

HS6760 应用手册

Version 1.3

WangTong

7/8/16

更改记录

Version/Date	Author	Modification
V1.0:06/07/2016	王同	创建
V1.1:06/23/2016	王同	改正一些性能指标参数 增加不同频段的天线匹配网络
V1.2:06/23/2016	王同	改正 I2C 时序图 初始化代码中加入了对 A1 版几个寄存器的配置 增加了 FCC 认证时 1KHz 音频输入要求
V1.3:07/08/2016	王同	更新 A3 版性能参数

目 录

1 文档说明	4
2 HS6760 芯片简介	5
3 HS6760 硬件设计	7
3.1 原理图设计	7
3.1.1 带 MCU 的应用	7
3.1.2 免 MCU 的应用	7
3.1.3 天线匹配网络	8
3.1.4 过 FCC/CE 认证电路	9
3.2 PCB 设计	9
4 HS6760 软件设计	10
4.1 控制接口总线	10
4.2 复位及初始化 (Reset&Initial)	11
4.3 设置频点 (Set Frequency)	13
4.4 设置功率 (Set Power)	14
5 HS6760 认证测试	16
5.1 功率设置	16
5.2 调制频偏设置	17
5.3 音频输入	17

1 文档说明

1、本文档为 HS6760 芯片应用指导，包括硬件设计，软件设计，FCC/CE 认证测试等。

2、更多信息请阅读《HS6760_DataSheet》。

2 HS6760 芯片简介

HS6760 是一款高性能立体声调频发射芯片。

HS6760 支持 MCU 控制模式和免 MCU 模式，是一个低成本无线系统解决方案。可以满足手机和其他手持电子产品客户用调频信号来传输音频的需求。

芯片特点：

- 音频输入：
 - 支持 line-in 输入和直接 MIC 输入
 - 内置 VGA 可调增益(-6~24dB@3dB/step)
- FM 发射性能：
 - 支持单声道和立体声，不支持 RDS
 - FM 频率：27~125MHz;
 - 频率步进：最小支持 50kHz;
 - 50us, 75us 预加重可选;
 - 调制频偏：75kHz/50kHz/22.5kHz 三种可选
 - 发射信噪比 ≥ 60 dB (立体声);
- 输入参考时钟支持 7.6MHz/12MHz/24MH 三种晶振
- 发射功率：
 - 外接电感情况下 >10 dBm;
 - 支持天线调谐
 - 针对非 50 欧姆的天线支持阻抗变换得到更高的发射功率
- 电池输入： 2.2-4.2V; 支持 2 节/3 节干电池和锂电池
- 免 MCU 发射功能：（选择 12MHz 晶体）
 - 通过外部电阻实现 8 档调台
 - 可以选择单声道或立体声
- 寄存器接口：标准两线 I2C 接口，支持免 MCU 应用（复用 SDA, SCK）
- SSOP-10 封装

典型应用:

- ◆ 车载 FM 发射
- ◆ 传音棒
- ◆ 玩具对讲机
- ◆ 无线 mic (导游)
- ◆ 婴儿监听

3 HS6760 硬件设计

本章描述了基于 HS6760 的硬件设计，主要由两部分内容组成：原理图设计和 PCB 设计。

3.1 原理图设计

3.1.1 带 MCU 的应用

在有 MCU 的应用中，MCU 通过 I2C 总线配置 HS6760 的工作方式，可以灵活地选择参考输入时钟频率，FM 发射频率，输出功率大小，单声道还是立体声，是否打开预加重等参数。注意，HS6760 内部有 40K 的上拉电阻，所以 I2C 总线外面不需要外挂到 VCC 的电阻。如果距离太长，I2C 速度不够，则可以加小于 10K 的上拉电阻到 VCC。

