0: CTC0 = (rdata >> 7) & 1; break;
                                case 1: CTC1 = (rdata >> 7) & 1; break;
                        }
                case 6:
                        switch ((rdata >>10) & 1) {
                                case 0: DCSP = (rdata >> 11) & 1; break; //DCS 正码
                                case 1: DCSN = (rdata >> 11) & 1; break; //DCS 反码
                        }
                case 7: break; //PLL 失锁，可以重新触发 PLL，方法是在接收或发射状态下先
REG_03H[15]=0 后再 REG_03H[15]=1
                case 8: CODE = (rdata >>12) & 0xf; break; //DTMF or SELCALL CODE
                case 9: break; //FSK SyncN Found
                case 10: break; //FSK SyncP Found
                case 11: break; //FSK Header Found
                case 12: break; //Read Data from FIFO
                case 13: break; //FSK Rx Succeed
                case 14: break; //WRITE Data to FIFO
                case 15: break; //FSK Tx Finished
        } //end switch
} //end if
```

CONFIDENTIAL

GPIO、中断设置

附：GPIO 寄存器设置

Register	Address(HEX)	Description
gpio_oen_b[7:0]	REG_25H[15:8]	gpioX output enable, low active, (X=0..7)
gpio_out_val[7:0]	REG_25H[7:0]	gpioX output value when gpioX_out_sel=0, (X=0..7)
gpio7_out_sel[3:0]	REG_27H[15:12]	gpio7 ouput select. 0=gpio_out_val[X], (X=0..7) 1=INT 2=SQ 3=VOX 4=subau_cmp (CTCSS/CDCSS compare result output) 5=CTCSS/CDCSS code output for software decode 6=SDO for 4-wire mode 7:11=reserved ~~12=I2S/MCBSP DOUT output~~ ~~13=I2S/MCBSP DFS output~~ ~~14=I2S/MCBSP DCLK output~~ ~~15=XTAL CLK div2 output~~
gpio6_out_sel[3:0]	REG_27H[11:8]	Description is the same as gpio7_out_sel[3:0]
gpio5_out_sel[3:0]	REG_27H[7:4]	Description is the same as gpio7_out_sel[3:0]
gpio4_out_sel[3:0]	REG_27H[3:0]	Description is the same as gpio7_out_sel[3:0]
gpio3_out_sel[3:0]	REG_26H[15:12]	Description is the same as gpio7_out_sel[3:0]
gpio2_out_sel[3:0]	REG_26H[11:8]	Description is the same as gpio7_out_sel[3:0]
gpio1_out_sel[3:0]	REG_26H[7:4]	Description is the same as gpio7_out_sel[3:0]
gpio0_out_sel[3:0]	REG_26H[3:0]	Description is the same as gpio7_out_sel[3:0]
gpio_in_val[7:0]	REG_28H[15:8]	gpioX input value, (X=0..7). (Read Only)

提示： GPIO0~GPIO7 均有下拉。

CONFIDENTIAL

例如：用 GPIO4 做 SQ 输出，用 GPIO5 做 VOX 输出，则需要设置

gpio_oen_b[4]=0, gpio4_out_sel[3:0]=2;

gpio_oen_b[5]=0, gpio5_out_sel[3:0]=3;

例如：用 GPIO4 做可变输出，则需要设置

gpio_oen_b[4]=0, gpio4_out_sel[3:0]=0, GPIO4 的输出值为 gpio_out_val[4]对应的值。

例如：用 GPIO5 做输入，则需要设置

gpio_oen_b[5]=1, 读取 gpio_in_val[5]的值即为 GPIO5 的输入值。

CONFIDENTIAL

晶体/晶振时钟频率设置

XTAL 为时钟频率（单位 Hz），那么 N=XTAL/10*2，晶体/晶振的时钟频率设置如下：

WRITE(0x3B,N);

WRITE(0x3C,(N>>16)<<8);

除时钟频率为，如果时钟频率在 12M~13M 范围内，则需设置：

WRITE(0x22,0x9E14);

WRITE(0x41,0x81C1);

WRITE(0x3D,0x4EC5);

如果时钟频率在 18M~20M 范围内，则需设置：

WRITE(0x22,0x5E14);

