

RDA5807SP 编程指南 V2.5

CONFIDENTIAL

## 一. 控制接口:

RDA5807SP 提供了 I2C 作为控制接口，接口的控制模式如下描述:

I2C 接口与 I2C-Bus Specification 2.1 兼容,包含 2 个信号:SCLK 和 SDIO。

I2C 接口是由 START, 命令字节, 数据字节, 及每个字节后的 ACK 或 NACK 比特, 和 STOP 组成, 命令字节包括一个 7 比特的 chip 地址 (0010000b) 和一个读写  $r/\bar{w}$  命令比特。ACK (或 NACK) 由接收器发出。

在该接口下, 有两中读写方式, 分别是连续读写方式和带寄存器地址的标准 I2C 方式, 两种方式通过 I2C 的器件地址来区分实现, 读写方式和器件地址详细描述如下:

1、连续读写方式: 在该模式下, 寄存器的地址是不可见的, 即有一个固定的起始寄存器地址 (写操作时为 02H, 读操作时为 0AH), 并有一个内部递增计数器, I2C 器件地址为 0010000B, 加上读写标志, 即 I2C 器件地址为 0x20 (写操作) 和 0x21 (读操作)

(1)、写操作: 写操作默认起始寄存器为 02H, 即所有写操作都是默认从 02H 开始, 即使只写如 03H 或者 05H, 都必须从 02H 写起, MCU 写入寄存器的顺序如下: 02H 的高字节, 02H 的低字节, 03H 的高字节, ... .., 直到结束。

芯片在 MCU 写入每个字节后都会返回一个 ACK。MCU 会给出 STOP 来结束操作。

(2)、读操作：读操作默认起始寄存器为 0AH，即所有写操作都是默认从 0AH 开始。在对芯片进行读操作时，MCU 给出命令字节后，RDA5807SP 会送出数据字节，顺序如下：0AH 高字节，0AH 低字节，0BH 高字节，……，直到芯片接收到从 MCU 发出的 NACK，MCU 送出 STOP，读操作结束。除了最后一个字节，MCU 在读到每个字节后都要给出 ACK，在读到最后一个字节后，MCU 给出 NACK，使芯片把总线交给 MCU，然后 MCU 发出 STOP，结束整个操作。

