
(UM0001) User Manual

HS6200 应用手册

应用概述 Application overview

- 本文档为 HS6200 芯片应用指导，包括硬件设计，软件移植指导帮助，常见问题分析等。
- 更多信息请阅读《HS6200_DataSheet》。

参考文档 Reference documents

- 《HS6200_DataSheet》
- 《HS6200_reference_manual》

目录

1	HS6200 芯片简介	5
2	HS6200 硬件设计	6
2.1	原理图设计	6
2.1.1	电源滤波电路.....	7
2.1.2	时钟电路	7
2.1.3	天线匹配电路.....	7
2.1.4	接口电路	9
2.1.5	BOM清单	10
2.2	PCB设计	11
2.2.1	HS6200 芯片 Layout.....	11
2.2.2	VDD电源 Layout.....	12
2.2.3	16M晶体Layout.....	13
2.2.4	控制线Layout	13
2.2.5	射频匹配电路Layout	14
2.3	2.4G天线设计.....	16
2.3.1	鞭状天线	16
2.3.2	F_PIFA天线.....	17
2.3.3	Normal_PIFA天线	19
2.3.4	正常Wiggle天线.....	22
2.3.5	Mini_PIFA天线	23
2.3.6	天线设计总结.....	25
3	HS6200 软件设计	27
3.1	基本描述	27

3.1.1 SPI接口	27
3.1.2 CE控制	29
3.1.3 IRQ描述.....	30
3.1.4 BANK切换	30
3.1.5 静态和动态数据包长功能.....	31
3.1.6 自动回复ACK功能(auto ACK)	31
3.1.7 自动重传功能（ART）	32
3.1.8 packet描述	32
3.1.9 常用操作时间计算.....	34
3.2 工作流程	35
3.2.1 HS6200 初始化流程及配置.....	35
3.2.2 HS6200 收发数据流程	37
3.2.2.1 发送数据.....	37
3.2.2.2 接收数据.....	38
3.3 驱动程序	40
3.3.1 介绍	40
3.3.2 函数表	40
3.3.3 函数描述	41
3.3.4 驱动移植	45
3.4 跳频方式	46
3.4.1 频率表	46
3.4.2 被动跳频方式.....	46
3.4.3 主动跳频方式.....	48
4 HS6200 Module 射频测试	50
4.1 测试准备	50
4.2 发射功率测试	51

4.3 接收灵敏度测试	51
4.3.1 接收灵敏度测试方法.....	51
4.3.2 接收灵敏度测试步骤.....	51
5 HS6200 认证测试	53
5.1 传导杂散测试结果.....	53
5.2 辐射杂散测试结果.....	54
5.3 FCC测试前Checklist.....	55
6 HS6200 常见问题总结	55
6.1 故障排查流程	55
6.2 如何控制HS6200 发送载波	56
6.3 怎样使Power down模式下电流消耗最低.....	56
6.4 如何更改发射速率.....	56
6.5 如何更改发射功率.....	56
6.6 为什么我设置的自动重发次数ARC为非 0，系统却没有自动重发.....	57
6.7 在跳频通信中，偶尔能收到数据，而定频却没有问题.....	57
6.8 HS6200 的可以软复位操作么.....	57
6.9 用HS6200 的裸die直接邦定(Bonding)在PCB 上需要注意什么？	58
6.10 晶体需要外接电容么.....	58
6.11 如何增强HS6200 的ESD能力	58
6.12 HS6200 过CE认证需要注意的问题？	59
6.13 HS6200 过FCC认证需要注意的问题？	59
7 版本变更记录Revision history	59

1 HS6200 芯片简介

HS6200 是一款嵌入基带通信协议的单片 2.4GHz 收发芯片，工作在 2.4GHz~2.483GHz 世界通用的 ISM 频段，适用于超低功耗无线应用领域。

HS6200 无线通信系统的设计只需要一颗 MCU 和少数外围器件，是一个低成本无线系统解决方案。

调制方式采用 GFSK/FSK 调制，最大发射功率可以达到 8dBm，接收灵敏度在 1Mbps 速率下可以达到 -88dBm。

片内集成电压调节器，可确保高电源抑制比（PSRR）和宽电源电压范围。最低工作电压为 1.8V，最高工作电压为 3.6V。

芯片特点：

- ◆工作频段：2400~2483MHz
- ◆低工作电压：1.8~3.6V
- ◆空口速率：500Kps、1 Mbps 、2Mbps
- ◆SPI 接口：4 线制，最大速率为 10Mbps
- ◆调制方式：GFSK/FSK
- ◆可用低成本晶体：±60ppm
- ◆硬件自动应答及自动重传
- ◆快速频道切换，可应用于跳频算法
- ◆20 引脚 QFN 4mm×4mm 封装，SOP16

封装

典型应用：

- ◆无线键鼠
- ◆无线航模
- ◆无线工控
- ◆无线 LED
- ◆智能家居
- ◆无线音频

2 HS6200 硬件设计

本章描述了基于 HS6200 的 2.4G 无线终端的硬件设计，主要由三部分内容组成：原理图设计、PCB 设计和 2.4G_PIFA 天线设计。该硬件方案可广泛应用于计算机周边及消费类无线终端设备中，如无线键鼠、无线航模和无线 LED 等。