WRITE(0x41,0x81C2);

WRITE(0x3D,0x3483);

如果时钟频率在 24M~26M 范围内，则需设置：（默认 26MHz，可不用设置）

WRITE(0x22,0x3E14);

WRITE(0x41,0x81C4);

WRITE(0x3D,0x2762);

CONFIDENTIAL

寄存器汇总

Address	R/W	Default	Description		
REG_00H	R	0x6818	chip_id[15:0]	[15:0]	Chip identification code
REG_01H	R	0x0000	revision_id[15:0]	[15:0]	Revision identification code
REG_02H	R		irq_vector[15:0]	[15:12]	dtmf/selcall code index[3:0]
				[11:10]	10=CDCSS positive code receive 11=CDCSS negative code receive 00=CDCSS positive code lost 01=CDCSS negative code lost
				[9:8]	reserved
				[7:6]	10=CTCSS0 receive interrupt 00=CTCSS0 lost interrupt 11=CTCSS1 receive interrupt 01=CTCSS1 lost interrupt
				[5]	0=VOX lost 1=VOX receive
				[4]	0=SQ lost 1=SQ receive
				[3:0]	15=FSK Tx Finished 14=FSK FIFO(almost empty) need to WRITE 13=FSK Rx Succeed 12=FSK FIFO(almost full) need to read 11=FSK Header received interrupt 10=FSK SyncP found interrupt 9=FSK SyncN found interrupt 8=PLL lock lost interrupt 6=CDCSS receive/lost interrupt

CONFIDENTIAL

					5=CTCSS receive/lost interrupt 4=DTMF/SELCALL receive interrupt 3=reserved 2=TOT time out interrupt 1=VOX receive/lost interrupt 0=SQ receive/lost interrupt
REG_03H	R/W	0x0000	vco_cal_en	[15]	0->1=Enable PLL Calibration
			pabias_en	[14]	1=Enable PABIAS
			rxlink_en[3:0]	[13:10]	1111=Enable Rx Link, (LNA,MIXER,FILTER,ADC)
			afout_en	[9]	1=Enable AFOUT DAC
			pll_en[4:0]	[8:4]	11111=Enable PLL, VCO
			padrv_en	[3]	1=Enable PADRV
			micin_en	[2]	1=Enable MICIN ADC
			txon	[1]	1=Enable Tx DSP
			rxon	[0]	1=Enable Rx DSP
REG_04H	R/W	0x0000	reserved	[15:10]	Reserved
			adc_cal_en	[9]	0->1=Enable ADC calibration
			offset_cal_en	[8]	0->1=Enable offset calibration
			pkd_disable	[7]	1=Disable Peak Detection
			tot_en	[6]	1=Enable TOT detection
			scramb_en	[5]	1=Enable SCRAMBLER
			fsk_en	[4]	1=Enable FSK mode
			~~amdem_en~~	~~[3]~~	~~1=Enable AM Demodulation~~
			vox_en	[2]	1=Enable VOX detection
			dtmf_en	[1]	1=Enable DTMF/SELCALL
			~~mcbsp_en~~	~~[0]~~	~~1=Enable MCBSP/I2S module~~
REG_07H	W		ctc_freq[15:0]	[15:0]	[13]=1 for CTCSS1/CDCSS [13]=0 for CTCSS0/CDCSS [12:0]: CTCSS frequency control word = freq(Hz)*2^27/6500000

CONFIDENTIAL

REG_08H	W		dcs_code[15:0]	[15:0]	[15]=1 for CDCSS high 12bit [15]=0 for CDCSS low 12bit [11:0] CDCSS 23/24bit code
REG_09H	W		dtmf_coef [15:0]	[15:0]	[15:12]=coefficient index [7:0]=coefficients for DTMF/SELCALL detection
REG_0BH	R		reserved	[15:13]	Reserved
			dtmf_code_ready	[12]	1=DTMF/ SELCALL symbol received ready.
			dtmf_code[3:0]	[11:8]	DTMF/ SELCALL symbol received.
			reserved	[7:0]	Reserved
REG_0CH	R		reserved	[15:10]	Reserved
			pll_lock	[9]	1=pll lock 0=pll lock lost
			afc_rail	[8]	0 = AFC not railed 1 = AFC railed
			weak_rssi	[7]	1=signal is too weak.
