RDA5807N 系列编程指南 V3.5

一. 说明

RDA5807N 系列包括 RDA5807NP、RDA5807FP、RDA5807NN、RDA5807M，下文简称为 RDA5807N。

二. 控制接口:

RDA5807N 提供了 I2C 作为控制接口，控制模式如下描述:

(一)、I2C 控制接口:

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/$\overline{w}$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，RDA5807N 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)、连续读写方式时续图：

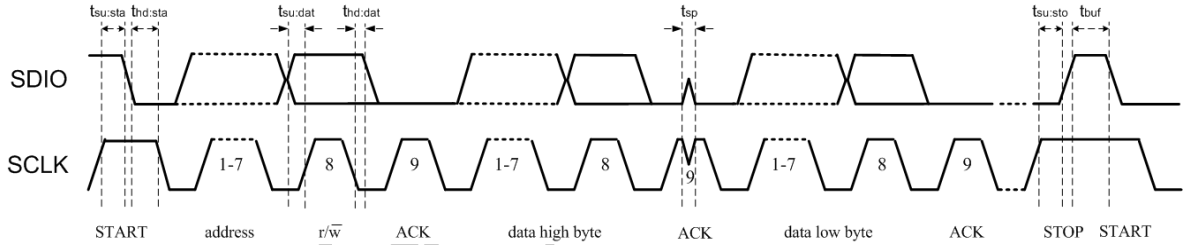

图 1 写数据时序

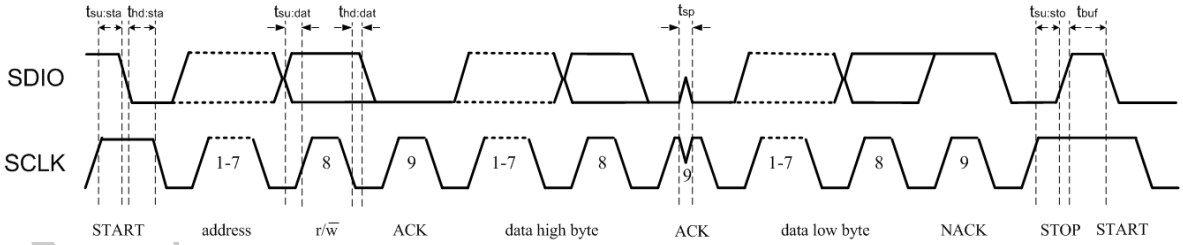

图 2 读数据时序

I2C Timing Characteristics

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
SCLK Frequency	fscl		0	-	400	KHz
SCLK High Time	thigh		0.6	-	-	us
SCLK Low Time	tlow		1.3	-	-	us
Setup Time for START	tsu:sta		0.6	-	-	us