2.1 原理图设计

HS6200_DEMO 原理图由四部分组成，分别是电源滤波电路、时钟电路天线匹配电路和接口电路组成。下图给出了 HS6200_DEMO 最简外围电路原理图。

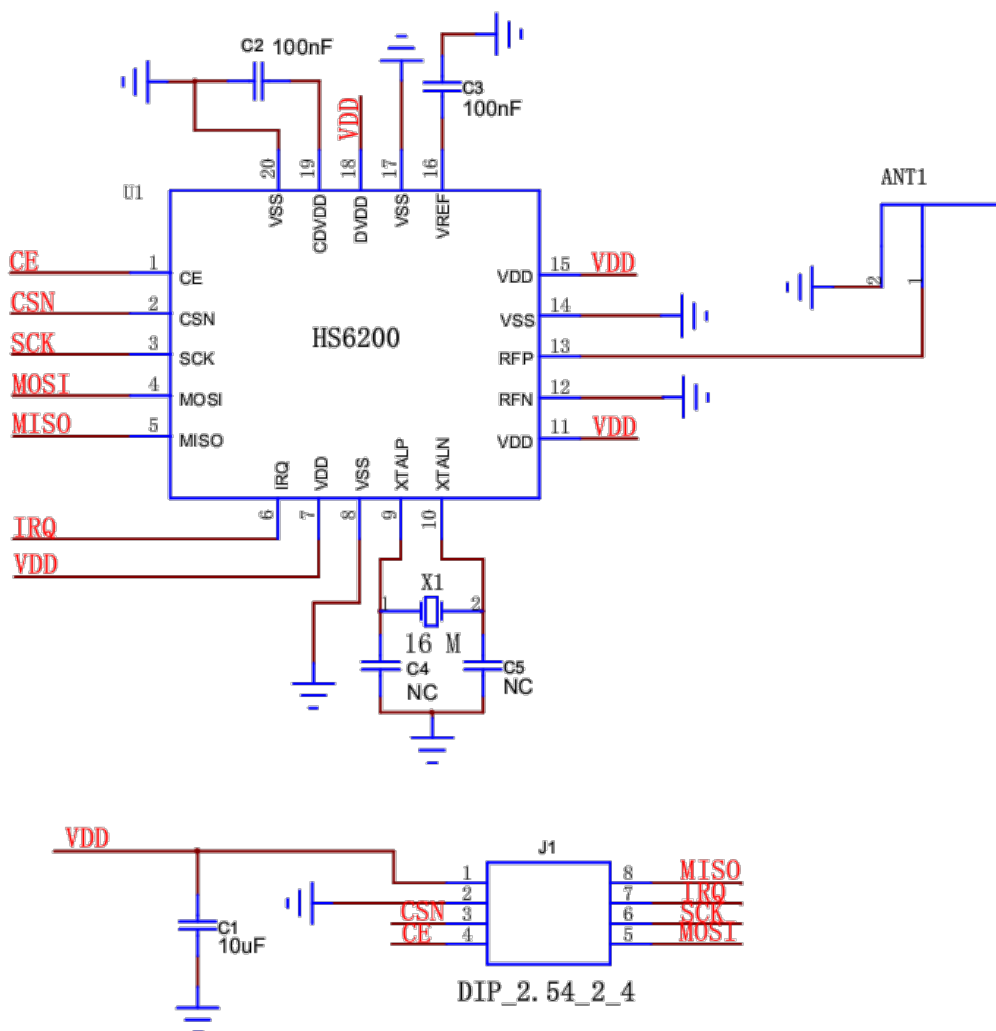


图 2.1 HS6200_DEMO 最简外围电路原理图

2.1.1 电源滤波电路

外部 LDO 给 HS6200 供 3.3V 电压, 需要经过滤波电路, 得到干净的电源, 如图 2.1 中的 C1, $C1=10\mu\text{F}$ 。LAYOUT 时, 尽量靠近芯片放置。

HS6200 内部 LDO 也需要滤波电路, 如图 2.1 中的 C2 和 C3, $C2=C3=100\text{nF}$ 。LAYOUT 时, 尽量靠近芯片的 PIN 脚放置。

2.1.2 时钟电路

晶体采用 16MHz, 频率误差为 $\pm 60\text{ppm}$ 。由于 HS6200 片内集成了负载电容, 外部的负载电容可不焊接, 具体如图 2.1 所示。

2.1.3 天线匹配电路

对于不同的应用需求和应用环境, 天线的匹配电路也不同。根据应用需求和应用环境, 可以分三种情况, 具体如下:

(1) 不要求过 FCC 或 CE 标准, 使用 HS6200 最简外围电路, 外围器件只需 4 个器件。具体如图 3-1 所示。

(2) 要求过 FCC 或 CE 标准, 天线的匹配网络采用 3 个器件, C6, $C6=1.5\text{pF}$, C7, $C7=0.5\text{pF}$, L1, $L1=2.7\text{nH}$, 具体如图 2.2 所示。

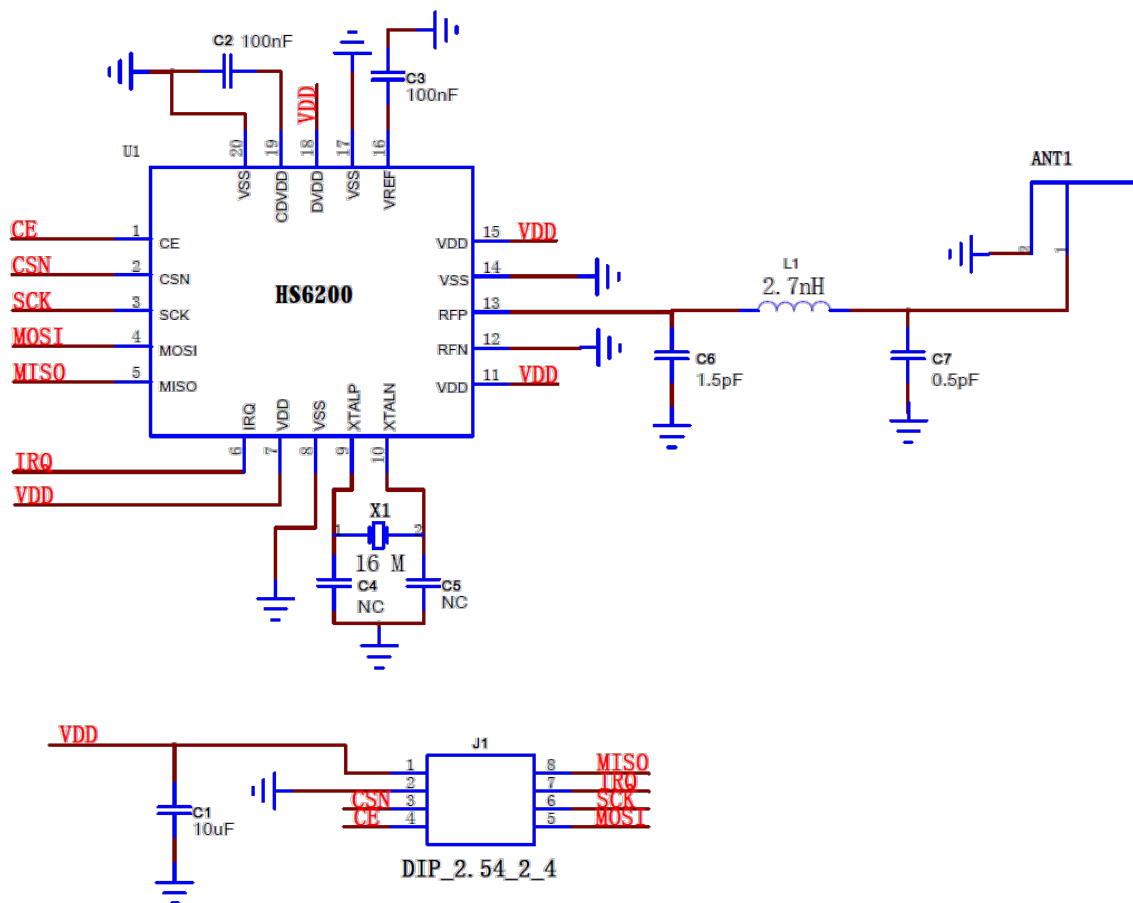
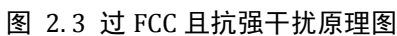