			reserved	[6:4]	Reserved
			tot_out	[3]	TOT output
			vox_out	[2]	VOX result output.
			sq_out	[1]	SQ result output.
			int_out	[0]	Interrupt output
REG_18H	R/W	0x4125	mic_pga_gain[4:0]	[15:11]	10000 max 01000(default) 00100 00010 00001 min
			reserved	[10:0]	Reserved
REG_19H	R/W	0x900F	reserved	[15:4]	Reserved
			pabias_out[3:0]	[3:0]	PAbias output 1.3~2.8V, 100mv/step 0000=1.3V … 1111=2.8V

CONFIDENTIAL

REG_23H	R/W	0x8F8F	reserved	[15:4]	Reserved
			dac_vgain[3:0]	[3:0]	DAC maximum output level
~~REG_24H~~	~~R/W~~	~~0x603F~~	~~reserved~~	~~[15:12]~~	~~Reserved~~
			~~mcbsp_clks_in_sel[2:0]~~	~~[11:9]~~	~~Select GPIOx as I2S/MCBSP DCLKS input (x=0..7)~~
			~~mcbsp_clk_in_sel[2:0]~~	~~[8:6]~~	~~Select GPIOx as I2S/MCBSP DCLK input (x=0..7)~~
			~~mcbsp_fs_in_sel[2:0]~~	~~[5:3]~~	~~Select GPIOx as I2S/MCBSP DFS input (x=0..7)~~
			~~mcbsp_dr_in_sel[2:0]~~	~~[2:0]~~	~~Select GPIOx as I2S/MCBSP DIN input (x=0..7)~~
REG_25H	R/W	0x0000	gpio_oen_b[7:0]	[15:8]	gpioX output enable, low active, (X=0..7)
			gpio_out_val[7:0]	[7:0]	gpioX output value when gpioX_out_sel=0, (X=0..7)
REG_26H	R/W	0x0005	gpio3_out_sel[3:0]	[15:12]	Gpio3 ouput select. 0=gpio_out_val[X], (X=0..7) 1=INT 2=SQ 3=VOX 4=subau_cmp (CTCSS/CDCSS compare result output) 5=CTCSS/CDCSS code output for software decode 6=SDO for 4-wire mode 7:11=reserved 12=I2S/MCBSP DOUT output 13=I2S/MCBSP DFS output 14=I2S/MCBSP DCLK output 15=XTAL CLK div2 output
			gpio2_out_sel[3:0]	[11:8]	Description is the same as gpio3_out_sel[3:0]
			gpio1_out_sel[3:0]	[7:4]	Description is the same as gpio3_out_sel[3:0]
			gpio0_out_sel[3:0]	[3:0]	Description is the same as

CONFIDENTIAL

					gpio3_out_sel[3:0]
REG_27H	R/W	0x3200	gpio7_out_sel[3:0]	[15:12]	Description is the same as gpio3_out_sel[3:0]
			gpio6_out_sel[3:0]	[11:8]	Description is the same as gpio3_out_sel[3:0]
			gpio5_out_sel[3:0]	[7:4]	Description is the same as gpio3_out_sel[3:0]
			gpio4_out_sel[3:0]	[3:0]	Description is the same as gpio3_out_sel[3:0]
REG_28H	R/W	0x003F	gpio_in_val[7:0]	[15:8]	gpioX input value, (X=0..7). (Read Only)
			reserved	[7:6]	Reserved
			padrv_gain[2:0]	[5:3]	PA output gain
			pa_gain_vreg[2:0]	[2:0]	PA output gain
REG_38H	R/W	0x3A98	freq [15:0]	[15:0]	Frequency =(REG_39H<<16 + REG_38H)*10 Hz
REG_39H	R/W	0x0271	freq [31:16]	[15:0]	
REG_3BH	R/W	0x3D62	xtal_freq [15:0]	[15:0]	XTAL Frequency =((REG_3CH>>8)<<16 + REG_3BH)*5 Hz
REG_3CH	R/W	0x1000	xtal_freq [23:16]	[15:8]	
			reserved	[7:0]	Reserved.