Condition						
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：

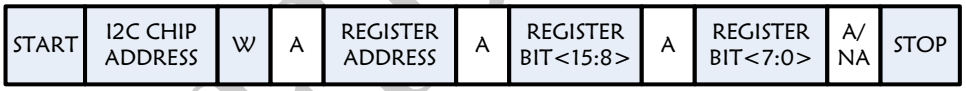

图 3 标准 i2c 写格式

图 4 标准 i2c 读格式

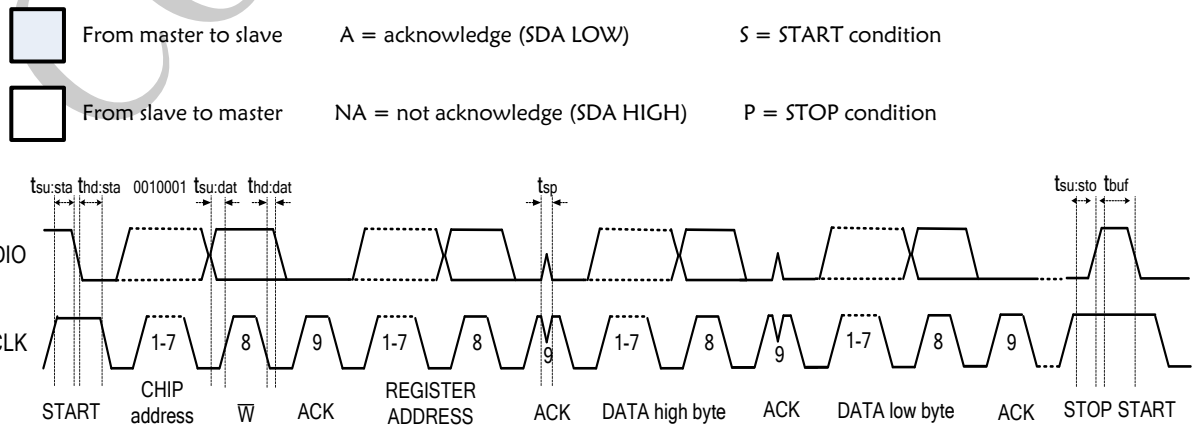

图 5 标准 i2c 写时序图

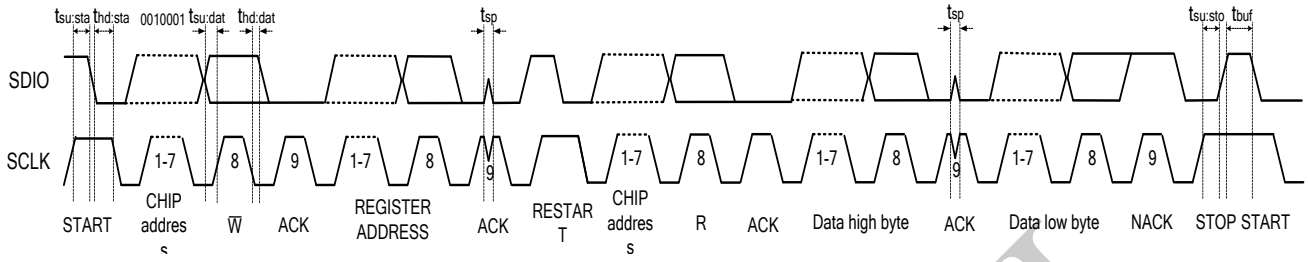

图 6 标准 i2c 读时序图

3、另外，RDA5807N 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 RDA5807N 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 RDA5807N 与其他 I2C 器件共用 I2C 总线时，请注意避免 I2C 器件地址冲突的问题。

三、状态转换

RDA5807N 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 RDA5807N 进入上电状态。软件通过编程相应寄存器，即可使 RDA5807N 进入 Tune 或 Seek 状态，这些操作之后，RDA5807N 进入正常工作状态（Working）。软件通过将 ENABLE 置为 0，可使 RDA5807N 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 RDA5807N 回到正常工作（Working）状态。