图 2.2 过 FCC/CE 原理图

(3)要求过 FCC 或 CE 标准,且强干扰环境下,天线的匹配网络可以使用 7 个器件, C6=1.8pF, C7=1pF, C8=4.7pF, C9=1.2pF, L1=2.7nH, L2=3.6nH, L3=2.7nH。此匹配电路可以使 HS6200 在强干扰环境中,还能表现出比较优的通信性能,具体如图 2.3 所示。



根据 MCU 的 I/O 资源情况，HS6200 芯片的片选引脚 CE，有两种接法，一种是使用 GPIO 控制，另一种是直接上拉 VDD，通过软件配置寄存器 PRE_GURD 的 Bit15 位控制，具体如下图所示。MCU 和 HS6200 通信，采用标准的四线制 SPI 接口（CSN，SCK，MISO，MOSI），用于切换工作模式、读取状态信息等。HS6200 的中断状态可以通过 IRQ 引脚读取，也可以读取相应的中断寄存器。

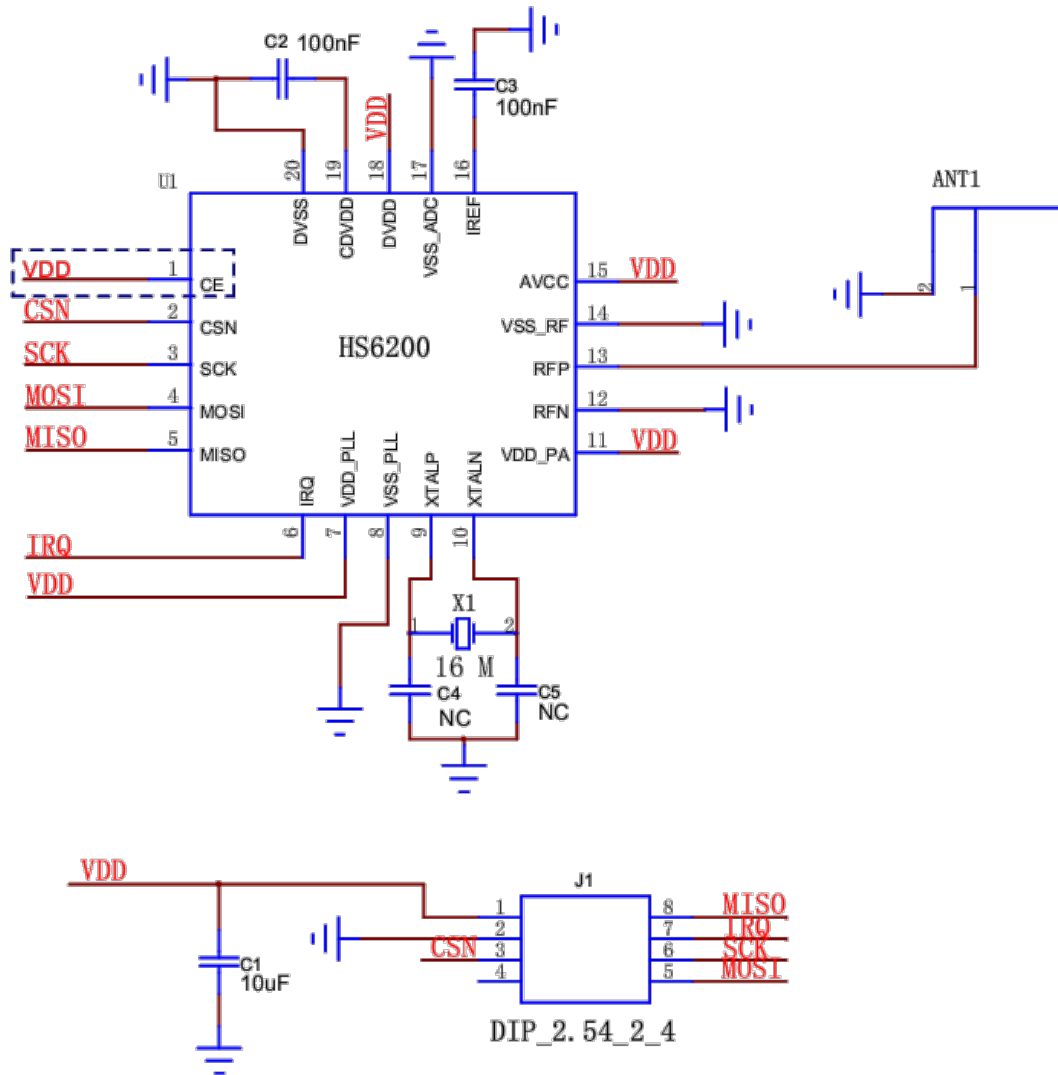


图 2.4 接口电路（寄存器控制 CE 方式）

2.1.5 BOM 清单

各个元件的具体型号及精度要求请参考 BOM 表，其中射频匹配电路部分元器件采用为 Murata 公司的器件，0402 封装：电感 Murata LQG15HS 系列；电容 Murata GRM1555 系列，0603 封装：电感 Murata LQG18HS 系列；电容 Murata GRM1855 系列。如采用不同厂商的元器件，匹配值可能会有稍微差别，需要重新验证射频性能。下表给出的射频匹配电路器件为 0402 封装。

表 2.1 BOM 清单

Ite m	Qty	Ref	Value	Footprint	Description	Vendor
1	1	U1	HS6200	QFN20	HS6200 chip	HunterSun

2	1	ANT1	2.4G PIFA	2.4G_Ant	2.4G PIFA Antenna	
3	1	X1	16MHz	XTAL	+/-60ppm	XTY
4	1	J1	4*2 插针	2.54mm	接口	Various
5	1	C1	10uF	0603	X5R, +/-10%, 6.3V	Various
6	2	C2,C3	100nF	0402	X5R, +/-10%, 6.3V	Various
7	2	C4,C5	12pF/NC	0402	NPO +/-5%, 6.3V	Various
8	1	C6	1.8pF	0402	NPO +/-0.25pF, 6.3V	Murata
9	1	C7	1pF	0402	NPO +/-0.25pF, 6.3V	Murata
10	1	C8	4.7pF	0402	NPO +/-0.25pF, 6.3V	Murata
11	1	C9	1.2pF	0402	NPO +/-0.25pF, 6.3V	Murata
12	2	L1,L3	2.7nH	0402	Inductor +/-0.3nH	Murata
13	1	L2	3.6nH	0402	Inductor +/-0.3nH	Murata