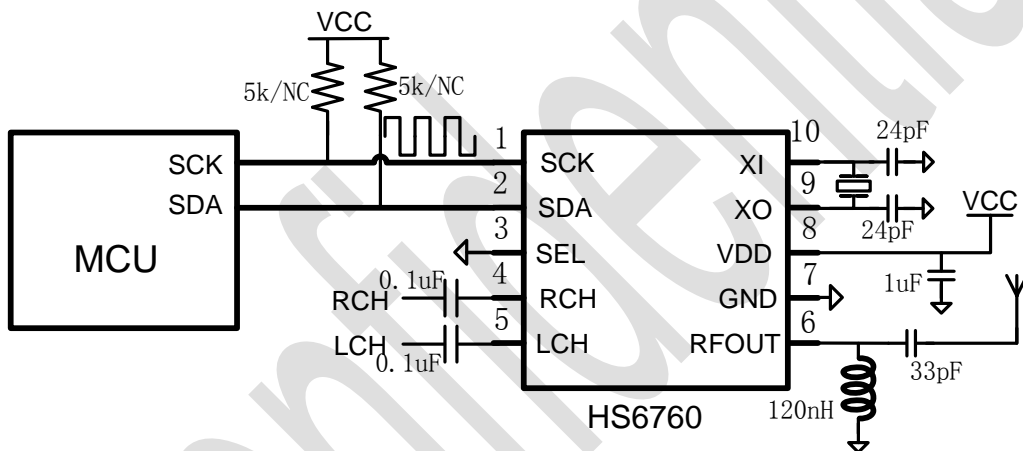


图 3-1 HS6760 带 MCU 应用电路

3.1.2 免 MCU 的应用

把 SEL 引脚接高电平，则芯片进入免 MCU 模式。此时不能通过 I2C 总线对 HS6760 进行配置，而只能工作在预设的频点上。在免 MCU 模式下，芯片一上电就进入发射模式，发射频点由 SDA 的电压来决定。芯片预先内置了 8 个频点，当 SDA 引脚的电压为 $[0, 1/8VCC, 2/8VCC \dots 7/8VCC]$ ，分别选择第 1~8 个预设的频点。因此可以很方便地使用一个分压电阻串与一个拨码开关来改变 SDA 点的电压从而改变工作频点。8 个预设的频点为 $[87.5M \ 88.1M \ 88.3M \ 88.5M \ 88.7M \ 40M \ 64M \ 76M]$ 。

在免 MCU 模式下，SCK 引脚复用为单声道/立体声的选择；接高电平表示工作在立体声模式下，接低电平表示为单声道模式。音频输入的 PGA 增益在此模式下为 24dB。其它预设的选项请见寄存器列表中的默认值。

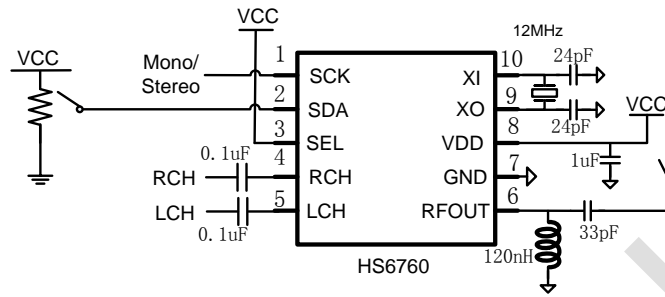


图 3-2 HS6760 免 MCU 应用电路

3.1.3 天线匹配网络

由于 HS6760 的发射频率要覆盖 27~125M，因此不同频段需要不同的 LC 调谐值。可以按照下表来选择 L、C 的值

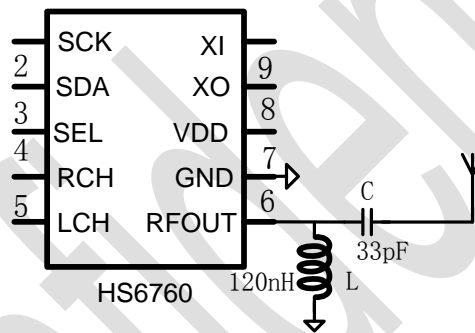


图 3-3 天线 LC 调谐

发射频段 (MHz)	27~45	45~76	75~87.5	87.5~108	108~125
L	220nH		120nH		
C	100pF		33pF		
寄存器设置	0x19=H 49	0x1A=H9 1	----	----	----
输出功率	11.5dBm	11dBm	>9dBm	>10dBm	>9dBm