REG_3FH	R/W	0x0000	irq_mask[15:0]	[15:0]	[15]: Enable FSK Tx finished interrupt [14]: Enable FSK FIFO almost empty interrupt [13]: Enable FSK Rx succeed interrupt [12]: Enable FSK FIFO almost full interrupt [11]: Enable FSK Header received interrupt [10]: Enable FSK SyncP succeed interrupt [9]: Enable FSK SyncN succeed interrupt [8]: Enable DTMF/SELCALL

CONFIDENTIAL

					code received interrupt [7]: Enable PLL lock lost interrupt [6]: Enable CDCSS receive/lost interrupt [5]: Enable CTCSS receive/lost interrupt [4:3]: reserved [2]: Enable TOT time out interrupt [1]: Enable VOX receive/lost interrupt [0]: Enable SQ receive/lost interrupt
REG_40H	R/W	0x3808	reserved	[15:13]	Reserved
			dev_en	[12]	Enable FM deviation
			dev_sh[3:0]	[11:8]	FM deviation coarse tuning， 0000=max, 1111=min
			dev_lvl[7:0]	[7:0]	FM deviation fine tuning， 0000=min, 1111=max, GAIN=(256+dev_lvl[7:0])>>dev_sh[3:0]
REG_43H	R/W	0x6009	reserved	[15]	Reserved
			firlpf_bw[2:0]	[14:12]	RF filter bandwidth (Apass=0.1dB) 000 = 1.7 kHz 001 = 2 kHz 010 = 2.5 kHz 011 = 3 kHz 100 = 3.75 kHz 101 = 4 kHz 110 = 4.25 kHz 111 = 4.5 kHz if wb=1, firlpf_bw *=2;
			firlpf_bw_for_weak[2:0]	[11:9]	RF filter bandwidth when signal is weak.

CONFIDENTIAL

			audio_lpf2_tx_sel[2:0]	[8:6]	Audio LPF bandwidth (Apass=1dB) for Tx 100 = 4.5 kHz 101 = 4.25 kHz 110 = 4 kHz 111 = 3.75 kHz 000 = 3 kHz 001 = 2.5 kHz 010 = 2 kHz 011 =1.7kHz
			reserved	[5:2]	Reserved
			firlpf_gain[1:0]	[1:0]	Gain after FIR LPF 00=0dB; 01=6dB; 10=12dB; 11=18dB
REG_44H	R/W	0x43EC	reserved	[15:13]	Reserved
			fmdem_gain[1:0]	[12:11]	Gain after FM Demodulation 00=0dB; 01=6dB; 10=12dB; 11=18dB
			reserved	[10:0]	Reserved
REG_46H	R/W	0x8050	vox_rssi_th[15:10]	[15:10]	RSSI threshold (2dB/step) for VOX. VOX works only when RSSI is lower than this threshold.
			vox_amp_th_in[9:0]	[9:0]	voice amp threshold for VOX=1 detect
REG_47H	R/W	0x6140	reserved	[15]	Reserved
			reserved	[14]	Reserved
			afout_invert	[13]	1 = invert AFOUT
			afout_mode[4:0]	[12:8]	0x10 = MUTE 0x11 = RX AFOUT 0x18 = BEEP/TX Side Tone （CTCSS/CDCSS is not include） 0x15 = RX ALARM TONE
			reserved	[7:4]	Reserved

CONFIDENTIAL

			dig_gain_tx[3:0]	[3:0]	Digital gain after MIC ADC, 1dB/step
REG_4FH	R/W	0x7E34	reserved	[15]	Reserved
			noise_sq_th_out[6:0]	[14:8]	noise threshold for SQ=0
			reserved	[7]	Reserved
			noise_sq_th_in[6:0]	[6:0]	noise threshold for SQ=1
REG_50H	R/W	0x033C	audio_tx_mute	[15]	1=Audio Tx Mute
			reserved	[14:11]	Reserved
			audio_tx_limit_bypass	[10]	1=Audio Tx Limit bypass
			audio_tx_limit[9:0]	[9:0]	Audio Tx Limit Value
REG_51H	R/W	0x1050	subau_en	[15]	1=Enable CTCSS/CDCSS
			reserved	[14]	Reserved
			dcs_invert	[13]	1=Transmit negative CDCSS code 0=Transmit positive CDCSS code
			ctc_dcs_sel	[12]	CTCSS/CDCSS mode select, 1=CTCSS, 0=CDCSS
			dcs_24b_mode	[11]	24/23bit CDCSS select, 1=24bit, 0=23bit
			reserved	[10:7]	Reserved
			subau_gain[6:0]	[6:0]	[6:5]: CTCSS/CDCSS coarse tuning gain [4:0]: CTCSS/CDCSS fine tuning gain
REG_52H	R/W	0x028F	subau_tail_gen	[15]	sub-audio tail generate 0=normal, 1=tail generate
			ctc_tail_offs[1:0]	[14:13]	CTCSS tail mode and phase change select, 00=No phase shift（generate CTCSS1） 01=CTCSS0 120°phase shift, 10= CTCSS0 180°phase shift 11= CTCSS0 240°phase shift CTCSS Phase Decode is not

CONFIDENTIAL

					supported.