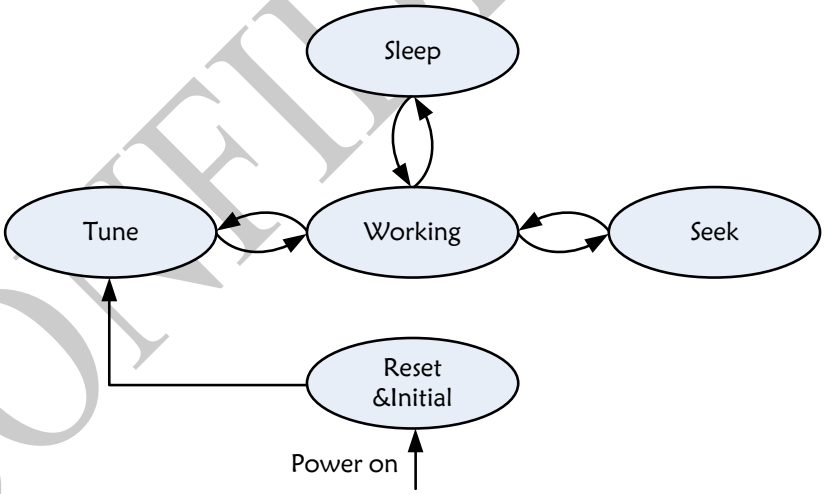

图 9 RDA5807N 状态转移图

四、复位及初始化（Reset&Initial）

RDA FM 系列芯片的初始操作顺序：复位 → 读取芯片 ID → 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 RDA5807N 带晶振模块，或者单独给 RDA5807N 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 TCXO)
(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 DCXO
Mov 0xC001, 02H  //write enable=1
Delay 600ms
```

(二)、芯片 ID 读取方法

当进行芯片复位操作后，可读取芯片的 ID，RDA5807N 系列 ID 为 0x5808，可以通过读取 0EH 来获得，以下是 ID 读取的方法:

1、连续读写方式读 ID 操作需连续读取 10 byte 数据，即读取 0AH、0BH、0CH、0DH，0EH;

2、采用标准 I2C 读写方式读 ID 则直接读 0EH。

读出来的值为：0CH：0x5803；0DH：0x5804；0EH：0x5808。

(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 RDA5807N 带晶振模块，或者单独给 RDA5807N 供 32.768KHz 晶振时，02H 的初始值为 0xC001;

2、当 RDA5807N 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]),并且同时把 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、请联系我司代理商 FAE 获取最新初始化参数列表。

五、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，RDA5807N 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，RDA5807N 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、搜台（Seek）

RDA5807N 搜台模式编程流程如下图所示。

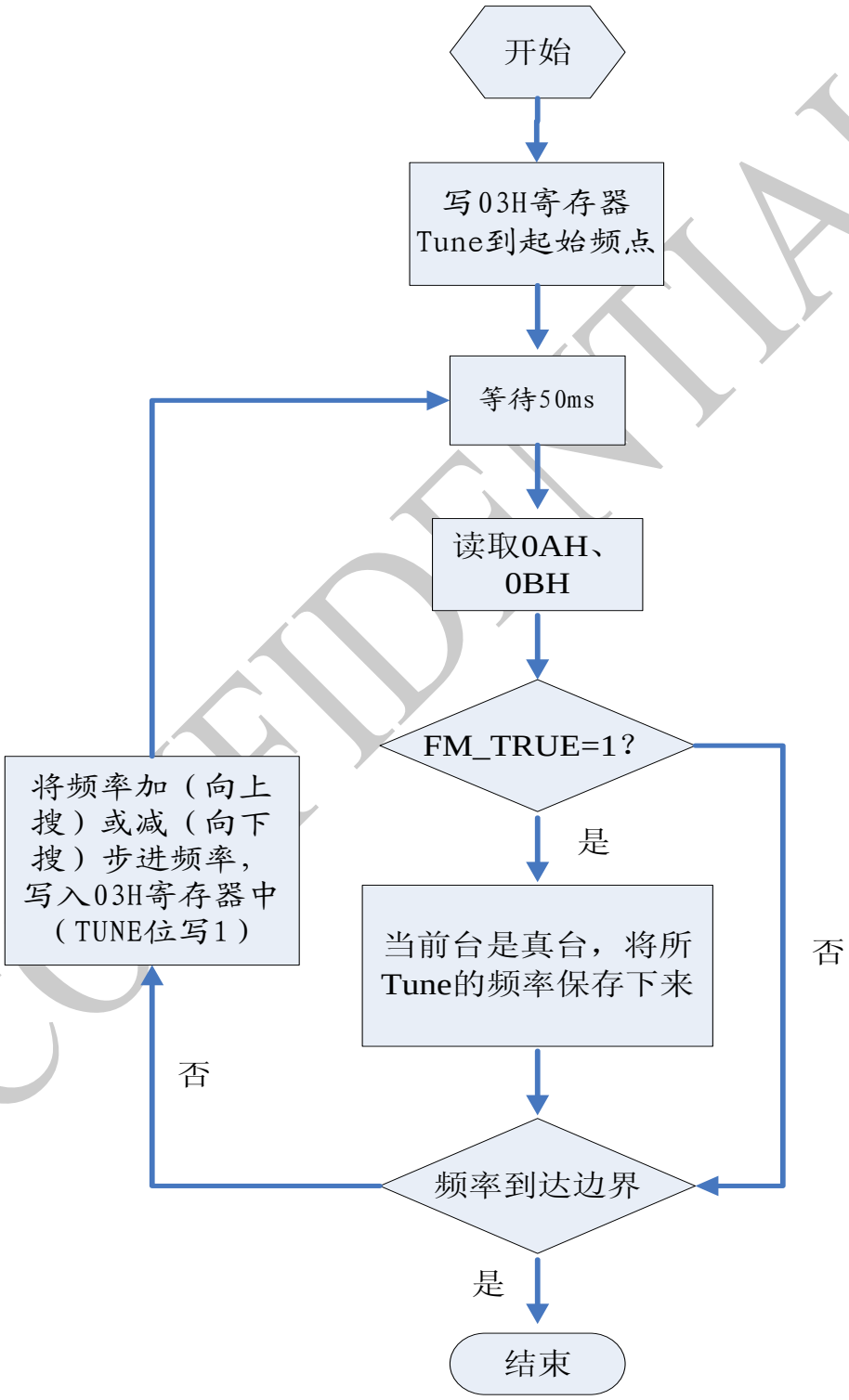

图 10 RDA5807N 搜台模式编程流程图

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, RDA5807N 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、休眠（Sleep）

在空闲时，软件可以通过编程 RDA5807N 中 ENABLE（置 0）使 RDA5807N 进入睡眠模式，以便减小功耗。在睡眠模式，RDA5807N 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 RDA5807N 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 RDA5807N into sleep mode
Exit Sleep Mode:
    Mov 0xC001, 02h       //set ENABLE bit high to bring RDA5807N 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 DCXO 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，RDA5807N 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 线接口或者使用带地址的 I2C 通讯方式而言则只需写相应寄存器即可。

八. RBDS/RDS