3.1.4 过 FCC/CE 认证电路

需要在带 MCU 电路上加 1 个电容和 2 个电感，通过这个滤波电路来抑制多次谐波。此电路的通信距离可以达到 10 米。

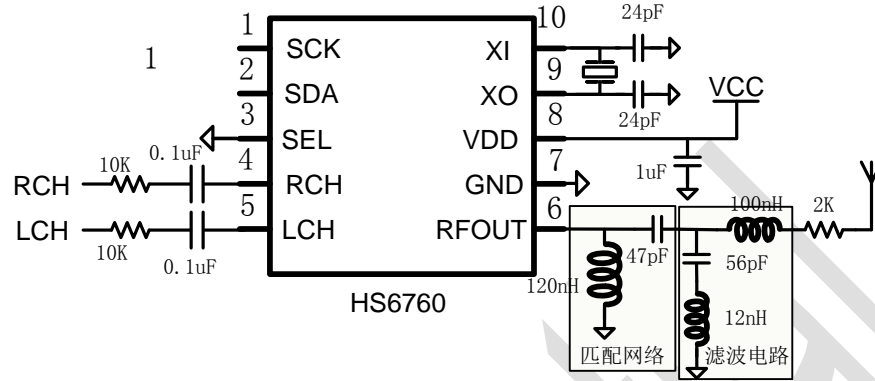


图 3-4 过 FCC/CE 原理图

3.2 PCB 设计

PCB Layout 以常用的 2 层板为例，此电路的设计，直接关系到芯片的通信性能。

PCB 材质采用最常用的双面 FR4 板材结构，元器件尽量放在 top 层，bottom 层尽量有完整的地平面，元器件的接地 PIN 应就近直接打过孔到 bottom 层。

4 HS6760 软件设计

4.1 控制接口总线

HS6760 可以通过 2 线 I2C 模式由一个 MCU 来控制，使用 MCU 控制芯片时，用户需要 SEL(PIN3)脚接地。HS6760 采用标准的 I2C 总线与 MCU 进行通信。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 (0011000b) 和一个读写 r/w 命令比特。

标准 I2C 读写方式：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 0011000B，加上读写标志位，即为 0x30(写操作)和 0x31(读操作)，读写方式的格式如下：



图 4-1 标准 I2C 写格式



图 4-2 标准 I2C 读格式

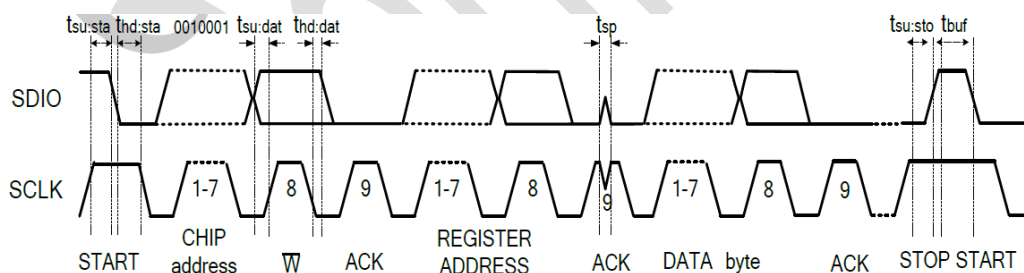
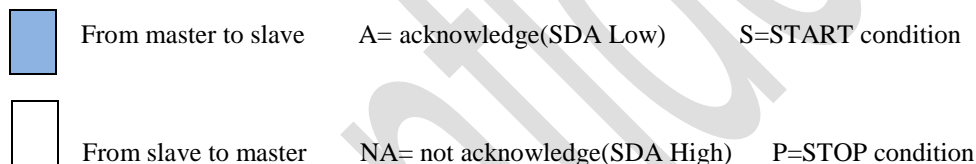