(3)、连续读写方式时序图：

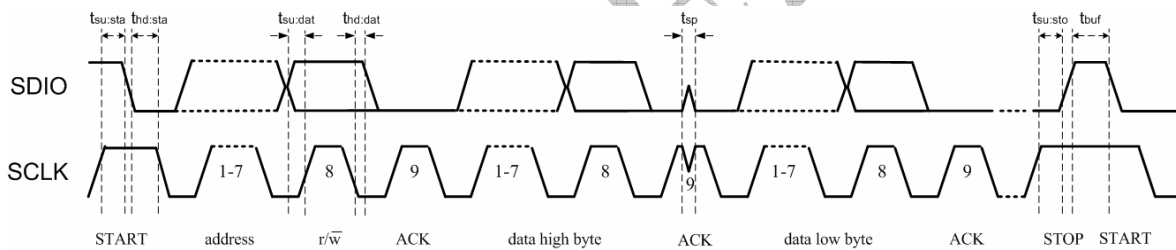


图 1 写数据时序

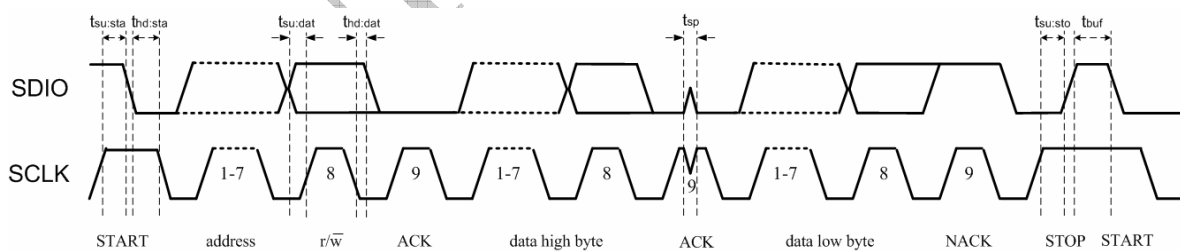


图 2 读数据时序

### I2C Timing Characteristics

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
SCLK Frequency	f <sub>scl</sub>		0	-	400	KHz
SCLK High Time	t <sub>high</sub>		0.6	-	-	us
SCLK Low Time	t <sub>low</sub>		1.3	-	-	us
Setup Time for START Condition	t <sub>su:sta</sub>		0.6	-	-	us

Hold Time for START Condition	thd: sta		0.6	-	-	us
Setup Time for STOP condition	tsu: sto		0.6	-	-	us
SDIO Input to SCLK ↑ Setup	tsu: dat		100	-	-	ns
SDIO Input to SCLK ↓ Hold	thd: dat		0	-	900	ns
STOP to START Time	tbuf		1.3	-	-	us
SDIO Output Fall Time	tf: out		20+0.1Cb	-	250	ns
SDIO Input, SCLK Rise/Fall Time	tr: in tf: in		20+0.1Cb	-	300	ns
Input Spike Suppression	tsp		-	-	50	ns
SCLK, SDIO Capacitive Loading	Cb		-	-	50	pF

2、标准 I2C 读写方式：该模式是与标准 I2C 读写方式一致，即带寄存器地址的方式，I2C 器件地址为 0010001B，加上读写标志位，即为 0x22(写操作)和 0x23(读操作)，读写方式的格式如下：

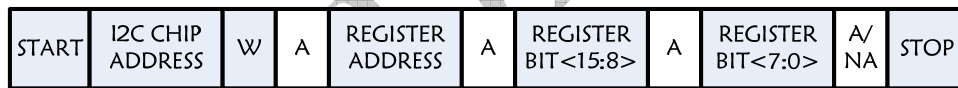


图 3 标准 i2c 写格式



图 4 标准 i2c 读格式

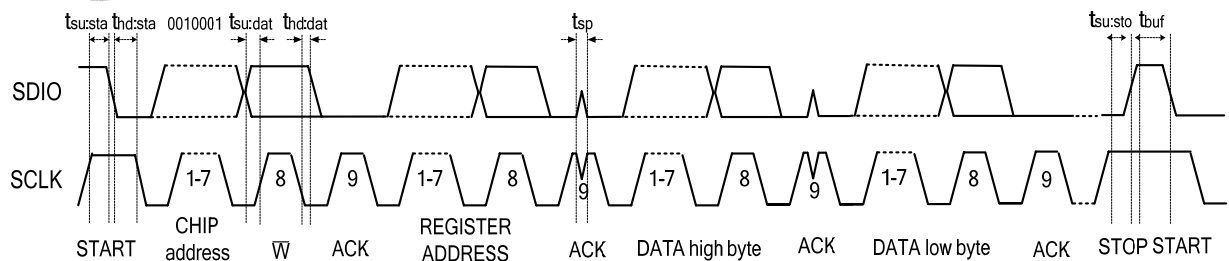
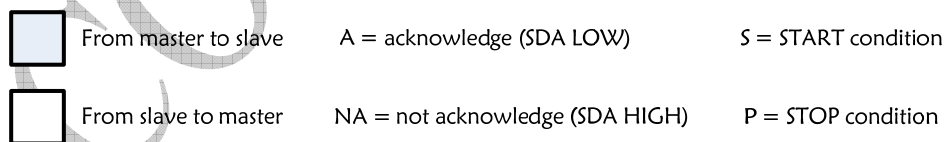