RDA5807N 中设计了 RBDS/RDS 处理器，该处理器包括 BLOCK 的同步，误码检测及纠错功能。软件可以通过将寄存器 02H 中的 RDS_EN 置 1 来开启 RBDS/RDS 接收功能。当寄存器 0AH 中的 RDSS（0AH,RBDS/RDS 同步信息）为 1 时，表明 RBDS/RDS 的 GROUP 已经同步，为 0 时，则表示同步丢失。当同步完成时，每当一个 GROUP 数据被接收后，RDSR(0AH)比特位会置 1，直到软件对寄存器 0AH 进行读操作，RDSR 才会被置 0。当每一个 GROUP 数据被接收时，同样可以送出中断信号，设置 RDSIEN(04H)为 1，并且 GPIO2[1:0]为 01 即可在 GPIO2 引脚送出 RDSR 的中断信号。RDA5807N 同样会通过寄存器 BLERA, BLERB, BLERC, BLERD 送出每个 BLOCK 的误码信息。其含义间寄存器说明。

RBDS/RDS 模式可以通过配置寄存器 04H 的 RBDS 来选择,RBDS 为置 1 则进入 RBDS 模式，置 0 则进入 RDS 模式。在 RBDS 接收模式下,接收到的 BLOCK 除了 RDS 的 BLOCK(A,B,C/c',D)外，还会有 BLOCK E（4 个 BLOCK E 组成 1 个 GROUP）。软件可以通过读寄存器 ABCD_E 来区分 BLOCK ID 信息。当 ABCD_E 为 0 时，则寄存器 0CH,0DH,0EH 和 0FH 当前的数据分别对应 A,B,C/c',D4 个 BLOCK。当 ABCD_E 为 1 时，则寄存器 0CH,0DH,0EH 和 0FH 当前的数据都为 BLOCK E。
软件接收 RDS 数据流程如下图:

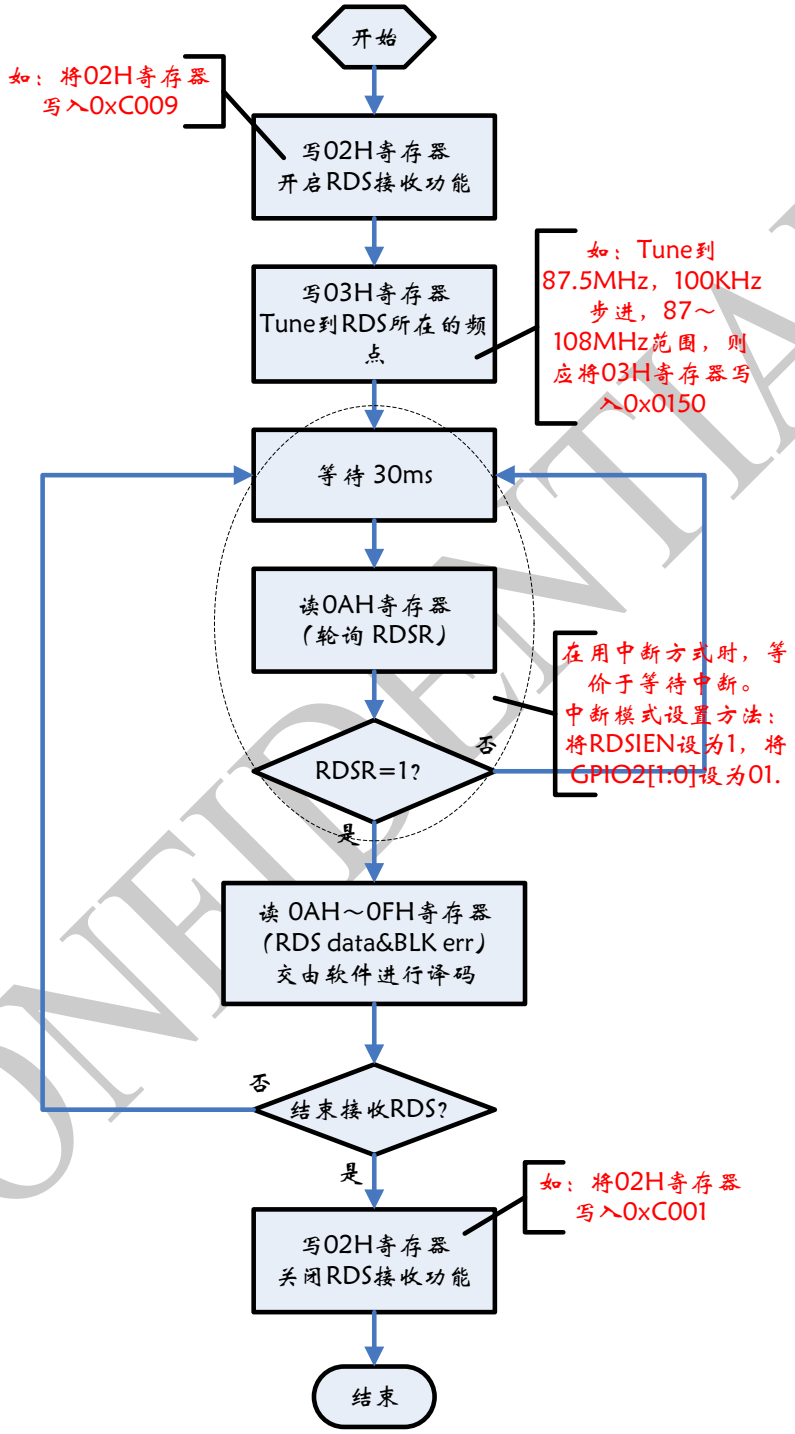

图 11 软件接收 RDS 数据流程图

编程伪程序:

Step1：

Mov 0xC009, 02H //set RBDS/RDS enable.

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

Step2：

*Wait for GPIO2=0 //optional, wait for RBDS/RDS group ready, if use interrupt

*Wait for RDSR=1 //optional, wait for RBDS/RDS group ready, if use polling method

Read 0A~0FH //read RDS data

Software decode RDS information

If end RDS receive, go to Step3; else go to Step 2.

Step3：

Mov 0xC001, 02H //clear RBDS/RDS enable.

Stop RDS receive

九. 寄存器说明

REG	BITS	NAME	FUNCTION	DEFAULT
00H	15:8	CHIPID[7:0]	Chip ID.	0x58
02H	15	DHIZ	Audio Output High-Z Disable. *0 = High impedance; 1 = Normal operation*	0
	14	DMUTE	Mute Disable. *0 = Mute; 1 = Normal operation*	0
	13	MONO	Mono Select. *0 = Stereo; 1 = Force mono*	0
	12	BASS	Bass Boost. *0 = Disabled; 1 = Bass boost enabled*	0
	11	RCLK NON-CALIBRATE MODE	0=RCLK clock is always supply 1=RCLK clock is not always supply when FM work (when 1, RDA5807N can't directly support -20 ℃ ~70 ℃ temperature. Only suppory ±20℃ temperature swing from tune point)	0
	10	RCLK DIRECT INPUT MODE	1=RCLK clock use the directly input mode (If share crystal,must be 1)	0
	9	SEEKUP	Seek Up. *0 = Seek down; 1 = Seek up*	0