2.2 PCB 设计

PCB Layout 以常用的 2 层板为例,对各个部分的 Layout 进行详细阐述,特别是射频匹配电路的 Layout,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。

2.2.1 HS6200 芯片 Layout

HS6200 为 4mm*4mm QFN 封装的芯片,其芯片底下是需要接地的,PCB 做库的时候在芯片中心需加接地的大 pad,为保证与 PCB 底层 bottom 的地平面完整接地,pad 中心至少需要 4 个 GND 过孔 (Vias 大小为 10*15mil,如下图高亮过孔),而且手工焊接时需预先在 pad 上加锡处理。HS6200 芯片下面对应的 bottom 层尽量不要有走线和元器件,特别是靠近射频匹配电路的部分,完整的地平面能保证良好的射频性能。具体 Layout 如图 2.5 所示。

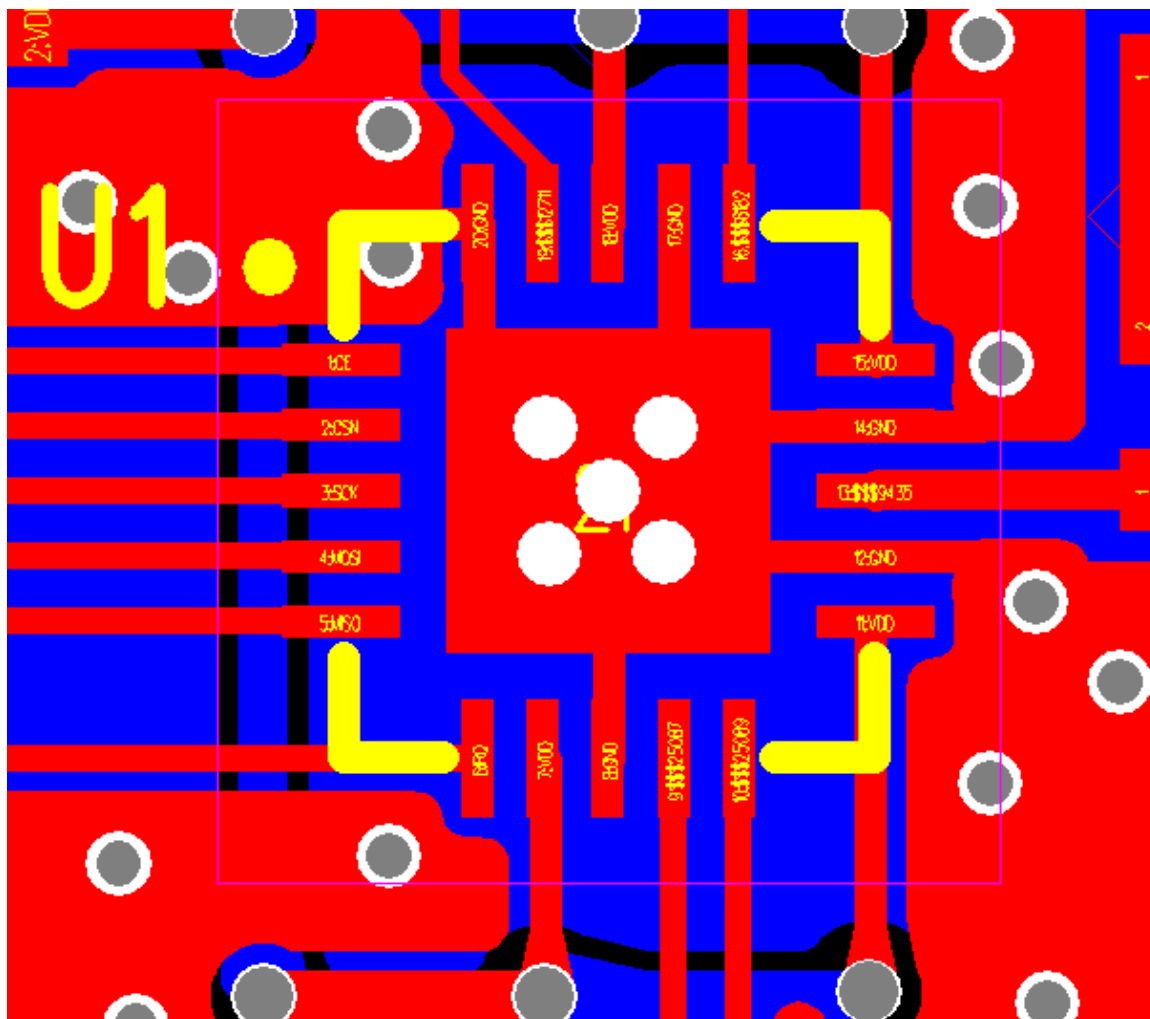


图 2.5 HS6200 芯片 Layout

2.2.2 VDD 电源 Layout

HS6200 的电源来自 MCU 的 LDO,两个并联电容对电源低通滤波,两个电容靠近 HS6200 芯片 PIN 脚放置,小电容需更靠近芯片 PIN 脚,电源走线需先进入电容 pad,再从电容 pad 引线出来,如图 2.6 高亮走线。