图 5 标准 i2c 写时序图

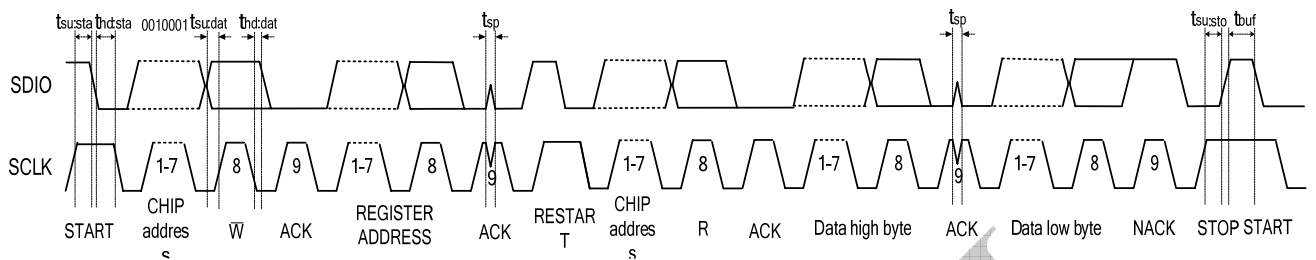


图 6 标准 i2c 读时序图

3、另外，RDA5807SP 还有另外一组未使用的 I2C 器件地址 1100000B，加上读写标志，即为 0xC0 和 0xC1，即 RDA5807SP 总共有三组 I2C 器件地址，分别是 0x20 和 0x21，0x22 和 0x23，0xC0 和 0xC1，在硬件设计时，如果 RDA5807SP 和其他 I2C 器件共用 I2C 总线时，请注意避免 I2C 器件地址冲突的问题。

## . 二、状态转换

RDA5807SP 中有 5 种状态: 复位初始化 (Reset&Initial), 设置频点 (Tune), 搜台 (Seek), 工作 (Working), 休眠 (Sleep)。

在芯片上电和复位后, 软件通过编写 ENABLE (02H, bit 0) 寄存器, 将其置为 1, 即可使 RDA5807SP 进入上电状态。软件通过编程相应寄存器, 即可使 RDA5807SP 进入 Tune 或 Seek 状态, 这些操作之后, RDA5807SP 进入正常工作状态 (Working)。软件通过将 ENABLE 置为 0, 可使 RDA5807SP 进入睡眠状态, 此时所有寄存器值保持不变 (与未睡眠之前相同)。在睡眠状态时, 软件可通过编写 ENABLE 为 1, 即可将 RDA5807SP 回到正常工作 (Working) 状态。