			ctc_th_mode	[12]	ctcss detect threshold mode, 1=~0.1%; 0=0.1 Hz
			ctc_th_in[5:0]	[11:6]	CTCSS found detect threshold
			ctc_th_out[5:0]	[5:0]	CTCSS lost detect threshold
REG_53H	R/W	0x97DD	audio_rx_gain_sh [2:0]	[15:13]	111=max,000=min 0~-42dB, 6dB/step, digital gain
			audio_tx_gain_sh [2:0]	[12:10]	111=max,000=min 0~-42dB, 6dB/step, digital gain
			audio_rx_gain[4:0]	[9:5]	0=mute,31=max,1dB/step, digital gain
			audio_tx_gain[4:0]	[4:0]	0=mute,31=max,1dB/step, digital gain
REG_54H	R/W	0x8517	scramb_freq[15:0]	[15:0]	SCRAMBLE/FSK frequency control word =3.3(KHz)*2^26/6500 - The scrambler inversion mixing frequency should be kept between 2.6kHz and 3.5kHz
REG_56H	R/W	0x1021	fsk_crc_polyn[15:0]	[15:0]	CRC polynomial coefficient, CCIT-16 is default
REG_59H	R/W	0x8078	fsk_tx_fifo_clear	[15]	Clear TX FIFO, 1=clear
			fsk_rx_fifo_clear	[14]	Clear RX FIFO, 1=clear
			fsk_scramble_en	[13]	1=Enable FSK Scramble
			fsk_rx_en	[12]	1=Enable FSK TX
			fsk_tx_en	[11]	1=Enable FSK RX
			fsk_rx_data_inv	[10]	1=invert FSK data before TX
			fsk_tx_data_inv	[9]	1=invert FSK data after RX
			reserved	[8]	Reserved
			fsk_prmb_size[3:0]	[7:4]	FSK preamble length select, length=(fsk_prmb_size[3:0]+1) bytes
			fsk_sync_size	[3]	FSK sync length select, 1=4 bytes

CONFIDENTIAL

					(fsk_sync_byte0,1,2,3) 0=2 bytes (fsk_sync_byte0,1)
			reserved	[2:0]	Reserved
REG_5AH	R/W	0x85CF	fsk_sync_byte0[7:0]	[15:8]	FSK sync byte0
			fsk_sync_byte1[7:0]	[7:0]	FSK sync byte1
REG_5BH	R/W	0xAB45	fsk_sync_byte2[7:0]	[15:8]	FSK sync byte2
			fsk_sync_byte3[7:0]	[7:0]	FSK sync byte3
REG_5CH	R/W	0x56F9	reserved	[15:8]	Reserved
			fsk_tx_type_byte[7:0]	[7:0]	FSK type byte to be transmitted. "0x01 "=FSK Format2 "0x21" =FSK Format1 without CRC "0x61" =FSK Format1 with CRC
REG_5DH	R/W	0x3FCC	fsk_length[7:0]	[15:8]	FSK data length when use FSK format1 length=(fsk_length[7:0]+1) bytes
			reserved	[7:0]	Reserved
REG_5EH	R/W	0x0004	reserved	[15:12]	Reserved
			fsk_tx_fifo_ae_th[5:0]	[11:6]	FSK TX FIFO almost empty threshold(unit is 1word=2 bytes)
			fsk_rx_fifo_af_th[5:0]	[5:0]	FSK RX FIFO almost full threshold(unit is 1word=2 bytes)
REG_5FH	R/W		fsk_data[15:0]	[15:0]	TX/RX FIFO Data
REG_60H	R		reserved	[15:12]	Reserved
			dcs_detect[1:0]	[11:10]	[0]:CDCSS positive code received [1]:CDCSS negative code received
			reserved	[9:2]	Reserved
			ctc_detect[1:0]	[1:0]	[1]:CTCSS1 received

CONFIDENTIAL

					[0]:CTCSS0 received
REG_61H	R		snr_out[7:0]	[15:8]	SNR indicator, dB/step.