图 4-3 标准 I2C 写时序

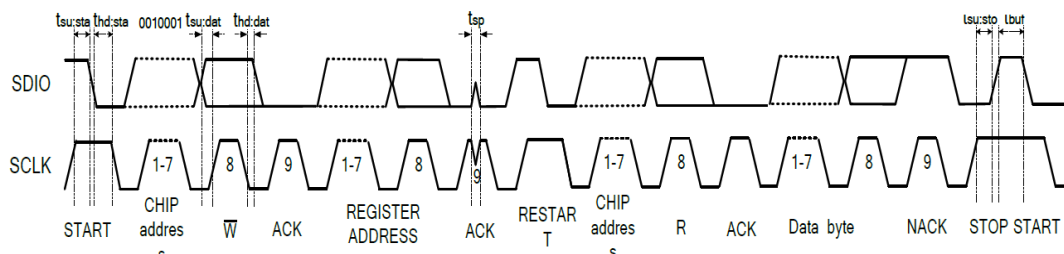


图 4-4 标准 I2C 读时序

4.2 复位及初始化（Reset&Initial）

HS6760 FM 发射芯片的初始化操作顺序：选择晶体频率→写初始化数据→复位。

选择晶体频率

HS6760 支持 7.6M、12M、24M 三种时钟频率，通过写寄存器 0x02 的 bit1 和 bit2 两位来实现。

Bit 2	Bit 1	晶体频率
0	1	7.6MHz
1	0	12MHz
1	1	24MHz

例如：

用户采用 7.6M 晶体时，可以写寄存器 0x02 为 0xf2

用户采用 12M 晶体时，可以写寄存器 0x02 为 0xf4

用户采用 24M 晶体时，可以写寄存器 0x02 为 0xf6

（一）写初始化数据

初始化操作通过 I2C 接口把初始化参数写到芯片内部来实现，主要工作有如下设置：

- 1.设置芯片工作模式为 Normal mode，通过写寄存器 0x01 为 0x06 来实现。
- 2.打开音频通路，通过写寄存器 0x11 为 0x01 来实现（仅 A1 版本需要此操作）。
- 3.手动调大频偏，通过写寄存器 0x1d 为 0x70 来实现（仅 A1 版本需要此操作）。

（二）复位操作

复位操作通过将寄存器 0x07 的最高位（bit7）先置 0，延时 10ms，再置 1 来实现。

例如：用户可通过先读取寄存器 0x07，将读到的值最高位清零后写入寄存器 0x07，延时 10ms 后，再将读到的值最高位置 1 后写入到寄存器 0x07。

C 语言参考代码：

➤ 初始化函数

void HS6760_initial(void)//初始化函数

```

{
    uint8_t RegData;
    #ifdef Clock_24M
        RegData=0x36;//PGA=12dB,24M 晶振
    #else
        #ifdef Clock_12M
            RegData=0x34;//PGA=12dB,12M 晶振
        #else
            #ifdef Clock_7v6M
                RegData=0x32;//PGA 为 12dB,7.6M 晶振
            #else
                #endif
            #endif
        #endif
    #endif
    I2C0_WriteBuffer(HS6760_I2C_ID,0x02,&RegData,1);
    Delay_ms(10);
    RegData=0x06;
    I2C0_WriteBuffer(HS6760_I2C_ID,0x01,&RegData,1);// normal 模式
    Delay_ms(10);
    RegData=0x01;
    I2C0_WriteBuffer(HS6760_I2C_ID,0x11,&RegData,1);//打开音频通路(仅 A1 版
需要此步操作)
    Delay_ms(10);
    RegData=0x70;
    I2C0_WriteBuffer(HS6760_I2C_ID,0x1d,&RegData,1);//增大频偏(仅 A1 版需要
此步操作)
    Delay_ms(10);
    HS6760_reset();
}