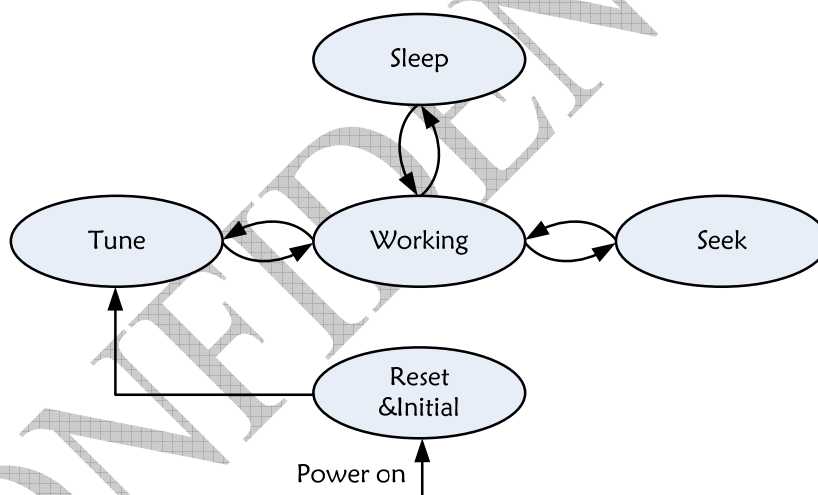


图 9 RDA5807SP 状态转移图

### 三、复位及初始化 (Reset&Initial)

上电过程中，RDA5807SP 需要正确的 Reset 和初始化过程来进行上电。

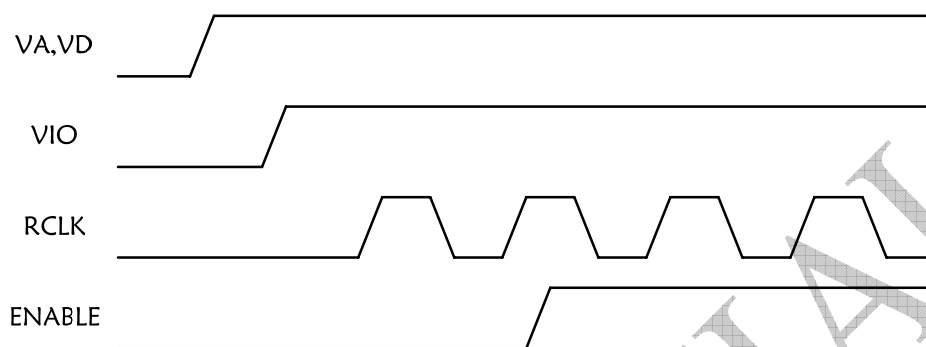


图 10 Reset&Initial 时序图

RDA FM 系列芯片的初始操作顺序：复位 → 上电 → 写初始化数据。

#### (一)、复位及上电操作

- 1、复位操作通过写 0x0002 到 02H 来实现；
- 2、当使用 RDA5807SP 带晶振模块，或者单独给 RDA5807SP 供 32.768KHz 晶振时，需要执行上电操作，这一操作通过写 0xC001 到 02H 来实现。

#### 3、编程伪程序：

Supply VA and VD.

Supply VIO

Provide 32.768KHz crystal clock. (optional, if use TCX0)

(or 12MHz/24MHz/13MHz/26MHz/19.2MHz/38.4MHz clk)

Wait 1ms

Mov 0x0002, 02H //write Soft\_Reset=1

---

```
Delay 50ms          //optional, for wait RCLK stable if use DCX0
Mov 0xC001, 02H    //write enable=1
Delay 600ms
```

## (二)、芯片 ID 读取

RDA5807SP 芯片 ID 可以通过读取 0CH 和 0DH 获得的, 0CH 和 0DH 的值均为 0x5804, 下面将介绍芯片 ID 的读取方法:

当进行复位及上电操作后, 可读取芯片的 ID:

- 1、连续读写方式读 ID 操作需连续读取 8 byte 数据, 即读取 0AH、0BH、0CH、0DH;
- 2、采用标准 I2C 读写方式读 ID 则直接读 0CH 和 0DH。

## (三)、初始化操作

初始化操作通过 I2C 或 3 线 SPI 接口把初始化参数写到芯片内部来实现, 而编程及调试时, 需要注意的是 02H 的初始值:

- 1、当使用 RDA5807SP 带晶振模块, 或者单独给 RDA5807SP 供 32.768KHz 晶振时, 02H 的初始值为 0xC001;
- 2、当 RDA5807SP 与其他器件共用晶振时, 需要按照晶振的频率来设置 02H 的初始值, 具体为设置 02H 的 CLK\_MODE (Bit [6:4]), 同时需要把 02H 的 Bit 10



置 1。

例：

当共用 12MHz 晶振时，02H 的初始值为 0xC411；

当共用 24MHz 晶振时，02H 的初始值为： 0xC451；

3、初始化参数列表(如有需要请联系我司当地 FAE 获取最新初始化参数列表)

RDA5807SP 初始化列表如下：

```
0xC001, //0xC411//0xC451 //02H: //请注意按照 1 和 2 中所描述进行设置
#if defined(_FM_STEP_50K_)
0x1b, 0x92,
0x0C, 0x00,
#else //Step 100K
0x1b, 0x90,
0x04, 0x00,
#endif

0x86, 0xad, //05H:
0x80, 0x00,
0x5F, 0x1A, //07H
0x5e, 0xc6,
0x00, 0x00,
0x40, 0x6e, //0AH:
0x2d, 0x80,
0x58, 0x03,
0x58, 0x04,
0x58, 0x04,
0x58, 0x04,
0x00, 0x47, //10H:
0x90, 0x00,
0xF5, 0x87,
```