			reserved	[7:0]	Reserved
REG_62H	R		agc_rssi[7:0]	[15:8]	RSSI after DCC, 1dB/step.
			lna_peak_rssi[7:0]	[7:0]	RSSI after LNA, 1dB/step.
REG_64H	R		reserved	[15:10]	Reserved
			vox_amp[9:0]	[9:0]	voice amp out.
REG_65H	R		rssi_rel[7:0]	[15:8]	RSSI relative, dB/step
			noise_sq[7:0]	[7:0]	NOISE indicator, dB/step.
REG_67H	R		reserved	[15:9]	Reserved
			rssi_sq[8:0]	[8:0]	0.5dB/step, RSSI (dB) = rssi_sq/2 – 160.
REG_70H	R/W	0x7070	tone1_gen	[15]	Enable TONE1
			tone1_gain[6:0]	[14:8]	TONE1 tuning gain
			tone2_gen	[7]	Enable TONE2
			tone2_gain[6:0]	[6:0]	TONE2 tuning gain
REG_71H	R/W	0x2854	tone1_freq[15:0]	[15:0]	TONE1 frequency· control word =freq(kHz)* 2^26/6500
REG_72H	R/W	0x3065	tone2_freq[15:0]	[15:0]	TONE2 frequency· control word =freq(kHz)* 2^26/6500
REG_73H	R/W	0x568A	reserved	[15:14]	Reserved
			afc_range[2:0]	[13:11]	AFC Range: 000: ~=2.2 kHz 001: ~=1.5 kHz 010: ~=1.1 kHz 011: ~=750 Hz 100: ~=550 Hz 101: ~=375 Hz 110: ~=275 Hz 111: ~=188 Hz if wb, afc range*=2
			reserved	[10:5]	Reserved

CONFIDENTIAL

			afc_disable	[4]	1=disable AFC
			reserved	[3:0]	Reserved
~~REG_75H~~	~~R/W~~	~~0x0FB4~~	~~resreved~~	~~[15]~~	~~Reserved~~
			~~mcbsp_test~~	~~[14]~~	~~0=normal;~~ ~~1=Transmit data equals to REG_78H~~
			~~reserved~~	~~[13]~~	~~Reserved~~
			~~mcbsp_fwid[4:0]~~	~~[12:8]~~	~~Transmit/Receive frame-synchronization width.~~
			~~mcbsp_flen~~	~~[7]~~	~~Transmit/Receive frame-length. 1=32 bit; 0=16 bit~~
			~~mcbsp_delay~~	~~[6]~~	~~0=1 data delay; 1=2 data delay~~
			~~mcbsp_master~~	~~[5]~~	~~0=slave mode; 1=master mode~~
			~~mcbsp_clks_master~~	~~[4]~~	~~1= mcbsp clk source master mode;~~ ~~0= mcbsp clk source slave mode~~
			~~mcbsp_clks_psel~~	~~[3]~~	~~mcbsp clk source polarity select.~~
			~~mcbsp_clks_enable~~	~~[2]~~	~~mcbsp clk source enable~~
			~~mcbsp_clk_psel~~	~~[1]~~	~~Transmit/Receive clock polarity bit. It determines the polarity of DCLK as seen on the GPIO pin.~~ ~~0=Transmit/Receive data is sampled on the rising edge of DCLK.~~ ~~1=Transmit/Receive data is sampled on the falling edge of DCLK.~~
			~~mcbsp_fs_psel~~	~~[0]~~	~~Transmit/Receive frame-synchronization polarity bit. It determines the polarity of DFS as seen on the GPIO pin.~~ ~~0=Transmit/Receive frame-synchronization pulses are~~

CONFIDENTIAL

					~~active high.~~ ~~1=Transmit/Receive frame synchronization pulses are active low.~~