	8	SEEK	Seek. *0 = Disable stop seek; 1 = Enable* *Seek begins in the direction specified by* SEEKUP *and ends when a channel is found, or the entire band has been searched.* *The* SEEK *bit is set low and the* STC *bit is set high when the seek operation completes.*	0
	7	SKMODE	Seek Mode 0 = wrap at the upper or lower band limit and continue seeking 1 = stop seeking at the upper or lower band limit	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
	3	RDS_EN	RDS/RBDS enable If 1, rds/rbds enable	0
	2	NEW_METHOD	New Demodulate Method Enable, can improve the receive sensitivity about 1dB.	0

REG	BITS	NAME	FUNCTION	DEFAULT
	1	**SOFT_RESET**	**Soft reset.** **If 0, not reset;** **If 1, reset.**	0
	0	**ENABLE**	**Power Up Enable.** ***0 = Disabled; 1 = Enabled***	0
03H	**15:6**	**CHAN[9:0]**	**Channel Select.** **BAND =** ***0*** ***Frequency =*** ***Channel Spacing (kHz) x* CHAN+ *87.0 MHz*** **BAND =** ***1or 2*** ***Frequency =*** ***Channel Spacing (kHz) x* CHAN + *76.0 MHz*** **BAND =** ***3*** ***Frequency =*** ***Channel Spacing (kHz) x* CHAN + *65.0 MHz*** **CHAN** ***is updated after a seek operation.***	0x00
	5	**DIRECT MODE**	**Directly Control Mode, Only used when test.**	0
	4	**TUNE**	**Tune** **0 = Disable** **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..**	0
	3:2	**BAND[1:0]**	**Band Select.** **00 = 87–108 MHz (US/Europe)** **01 = 76–91 MHz (Japan)** **10 = 76–108 MHz (world wide)** **11**[1] **= 65 –76 MHz （East Europe） or 50-65MHz**	00
	1:0	**SPACE[1:0]**	**Channel Spacing.** **00 = 100 kHz** **01 = 200 kHz** **10 = 50kHz** **11 = 25KHz**	00
04H	**15**	**RDSIEN**	**RDS ready Interrupt Enable.** **0 = Disable Interrupt** **1 = Enable Interrupt** **Setting STCIEN = 1 will generate a low pulse on GPIO2 when the interrupt occurs.**	0
	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**

[1] If 0x07h_bit[9](band)=1, 65-76MHz; =0, 50-76MHz

REG	BITS	NAME	FUNCTION	DEFAULT
	13	RBDS	1 = RBDS mode enable 0 = RDS mode only	0
	12	RDS_FIFO_EN	1 = RDS fifo mode enable.	0
	11	DE	De-emphasis. *0 = 75 μs; 1 = 50 μs*	0