0x7F, 0x0B, //13H:  
0x00, 0xF1,  
0x42, 0xc0, //15H:  
0x41, 0xe0,  
0x50, 0x6f,  
0x55, 0x92,  
0x00, 0x7d,  
0x10, 0xC0, //1AH  
0x07, 0x80,  
0x41, 0x1d, //1CH,  
0x40, 0x06,  
0x1f, 0x9B,  
0x4c, 0x2b, //1FH.  
0x81, 0x10, //20H:  
0x45, 0xa0, // 21H

#if defined(\_FM\_STEP\_50K\_)

0x55, 0x3F, //22H

#else

0x19, 0x3F, //22H

#endif

0xaf, 0x40,  
0x04, 0x81,  
0x1b, 0x2a, //25H  
0x0D, 0x04,  
0x80, 0x2F,  
0x17, 0x8A,  
0xD3, 0x49,  
0x11, 0x42,  
0xA0, 0xC4, //2BH  
0x3C, 0x3B,  
0x00, 0x00,  
0x58, 0x04,  
0x58, 0x04, //2FH  
0x58, 0x04,  
0x00, 0x74,  
0x3D, 0x00,  
0x03, 0x0C,  
0x2F, 0x68,  
0x38, 0x77, //35H

#### 四、设置频点 (Tune)

软件可以通过配置 03H 寄存器来选择 FM 频道。步进长度 (100KHz, 200KHz, 50KHz 或 12.5KHz) 由 SPACE 来选择, 频道由 CHAN[9:0] 来选择, 频率范围 (76MHz-91MHz, 87MHz-108MHz, 76MHz-108MHz) 由 BAND[1:0] 来选择。

当软件写 03H 寄存器的 TUNE 位为 1 时, RDA5807SP 会自动开始 Tune。在 Tune 结束时 (如果 STCIEN 设为 1, 会产生一个中断信号 INT 由 GPIO2 送出), STC 会被置 1, 软件可以通过读 0AH 和 0BH 寄存器来得到当前频点的状态值 (ST, FM\_TRUE, FM\_READY, RSSI, READCHAN 等)。整个 Tune 过程要持续 50ms。

频点计算方法见寄存器 CHAN 和 READCHAN 的换算公式。

编程伪程序:

```
Mov 0x1A10, 03H    //Set channel number to 97.4MHz, space to 100KHz, band to 87_108MHz
Delay 50ms
*Wait for GPIO2=0  //optional, wait for tune complete, if use interrupt
*Wait for STC=1    //optional, wait for tune complete, if use polling method
Read 0A, 0BH      //read stauts
Stop Tune
```

注意: 对 I2C 接口而言, RDA5807SP 的寄存器设置是由一连串的写操作来完成的, 所以软件要注意写寄存器的顺序。对 3 线接口而言则只需写相应寄存器即可。