~~REG_76H~~	~~R/W~~	~~0xE307~~	~~mcbsp_phase_swap[1:0]~~	~~[15:14]~~	~~MCBSP/I2S LR data swap for 32 bit mode.~~ ~~[1]: for Tx; [0]: for Rx~~
			~~reserved~~	~~[13:12]~~	~~Reserved~~
			~~mcbsp_dsel~~	~~[11]~~	~~MCBSP/I2S data output select.~~ ~~1=IQ; 0=FM Demodulated~~
			~~reserved~~	~~[10:9]~~	~~Reserved~~
			~~mcbsp_fifo_bypass~~	~~[8]~~	~~MCBSP/I2S FIFO bypass. FIFO is not needed in master mode, set "1" to bypass FIFO which can reduce time on the path.~~
			~~mcbsp_clkgdv[7:0]~~	~~[7:0]~~	~~MCBSP/I2S generator divide number.~~
REG_78H	R/W	0x44FF	rssi_sq_th_in[7:0]	[15:8]	RSSI threshold for SQ=1, 0.5dB/step
			rssi_sq_th_out[7:0]	[7:0]	RSSI threshold for SQ=0, 0.5dB/step
REG_79H	R/W	0x1040	reserved	[15:10]	Reserved
			vox_amp_th_out[9:0]	[9:0]	voice amp threshold for VOX=0 detect
REG_7AH	R/W	0x881A	vox_delay[3:0]	[15:12]	VOX=0 delay, *128ms
			reserved	[11:0]	Reserved
REG_7DH	R/W	0x4200	reserved	[15:13]	Reserved
			dig_gain_rx[4:0]	[12:8]	Gain after AGC, digital down conversion. 1dB/step
			reserved	[7:0]	Reserved
REG_7EH	R/W	0x341E	reserved	[15:8]	Reserved
			dcc_tx_bypass	[7]	1=MICIN DC Filter bypass
			dcc_rx_bypass	[6]	1=Rx DC Filter bypass
			dcc_tx_bw[2:0]	[5:3]	MICIN DC Filter bandwidth(3dB) select 111=15Hz;110=30Hz;101=60H

CONFIDENTIAL

					z;100=120Hz; 011=240Hz;010=480Hz;
			reserved	[2:0]	Reserved
REG_87H	R		reserved	[15:7]	Reserved
			cmpd_db_out[6:0]	[6:0]	Audio amplitude output, 1dB/step.
REG_90H	R/W	0x8C20	reserved	[15:13]	Reserved
			soft_mute_en	[12]	1=Enable Soft Mute
			reserved	[11:10]	Reserved
			soft_mute_rate[1:0]	[9:8]	Soft Mute Rate (SNR/GAIN) 00=1/4; 01=1/2; 10=1; 11=2
			soft_mute_atten[1:0]	[7:6]	Soft Mute Atten Level 00=-16dB; 01=-12dB; 10=-8dB; 11=-4dB
			snr_th_for_sm[5:0]	[5:0]	SNR Threshold for Soft Mute. If SNR little than this value, Soft Mute begin.
REG_94H	R/W	0x8F5E	reserved	[15:13]	Reserved
			dtmf_det_th[5:0]	[12:7]	DTMF/ SELCALL detect threshold, 1dB/step
			dtmf_symbol_mode	[6]	DTMF/ SELCALL symbol mode 1=" symbol" +" idle" +" symbol" +... (DTMF) 0=" symbol" +" symbol" +" symbol" +...(5TONE)
			dtmf_in_sel	[5]	1=select gain out 0=select audio rx out
			dtmf_mode	[4]	Dual/Single Tone Detect mode 1=dual tone 0=single tone
			dtmf_symbol_max[3:0]	[3:0]	SELCALL maximum symbol number 15=symbol "0" ~" F" (DTMF) 14=symbol "0" ~" E" (EIA)

CONFIDENTIAL

					...