```

➤ 芯片复位函数

void HS6760_reset(void) //HS6760 芯片复位函数

```

{
    uint8_t Data8;

```

```

    I2C0_ReadBuffer(HS6760_I2C_ID,0x07,&Data8,1);//读取 0x07 寄存器的值到
Data8 中，读取长度为 1 个 Byte
    Data8&=0x7f;//最高位置 0
    I2C0_WriteBuffer(HS6760_I2C_ID,0x07,&Data8,1);//将 Data8 的数据写入到寄
存器 0x07 中，写入的长度为 1Byte
    Delay_ms(10);//延时 10mS
    Data8|=0x80;//最高位置 1
    I2C0_WriteBuffer(HS6760_I2C_ID,0x07,&Data8,1);//将 Data8 的数据写入到寄
存器 0x07 中，写入的长度为 1Byte
}

```

4.3 设置频点（Set Frequency）

HS6760 覆盖的波段范围为 27MHz~125MHz，最小频率步进为 50KHz。HS6760 芯片频点的设置可以通过配置寄存器 0x00 和 0x01 来实现。频点的计算步骤为：

- ① CurChan=频点×20(MHz) //频点为 10 进制
- ② 将 CurChan 转换为 16 进制
- ③ 将 16 进制 CurChan 的低 8bit 写入 0x00 寄存器
- ④ 将 16 进制 CurChan 的高 8bit 写入 0x01 寄存器

例如：要设置频点为 90.5MHz

- ① CurChan=90.5×20=1810
- ② CurChan=0x0712
- ③ 寄存器 0x00 写 0x12
- ④ 寄存器 0x01 写 0x07

频点设置完后，还应写寄存器 0x02 的最低位（bit0）为 0，延时 10ms，然后再写 0x02 的最低位（bit0）为 1，来使设置的频点生效。

C 语言参考代码：

```

/*****

```

```

Function: HS6760_SetFreq()

```

```

Parameter:

```

```

curFreq: FM frequency *10 ,such as 1017,933,1077..

```

```

Description:

```

```

    set FM to a frequency

```

```

*****/
void HS6760_SetFreq(uint16_t curFreq)
{
    uint16_t curChan;
    uint8_t Data8bit;
    uint8_t TmpData8bit[2];

    curChan=curFreq*2;
    TmpData8bit[0]=curChan>>8;//取高 8 位
    TmpData8bit[1]=curChan&0xff;//取低 8 位
    I2C0_WriteBuffer(HS6760_I2C_ID,0x01,&TmpData8bit[0],1);//
    Delay_ms(10);
    I2C0_WriteBuffer(HS6760_I2C_ID,0x00,&TmpData8bit[1],1);//
    Delay_ms(10);
    /*使写入的频点生效*/
    I2C0_ReadBuffer(HS6760_I2C_ID,0x02,&Data8bit,1);
    Data8bit&=0xfe;//最低位置 0
    I2C0_WriteBuffer(HS6760_I2C_ID,0x02,&Data8bit,1);//
    Delay_ms(10);
    Data8bit|=0x01;//最低位置 1
    I2C0_WriteBuffer(HS6760_I2C_ID,0x02,&Data8bit,1);//
    Delay_ms(10);
}
调用设置频点函数来更改发射频点
int main()
{
    //……(其它代码省略)
    HS6760_initial();
    HS6760_SetFreq(905);//设置频点为 90.5MHz
    HS6760_SetPow(31);//最大功率发射
}

```

4.4 设置功率（Set Power）

HS6760 芯片发射功率最大可达到 10dBm，最小为-15dBm。HS6760 芯片频点的设置可以通过配置寄存器 0x07 的低 5bit(bit4-bit0)来实现。当寄存器 0x07 的低