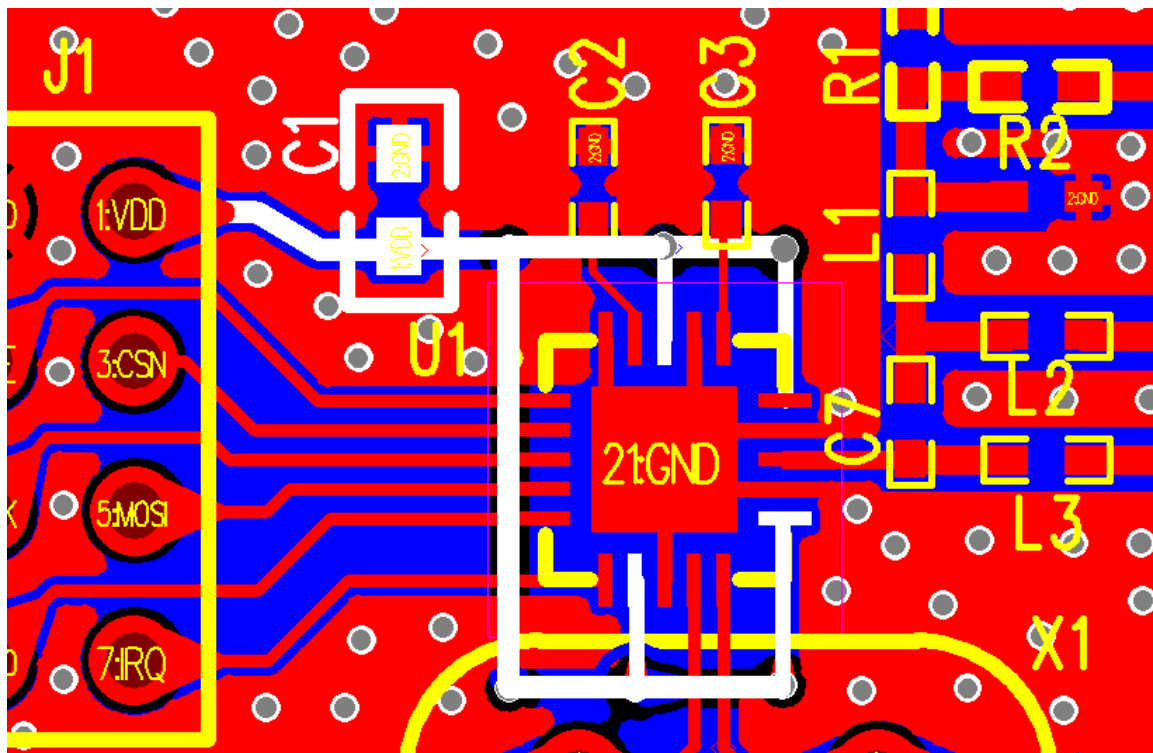


图 2.6 VDD 电源 Layout

2.2.3 16M 晶体 Layout

晶体对应的平面应尽量为完整地平面，最好不要有任何走线和元器件，图 2.7 给出两种晶体封装的 Layout。

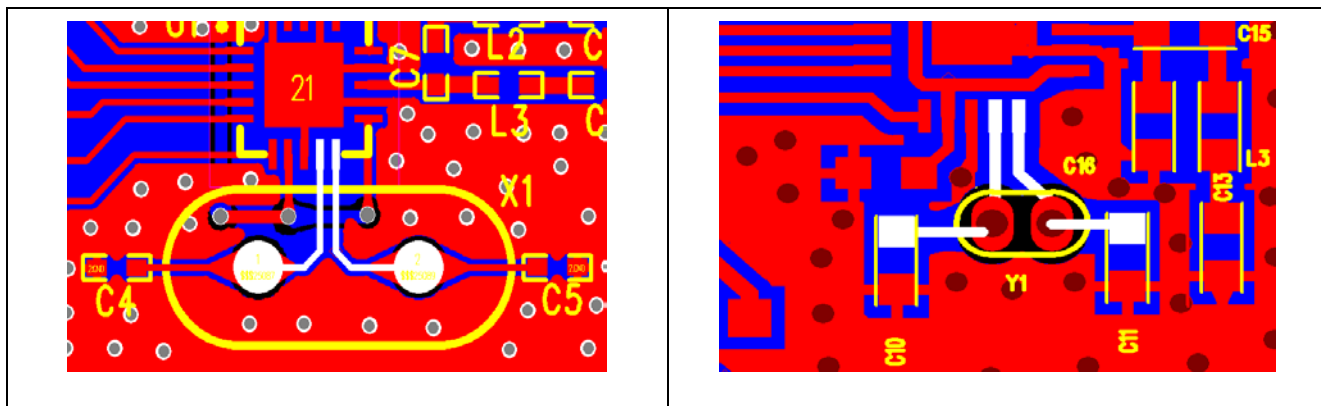


图 2.7 晶体的 Layout

2.2.4 控制线 Layout

HS6200 与 MCU 交互的 SPI 控制线，特别是其中两根对 HS6200 芯片输入的 SCK、MOSI 线，需要特别小心，尽量不要与其它器件共用，走线尽量短，布局时需考虑 MCU 的 SPI I/O 口靠近 HS6200 放置，线四周需包地处理，以免受外界干扰。

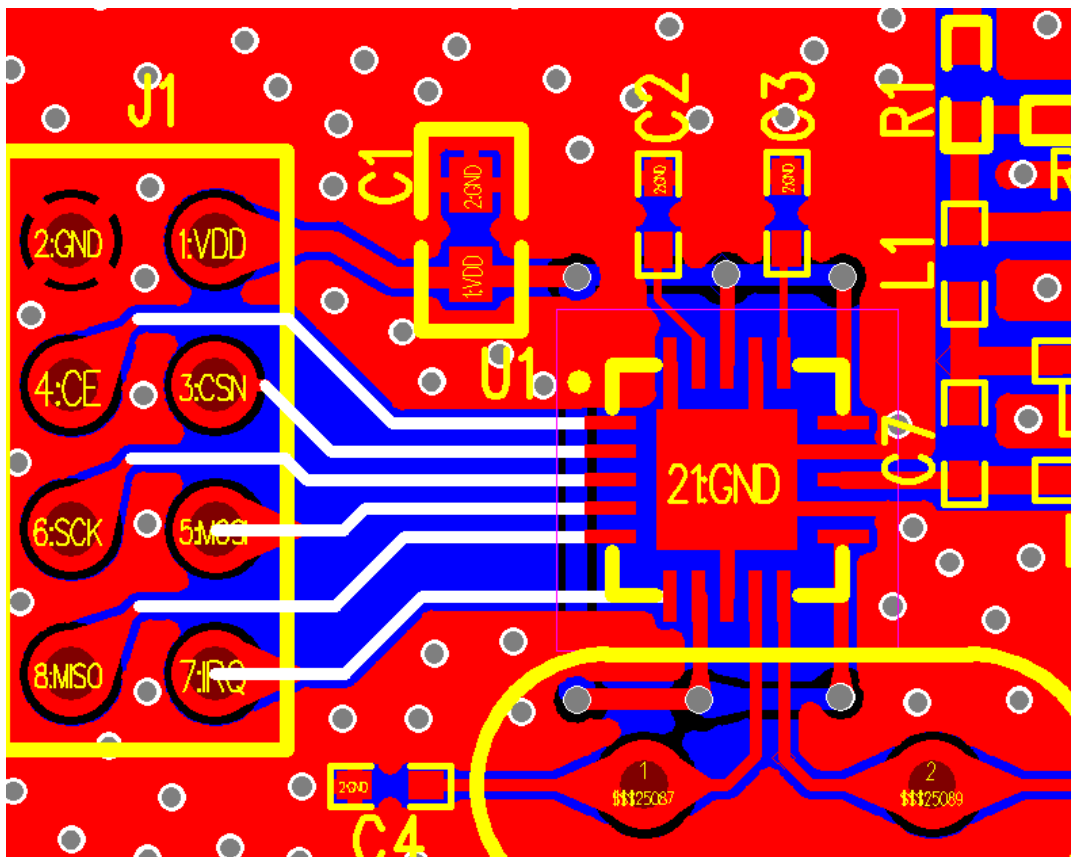


图 2.8 控制线 Layout

2.2.5 射频匹配电路 Layout

由于射频匹配电路的 PCB Layout 会严重影响到射频性能，强烈建议完全按照我们已经优化的值和 layout 进行布局和布线，包括元器件与铺铜地之间的间隔（推荐为 0.2mm），GND 过孔的选择（应尽可能的多），射频线不要与电源线和晶体线等交叉走线，更不能平行走线。下两图中高亮部分即为匹配电路的 Layout。