REG_96H	R/W	0x1AE8	reserved	[15:14]	
			audio_rx_gain1[5:0]	[13:8]	0=mute, 63=max, 0.5dB/step
			cmpd_rx_gain[4:0]	[7:3]	Audio CMPD Gain, 1dB/step
			cmpd_rx_bypass[2:0]	[2:0]	Audio CMPD bypass for RX 110=bypass with gain 001=bypass without gain 000=CMPD ON for RX Others=reserved
REG_97H	R/W	0x0AA8	reserved	[15:14]	Reserved
			~~mic_sens_gain[5:0]~~	~~[13:8]~~	~~MIC sensitivity adjust gain 0=mute, 63=max, 0.5dB/step~~
			cmpd_tx_gain[4:0]	[7:3]	Audio CMPD Gain, 0.5dB/step
			cmpd_tx_bypass[2:0]	[2:0]	Audio CMPD bypass for TX 110=bypass with gain 001=bypass without gain 000=CMPD ON for TX Others=reserved
REG_98H	R/W	0x7A14	reserved	[15]	Reserved
			cmpd_rx_factor[2:0]	[14:12]	Expanding Factor. 100=1:3;011=1:2.5;010=1:2;001=1:1.5
			cmpd_rx_th_high[5:0]	[11:6]	Above this amplitude point, audio will be expanded according to the expanding factor. The unit of this threshold is 2dB.
			cmpd_rx_th_low[5:0]	[5:0]	Under this amplitude point, audio will be attenuated. The unit of this threshold is 2dB.
REG_99H	R/W	0x7855	reserved	[15]	Reserved
			cmpd_tx_factor[2:0]	[14:12]	Compressing Factor. 111=8:1;110=4:1;100=2:1;000=1:1
			cmpd_tx_th_high[5:0]	[11:6]	Above this amplitude point,

CONFIDENTIAL

					audio will be compressed according to the compressing factor. The unit of this threshold is 2dB.
			cmpd_tx_th_low[5:0]	[5:0]	Under this amplitude point, audio will be attenuated. The unit of this threshold is 2dB.
REG_9AH	R/W	0x0728	reserved	[15:14]	Reserved
			cmpd_ct_intvl[5:0]	[13:8]	Companding Amplitude Detect Interval, 0.64ms/step
			cmpd_atk_step[3:0]	[7:4]	Companding Gain Attack Speed. 0000=most fast … 1111=most slow
			cmpd_rls_step[3:0]	[3:0]	Companding Gain Release Speed. 0000=most fast … 1111=most slow
REG_9BH	R/W	0x0004	reserved	[15:10]	Reserved
			audio_hpf_rx_bypass	[9]	Audio HPF 300Hz bypass for RX
			audio_lpf1_rx_bypass	[8]	Audio LPF1 bypass for RX
			audio_lpf2_rx_bypass	[7]	Audio LPF2 bypass for RX
			audio_emph_rx_bypass	[6]	DE-EMPHASIS bypass
			reserved	[5:4]	Reserved
			audio_hpf_tx_bypass	[3]	Audio HPF 300Hz bypass for TX
			audio_lpf1_tx_bypass	[2]	Audio LPF1 bypass for TX
			audio_lpf2_tx_bypass	[1]	Audio LPF2 bypass for TX
			audio_emph_tx_bypass	[0]	PRE-EMPHASIS bypass
REG_9DH	R/W	0x29AD	reserved	[15:4]	Reserved
			audio_tx_path_sel[1:0]	[3:2]	01=select pre-emphasis output 10=reserved

CONFIDENTIAL

					11=select LPF2 output
			reserved	[1:0]	Reserved
REG_B6H	R/W	0x9d08	reserved	[15:13]	Reserved
			~~subau_rx_dcc_bw[2:0]~~	~~[12:10]~~	~~CTC/DCS Rx HPF Bw 000=bypass;001=60Hz;010=30Hz;011=15Hz; 100=8Hz;101=4Hz;110=2Hz;111=1Hz~~
			subau_tx_atten_gain[1:0]	[9:8]	CTC/DCS Tx Atten Gain, -6dB/step
			subau_rx_gain1[1:0]	[7:6]	CTC/DCS Rx Gain1, -6dB/step
			subau_rx_gain2[1:0]	[5:4]	CTC/DCS Rx Gain2, -6dB/step
			subau_rx_gain3[3:0]	[3:0]	CTC/DCS Rx Gain3, -6dB/step

CONFIDENTIAL

CONFIDENTIAL