## 五、搜台 (Seek)

RDA5807SP 搜台模式编程流程如下图所示。

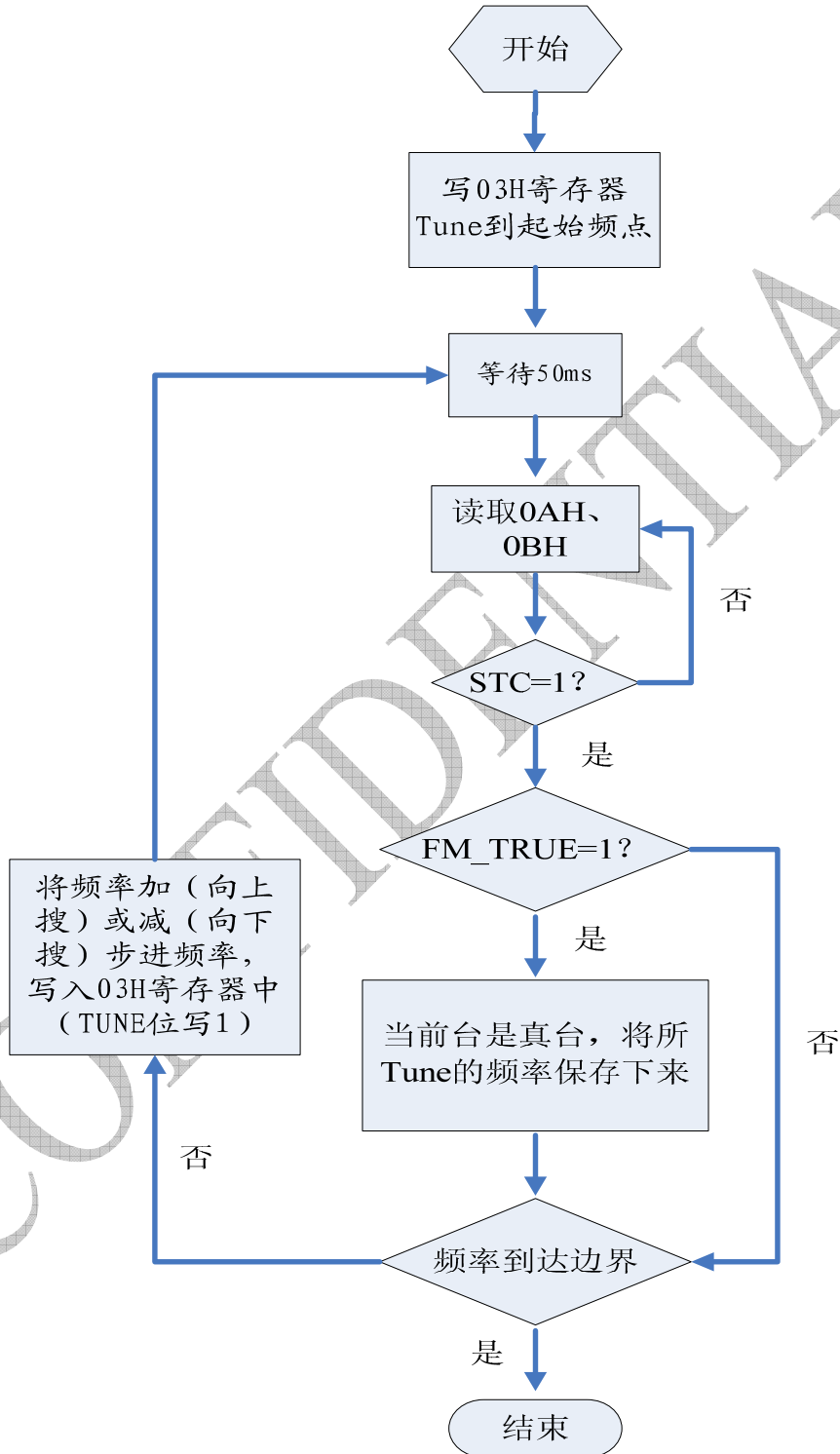


图 11 RDA5807SP 搜台模式编程流程图

编程伪程序:

Step1:

```
CHAN=0x0005;
```

```
VALUE=(CHAN<<6)+0x0010;
```

```
Mov VALUE, 03H //Set channel number to 87.5MHz, space to 100KHz, band to 87_108MHz
```

Step2:

```
Wait 50ms //minus 35ms
```

```
Read 0A, 0BH //read stauts
```

```
If freq beyond band limit, go to Step4. Else go to Step3.
```

Step3:

```
If STC=1 and FM-TRUE=1, memorize READCHAN.
```

```
CHAN=CHAN+1;
```

```
VALUE=(CHAN<<6)+0x0010;
```

```
Mov VALUE, 03H
```

```
Go to Step2.
```

Step4:

```
Stop Seek.
```

注意: 对 I2C 接口而言, RDA5807SP 的寄存器设置是由一连串的写操作来完成的, 所以软件要注意写寄存器的顺序。对 3 线接口而言则只需写相应寄存器即可。