5bit 全置 1 时（11111 'b），发射功率达到最大，为 10dBm；当只有最低位置 1 时（00001 'b）发射功率最小，为-15dBm。如果该低 5bit 全置为 0（00000 'b），则关掉了发射 PA，此时只有很小的功率输出，约-45dBm，所以不建议用户将此寄存器的低 5bit 全置 0。

正常工作时更改发射功率步骤：

- ① 读取寄存器 0x07 的值
- ② 将读到值的最低 5bit 清零
- ③ 将要设置的功率大小的值写入该值的低 5bit
- ④ 将配置好值重新写入寄存器 0x07
- ⑤ 复位一下芯片

C 语言参考代码：

```

/*****
Function: HS6760_SetPower(uint8_t num)
Parameter: num=1~31 (Decimalism)
          curFreq: FM power dBm ,such as 1、2、…… 31
Description:
          set FM to a Power
*****/
void HS6760_SetPow(uint8_t num)
{
    uint8_t Data8;

    I2C0_ReadBuffer(HS6760_I2C_ID,0x07,&Data8,1);//步骤①
    Data8&=0xe0;//把低 5 位清零， 步骤②
    Data8+=num;//步骤③
    I2C0_WriteBuffer(HS6760_I2C_ID,0x07,&Data8,1); //步骤④
    HS6760_reset();//复位 HS6760 芯片， 步骤⑤
}

```

5 HS6760 认证测试

本节主要介绍出口产品需要过的 FCC 标准（美国）和 CE 标准（欧洲）。

结合第 3 节中 FCC/CE 认证电路，本节主要介绍过认证时的软件设计。

5.1 功率设置

过 FCC/CE 认证需要测试 FM 频段的高（88.1MHz）、中（98.1MHz）、低（107.9MHz）三个频点的发射功率和杂散。为了抑制高次谐波，在天线的发射段加入了滤波电路，如图 3-2 所示。但是该电路对不同频率的抑制能力不同，这就需要软件调节发射功率来达到预期的效果。

为使三个频点的发射功率约在-45dBm，可按照表 5-1 进行功率设置。

频点	发射功率	寄存器 0x07 低 5bit
88.1MHz	-45dBm	00001 'b
98.1MHz	-45dBm	00010 'b
107.9MHz	-45dBm	00011 'b

表 5-1 不同频点功率设置

C 语言参考代码：

```
#define test_88v1 0
#define test_98v1 1
#define test_107v9 2
/*****
Function: FCC_test(uint8_t fre)
Parameter:
    fre:such as test_88v1 /test_98v1 /test_107v9

Description:
    set FM to a power according to frequency
*****/
void FCC_test(uint8_t fre)
{
    switch (fre)
    {
        case 0: HS6760_SetFreq(881);HS6760_SetPow(1);break;
```



```

        case 1: HS6760_SetFreq(981);HS6760_SetPow(2);break;
        case 2: HS6760_SetFreq(1079);HS6760_SetPow(3);break;
        default:break;
    }
}
int main()
{
    HS6760_initial();
    FCC_test(test_88v1);
    HS6760_Fre_dev();
}

```

5.2 调制频偏设置

HS6760 调制频偏支持 75kHz/50kHz/22.5kHz 三种，过认证时可选择 22.5KHz 档。设置步骤如下：

- ① 读取寄存器 0x03 的值
- ② 将读到值的最低 2 位清零
- ③ 将读到值的最低 2 位置为 10'b
- ④ 将修改好的值重新写回到寄存器 0x03

C 语言参考代码：

```

void HS6760_Fre_dev(void)
{
    uint8_t Data8;
    I2C0_ReadBuffer(HS6760_I2C_ID,0x03,&Data8,1);//步骤①
    Data8&=0xfc;//步骤②
    Data8|=0x02;//步骤③
    I2C0_WriteBuffer(HS6760_I2C_ID,0x03,&Data8,1); //步骤④
}

```

5.3 音频输入

输入 1KHz 的正弦信号，信号幅度 $V_{pp} < 600mV$ 。