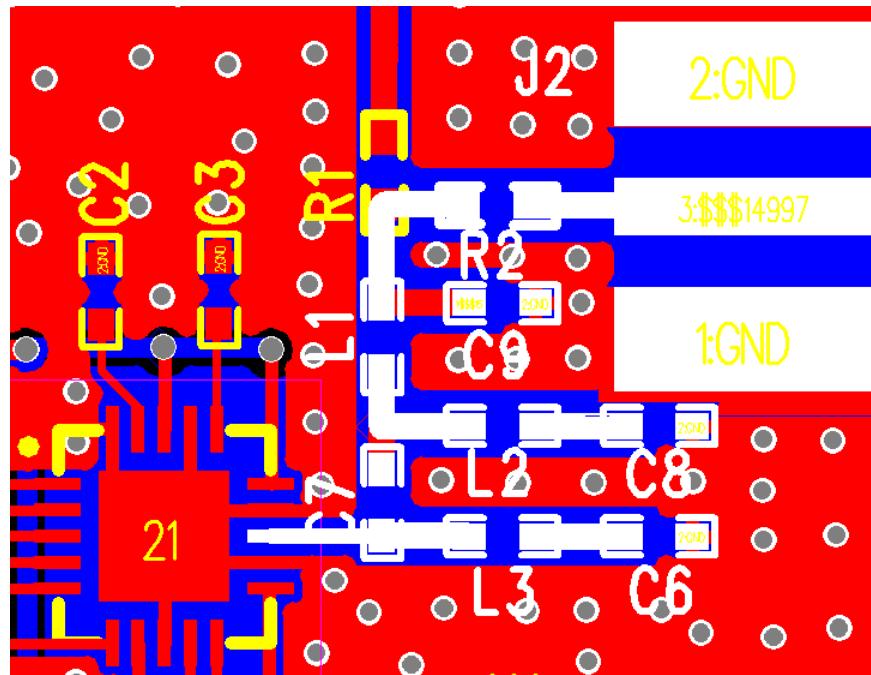


图 2.9 射频匹配电路的 Layout（接 SMA 接口）

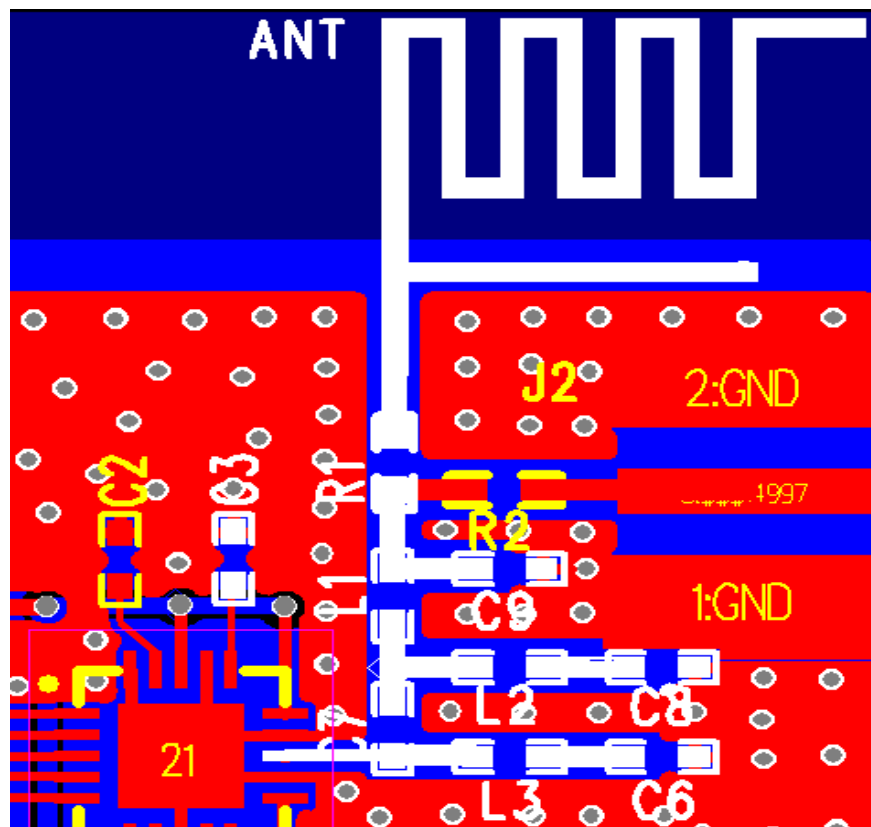


图 2.10 射频匹配电路的 Layout（接 PIFA 天线）

2.3 2.4G 天线设计

天线的设计关系到射频通信的性能，在选择 2.4G 天线时，需综合考虑 PCB 尺寸、通信性能、成本等因素。本节给出五种 2.4G 天线，供客户选择。

(1) 鞭状天线：材质采用铜，尺寸： $\phi 0.85 \times 30\text{mm}$ 。需要垂直焊接到 PCB 板上，对 PCB 平面尺寸无要求，但高度有要求（大于 30mm）。此天线的通信性比其他 4 种 PIFA 天线更优，通信距离更远，大约高出 20%。此天线适合高度无要求的产品，并应用在室外远距离通信，如无线航模的应用。

(2) F_PIFA 天线：微带线天线，尺寸： $4.9 \times 25\text{mm}$ 。此天线对 PCB 尺寸有要求，用于产品对 PCB 尺寸要求不高（一般尺寸 $10 \times 25\text{mm}$ ），通信性能次于鞭状天线，但成本低，便于生产。