	10	RSVD	Reserved Must be 1	1
	9	SOFTMUTE_EN	If 1, softmute enable	0
	8	AFCD	AFC disable. If 0, afc work; If 1, afc disabled.	0
	7	RSVD	Reserved	
	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; If 1, interrupt last until read reg0CH action occurs.	1
	14:13	SEEK_MODE[1:0]	RDA5807N Seek Mode Select Default value is 00; When = 10, will add the 5802E seek mode.	00
	12	RSVD	Reserved	0
	11:8	SEEKTH[3:0][2]	Seek SNR threshold value: Noise_th(dB) = 79 – seek_th	1000
	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	RSVD	Resvered	00
	3:0	VOLUME[3:0]	DAC Gain Control Bits (Volume).	1111

[2] The default noise threshold is 71dB

REG	BITS	NAME	FUNCTION	DEFAULT
			0000=min; 1111=max Volume scale is logarithmic When 0000, output mute and output impedance is very large	
06H	15	RSVD	reserved	0
	14:13	OPEN_MODE[1:0]	Open reserved register mode. 11=open behind registers writing function others: only open behind registers reading function	00
	12	I2S_MODE[3]	If 0, master mode; If 1, slave mode.	0
	11	SW_LR[3]	Ws relation to l/r channel. If 0, ws=0 ->r, ws=1 ->l; If 1, ws=0 ->l, ws=1 ->r.	10
	10	SCLK_I_EDGE[3]	When I2S enable If 0, use normal sclk internally; If 1, inverte sclk internally.	0
	9	DATA_SIGNED[3]	If 0, I2S output unsigned 16-bit audio data. If 1, I2S output signed 16-bit audio data.	0
	8	WS_I_EDGE[3]	If 0, use normal ws internally; If 1, inverte ws internally.	0
	7:4	I2S_SW_CNT[4:0][3] Only valid in master mode	4'b1000: WS_STEP_48; 4'b0111: WS_STEP=44.1kbps; 4'b0110: WS_STEP=32kbps; 4'b0101: WS_STEP=24kbps; 4'b0100: WS_STEP=22.05kbps; 4'b0011: WS_STEP=16kbps; 4'b0010: WS_STEP=12kbps; 4'b0001: WS_STEP=11.025kbps; 4'b0000: WS_STEP=8kbps;	0000
	3	SW_O_EDGE[3]	If 1, invert ws output when as master.	0
	2	SCLK_O_EDGE[3]	If 1, invert sclk output when as master.	0
	1	L_DELY[3]	If 1, L channel data delay 1T.	0
	0	R_DELY[3]	If 1, R channel data delay 1T.	0
07H	15	RSVD	Reserved	0