## 六、休眠 (Sleep)

在空闲时，软件可以通过编程 RDA5807SP 中 ENABLE (置 0) 使 RDA5807SP 进入睡眠模式，以便减小功耗。在睡眠模式，RDA5807SP 模拟和数字模块电源都被关掉，但各寄存器值保持不变，SPI 和 I2C 接口依然可以工作。

软件可以通过编程 ENABLE (置 1) 使 RDA5807SP 进入工作模式。进入工作模式后，软件需要重新设置所需要的频点，即重新进行一次 Tune 操作。

编程伪程序：

Enter Sleep Mode:

```
Mov 0x0000, 02H //clear ENABLE bit low to bring RDA5807SP into sleep mode
```

Exit Sleep Mode:

```
Mov 0xC001, 02h //set ENABLE bit high to bring RDA5807SP into working mode
```

```
//0xC411//0xC451 //02H: //请注意按照初始化操作章节中 1 和 2 中所描述进行设置
```

```
Wait 0.5s //optional, wait RCLK stable, if in DCX0 mode
```

```
Mov 0x0150, 03h //Set channel number to 87.5MHz, space to 100KHz, band to 87-108MHz
```

```
*Wait for GPIO2=0 //optional, wait for tune complete, if use interrupt
```

```
*Wait for STC=1 //optional, wait for tune complete, if use polling method
```

```
Read 0A, 0BH //read stauts
```

```
Stop Tune
```

注意：对 I2C 接口而言，RDA5807SP 的寄存器设置是由一连串的写操作来完成的，所以软件要注意写寄存器的顺序。对 3 线接口而言则只需写相应寄存器即可。

## 七. 寄存器说明

REG	BITS	NAME	FUNCTION	DEFAULT
02H	15	DHIZ	Audio Output High-Z Disable. <i>0 = High impedance; 1 = Normal operation</i>	0
	14	DMUTE	Mute Disable. <i>0 = Mute; 1 = Normal operation</i>	0
	13	MONO	Mono Select. <i>0 = Stereo; 1 = Force mono</i>	0
	12	BASS	Bass Boost. <i>0 = Disabled; 1 = Bass boost enabled</i>	0
	11	RESEVED	Must be 0	0
	10	CLK32_INPUT_ENB	0 = enable 1 = disenable (If share crystal,must be 1)	0
	8	RESEVED	Must be 0	0
	7	RESEVED	Must be 0	0
	6:4	CLK_MODE[2:0]	000=32.768kHz 001=12Mhz 101=24Mhz 010=13Mhz 110=26Mhz 011=19.2Mhz 111=38.4Mhz	000
	1	SOFT_RESET	Soft reset. If 0, not reset; If 1, reset.	0
	0	ENABLE	Power Up Enable. <i>0 = Disabled; 1 = Enabled</i>	0
03H	15:6	CHAN[9:0]	Channel Select. BAND = 0 <i>Frequency = Channel Spacing (kHz) x CHAN + 87MHz</i> BAND = 1 <i>Frequency = Channel Spacing (kHz) x CHAN + 76.0MHz</i> CHAN is updated after a seek operation.	0x00
	5	RESERVED	Must be 0	0
	4	TUNE	Tune 0 = Disable	0