(3) Normal_PIFA 天线：微带线天线，尺寸： $8 \times 13\text{mm}$ 。用于 PCB 无尺寸要求的条件下，所占 PCB 空间最大，最大增益可以达到 1.5dB，如 PCB 面积足够，建议用此天线。由此天线做成的 RF 模块板子大小为 $15\text{mm} \times 18\text{mm}$ 左右。

(4) Wiggle_PIFA 天线：微带线天线，尺寸： $7 \times 10\text{mm}$ 。用于 PCB 尺寸要求不苛刻的条件下，所占 PCB 空间比第二种稍小，增益也稍差 1dB，可以用于对体积稍有要求的无线终端，如对于空间比较紧凑的无线 LED 等设备。由此天线做成的 RF Module 板子大小为 $13\text{mm} \times 18\text{mm}$ 左右。

(5) Mini_PIFA 天线：微带线天线，尺寸： $2.6 \times 9.5\text{mm}$ 。此天线用于 PCB 板尺寸超小（如无线鼠标的 Dongle），但通信距离会短一半。此天线做成的 Dongle 大小为 $11\text{mm} \times 16\text{mm}$ 左右。

2.3.1 鞭状天线

鞭状天线采用铜芯的射频线，尺寸： $0.85 \times 30\text{mm}$ ，阻抗为 50ohm。鞭状天线需要垂直焊接到 PCB 板上，对 PCB 平面尺寸无要求，但高度有要求（大于 30mm）。此天线的通信性比其他 4 种 PIFA 天线更优，通信距离更远，大约高出 20%。此天线适合高度无要求的产品，并应用在室外远距离通信，如无线航模的应用。体如图 2.11 所示。

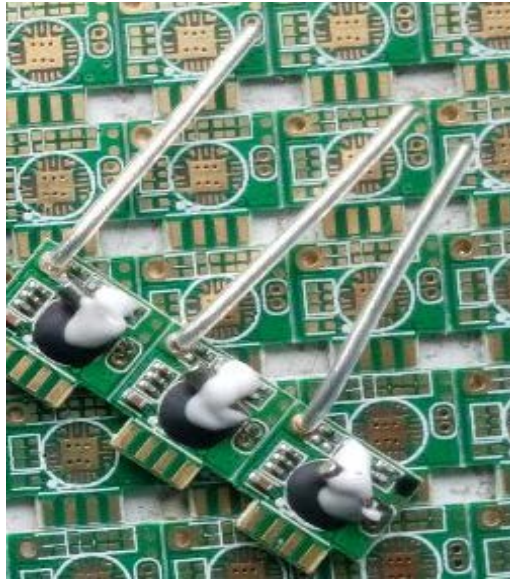


图 2.11 鞭状天线实物图

2.3.2 F_PIFA 天线

天线具体尺寸如图 2.12（板材为两层 FR4,板厚 1mm），图 2.13 为 S11 图，图 2.14 为电流图和图 2.15 为辐射图。该天线最大增益只有 2dB 左右。