	14:10	TH_SOFRBLEND[5:0]	Threshold for noise soft blend setting, unit 2dB	10000
	9	65M_50M MODE	Valid when band[1:0] = 2'b11 (0x03H_bit<3:2>) 1 = 65~76 MHz; 0 = 50~76 MHz.	1
	8	RSVD	Reserved	0
	7:2	SEEK_TH_OLD[4]	Seek threshold for old seek mode, Valid when Seek_Mode=10	000000
	1	SOFTBLEND_EN	If 1, Softblend enable	1
	0	FREQ_MODE	If 1, then freq setting changed. Freq = 76000(or 87000) kHz + freq_direct (08H) kHz.	0
0AH	15	RDSR	RDS ready	0

3 This function is open when I2S_Enabled=1.
4 0x05H_bit[14:13], SEEK_MODE register. Default value is 00; When = 01, will add the 5802E seek mode.

REG	BITS	NAME	FUNCTION	DEFAULT
			0 = No RDS/RBDS group ready(default) 1 = New RDS/RBDS group ready	
	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	RDSS	RDS Synchronization 0 = RDS decoder not synchronized(default) 1 = RDS decoder synchronized Available only in RDS Verbose mode	0
	11	BLK_E	When RDS enable: 1 = Block E has been found 0 = no Block E has been found	0
	10	ST	Stereo Indicator. *0 = Mono; 1 = Stereo* *Stereo indication is available on* GPIO3 *by setting* GPIO3[1:0] *=01.*	1
	9:0	READCHAN[9:0]	Read Channel. BAND = 0 Frequency = Channel Spacing (kHz) x READCHAN[9:0]+ 87.0 MHz BAND = 1 or 2 Frequency = Channel Spacing (kHz) x READCHAN[9:0]+ 76.0 MHz BAND = 3 Frequency = Channel Spacing (kHz) x READCHAN[9:0]+ 65.0 MHz READCHAN[9:0] is updated after a tune or seek operation.	8'h00
0BH	15:9	RSSI[6:0]	RSSI. 000000 = min 111111 = max RSSI scale is logarithmic.	0
	8	FM TRUE	1 = the current channel is a station 0 = the current channel is not a station	0
	7	FM_READY	1=ready 0=not ready	0
	6:5	RSVD	Reserved	00
	4	ABCD_E	1= the block id of register 0cH,0dH,0eH,0fH is E 0= the block id of register 0cH, 0dH, 0eH,0fH is A, B, C, D	0
	3:2	BLERA[1:0]	Block Errors Level of RDS_DATA_0, and is always read as Errors Level of RDS BLOCK A (in RDS mode) or BLOCK E (in	00

REG	BITS	NAME	FUNCTION	DEFAULT