REG	BITS	NAME	FUNCTION	DEFAULT
			1 = Enable The tune operation begins when the TUNE bit is set high. The STC bit is set high when the tune operation completes. The tune bit is reset to low automatically when the tune operation completes..	
	3:2	BAND[1:0]	Band Select. 00 = 87~108 MHz (US/Europe) 01 = 76~91 MHz (Japan) 10 = 76~108 MHz (Japan wide)	00
	1:0	SPACE[1:0]	Channel Spacing. 00 = 100 kHz 01 = 200 kHz 10 = 50kHz 11 = 12.5KHz	00
04H	14	STCIEN	Seek/Tune Complete Interrupt Enable. 0 = Disable Interrupt 1 = Enable Interrupt Setting STCIEN = 1 will generate a low pulse on GPIO2 when the interrupt occurs.	0
	12	RESERVED	Must be 0	0
	11	DE	De-emphasis. 0 = 75 $\mu$ s; 1 = 50 $\mu$ s	0
	9:7	RESERVED	Must be 000	000
	6	I2S_ENABLED	I2S bus enable If 0, disabled; If 1, enabled.	0
	5:4	GPIO3[1:0]	General Purpose I/O 3. 00 = High impedance 01 = Mono/Stereo indicator (ST) 10 = Low 11 = High	00
	3:2	GPIO2[1:0]	General Purpose I/O 2. 00 = High impedance 01 = Interrupt (INT) 10 = Low 11 = High	00
	1:0	GPIO1[1:0]	General Purpose I/O 1. 00 = High impedance 01 = Reserved 10 = Low 11 = High	00
05H	15	INT_MODE	If 0, generate 5ms interrupt;	1



REG	BITS	NAME	FUNCTION	DEFAULT
			If 1, interrupt last until read reg0CH action occurs.	
	14:8	SEEKTH[6:0]	Seek Threshold. RSSI scale is logarithmic. 0000000 = min RSSI	0001000
	7:6	LNA_PORT_SEL[1:0]	LNA input port selection bit: 00: no input 01: LNaN 10: LNaP 11: dual port input	10
	5:4	LNA_ICSEL_BIT[1:0]	Lna working current bit: 00=1.8mA 01=2.1mA 10=2.5mA 11=3.0mA	10
	3:0	VOLUME[3:0]	DAC Gain Control Bits (Volume). 0000=min; 1111=max Volume scale is logarithmic	1000
OAH	15	RESERVED		0
	14	STC	Seek/Tune Complete. 0 = Not complete 1 = Complete The seek/tune complete flag is set when the seek or tune operation completes.	0
	13	SF	Seek Fail. 0 = Seek successful; 1 = Seek failure The seek fail flag is set when the seek operation fails to find a channel with an RSSI level greater than SEEKTH[5:0].	0
	12	RESERVED		0
	11	RESERVED		
	10	ST	Stereo Indicator. 0 = Mono; 1 = Stereo Stereo indication is available on GPIO3 by setting GPIO1[1:0] = 01.	1
	9:0	READCHAN[9:0]	Read Channel. BAND = 0 Frequency = Channel Spacing (kHz) x READCHAN[7:0]+ 87 MHz BAND = 1 Frequency = Channel Spacing (kHz) x READCHAN[7:0]+ 76.0 MHz READCHAN[7:0] is updated after a tune or seek operation.	8' h00
OBH	15:9	RSSI[6:0]	RSSI. 000000 = min 111111 = max	0

REG	BITS	NAME	FUNCTION	DEFAULT
			RSSI scale is logarithmic.	
	8	FM_TRUE	1 = the current channel is a station 0 = the current channel is not a station	0
	7	FM_READY	Used for soft seek 1 = ready 0 = not ready	0

CONFIDENTIAL

The information provided here is believed to be reliable; RDA Microelectronics assumes no liability for inaccuracies and omissions. RDA Microelectronics assumes no liability for the use of this information and all such information should entirely be at the user's own risk. Specifications described and contained here are subjected to change without notice for the purpose of improving the design and performance. All of the information described herein shall only be used for sole purpose of development work of RDA5807SP, no right or license is implied or granted except for the above mentioned purpose. RDA Microelectronics does not authorize or warrant any RDA products for use in the life support devices or systems.

Copyright©2006 RDA Microelectronics Inc. All rights reserved



For technical questions and additional information about RDA Microelectronics Inc.:

Website: [www.rdamicro.com](http://www.rdamicro.com)

Mailbox: [info@rdamicro.com](mailto:info@rdamicro.com)

RDA Microelectronics (Shanghai), Inc.

Tel: +86-21-50271108

Fax: +86-21-50271099

RDA Microelectronics (Beijing), Inc.

Tel: +86-10-63635360

Fax: +86-10-82612663