			RBDS mode when ABCD_E flag is 1) **00= 0 errors requiring correction** **01= 1~2 errors requiring correction** **10= 3~5 errors requiring correction** **11= 6+ errors or error in checkword, correction not possible.** **Available only in RDS Verbose mode**	
	1:0	**BLERB[1:0]**	**Block Errors Level of RDS_DATA_1, and is always read as Errors Level of RDS BLOCK B (in RDS mode) or E (in RBDS mode when ABCD_E flag is 1).** **00= 0 errors requiring correction** **01= 1~2 errors requiring correction** **10= 3~5 errors requiring correction** **11= 6+ errors or error in checkword, correction not possible.** **Available only in RDS Verbose mode**	00
0CH	**15:0**	**RDSA[15:0]**	**BLOCK A (in RDS mode) or BLOCK E (in RBDS mode when ABCD_E flag is 1)**	16'h5803
0DH	**15:0**	**RDSB[15:0]**	**BLOCK B (in RDS mode) or BLOCK E (in RBDS mode when ABCD_E flag is 1)**	16'h5804
0EH	**15:0**	**RDSC[15:0]**	**BLOCK C (in RDS mode) or BLOCK E (in RBDS mode when ABCD_E flag is 1)**	16'h5808
0FH	**15:0**	**RDSD[15:0]**	**BLOCK D (in RDS mode) or BLOCK E (in RBDS mode when ABCD_E flag is 1)**	16'h5804
10H	**15:14**	**BLERC[1:0]**	**BLK Errors Level of RDS_DATA_2, and is always read as Errors Level of RDS BLOCK C/c'(in RDS mode) or BLOCK E(in RBDS mdoe when ABCD_E flag is 1).** **00 = 0 errors requiring correction** **01 = 1~2 errors requiring correction** **10 = 3~5 errors requiring correction** **11 = 6+ errors or error in checkword, correctionnot possible** **Available only in RDS Verbose mode**	
	13:12	**BLERD[1:0]**	**BLK Errors Level of RDS_DATA_3, and is always read as Errors Level of RDS BLOCK D(in RDS mode) or BLOCK E(in RBDS mdoe when ABCD_E flag is 1).** **00 = 0 errors requiring correction** **01 = 1~2 errors requiring correction** **10 = 3~5 errors requiring correction** **11 = 6+ errors or error in checkword, correctionnot possible** **Available only in RDS Verbose mode**	

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

Website: www.rdamicro.com
Mailbox: 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