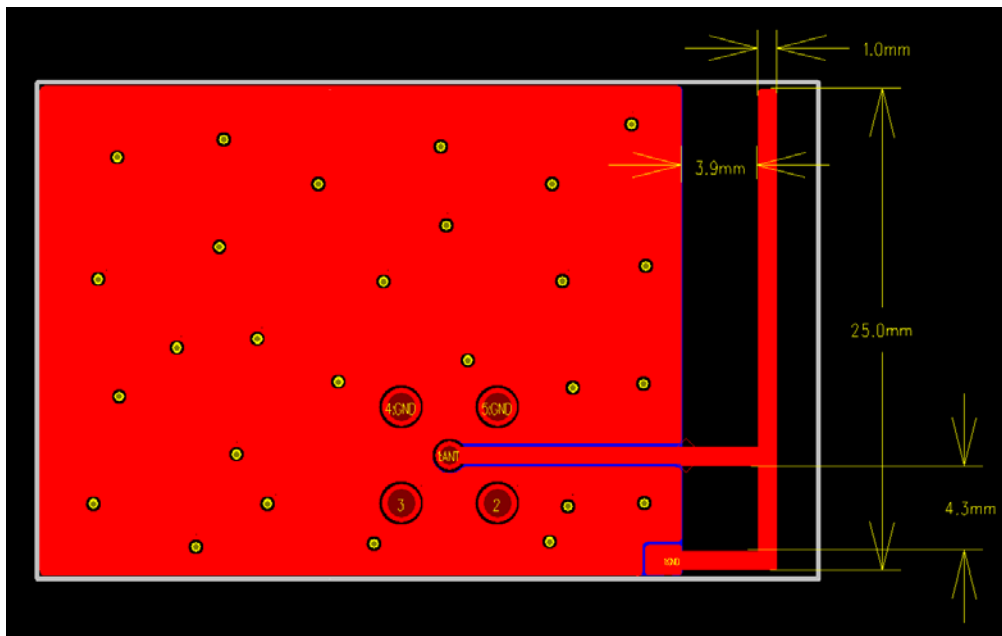


图 2.12 F_PIFA 天线的 PCB 尺寸

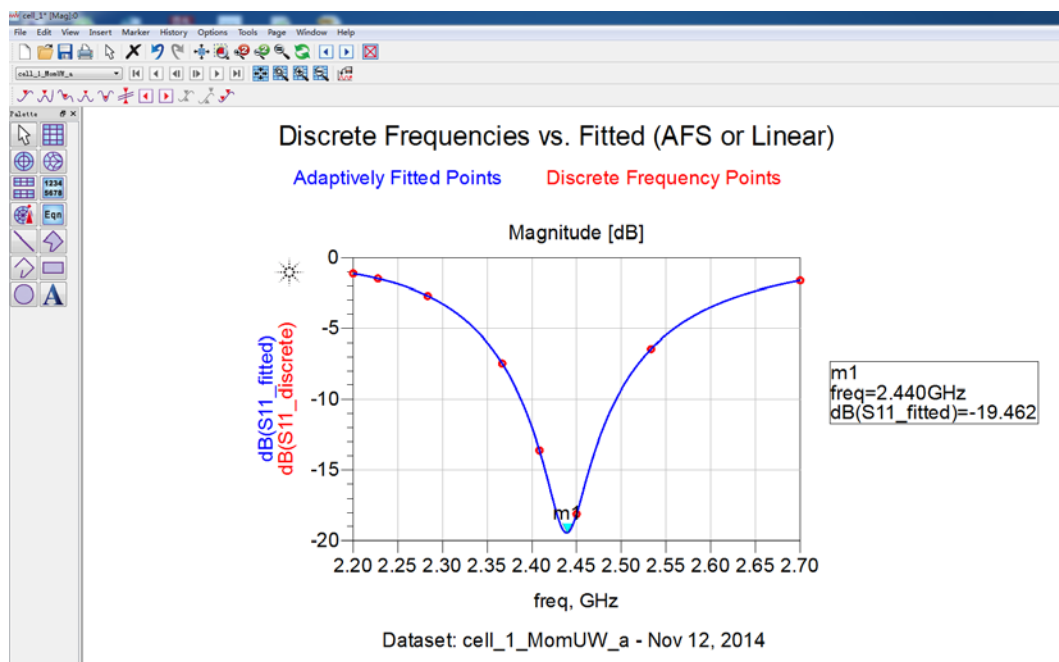


图 2.13 F_PIFA 天线的 S11 图

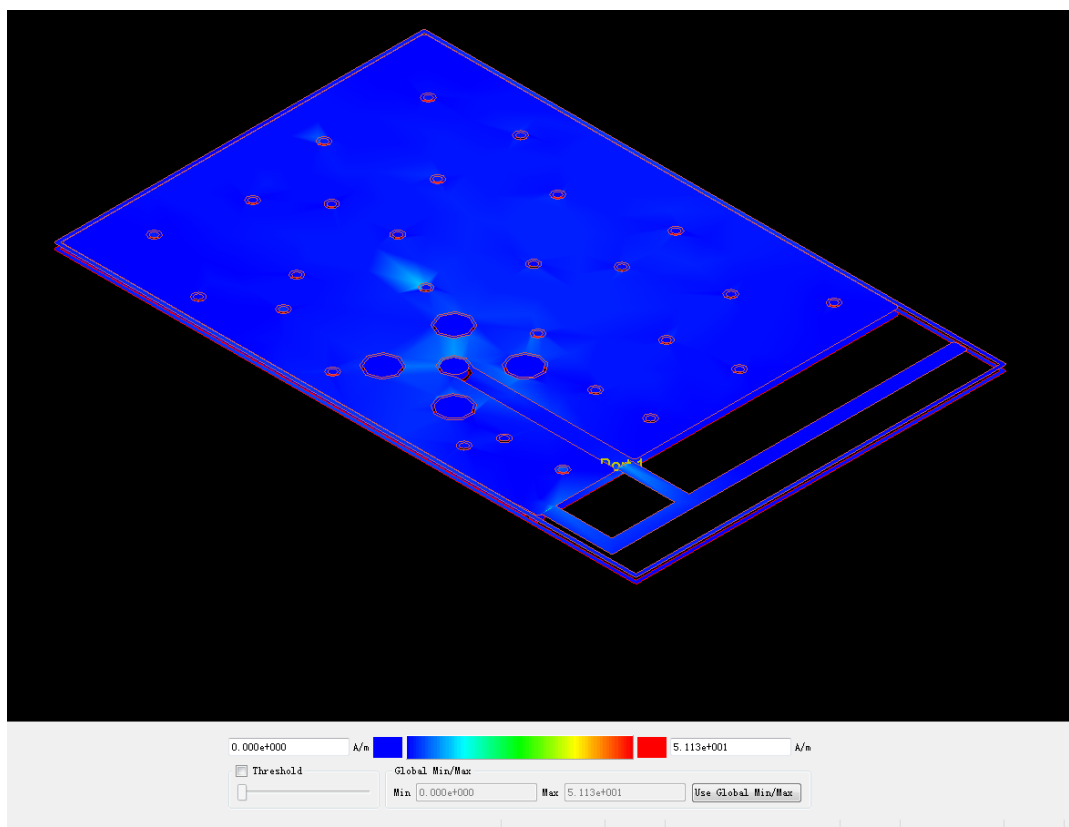


图 2.14 F_PIFA 天线的电流图

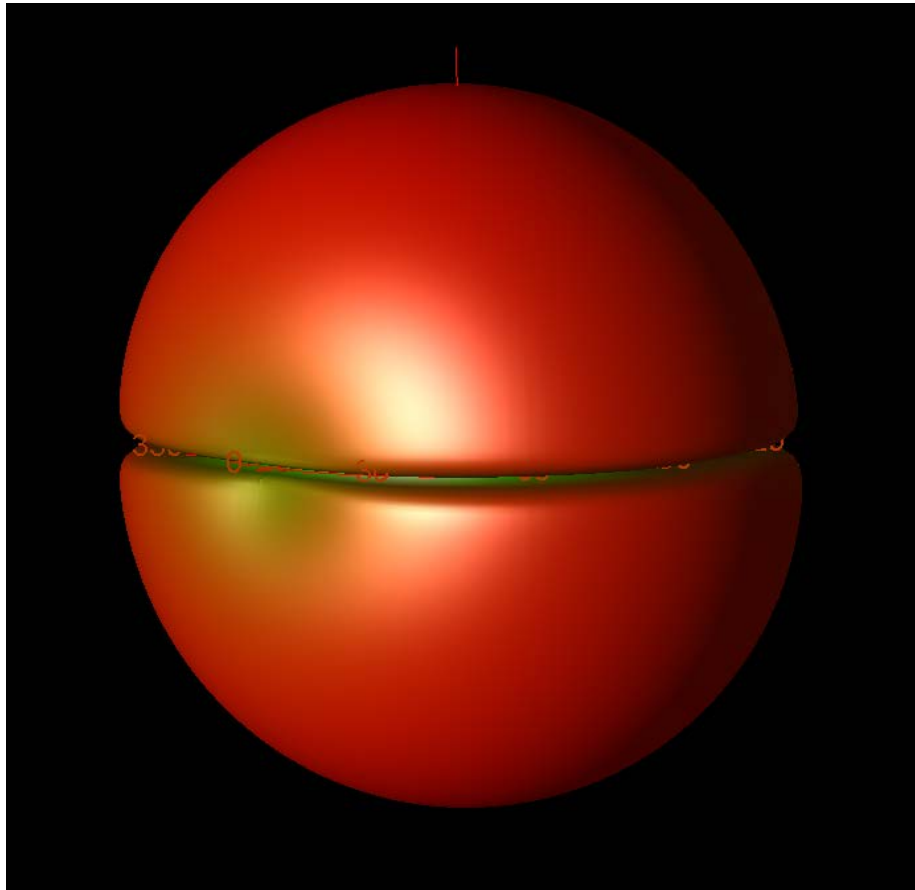


图 2.15 F-PIFA 天线的辐射图

2.3.3 Normal_PIFA 天线

天线具体尺寸如图 2.16 (板材为两层 FR4, 板厚 1.0mm), 图 2.17 为 S11 图, 图 2.18 为电流图和图 2.19 为辐射图。该天线最大增益只有 1.5dB。

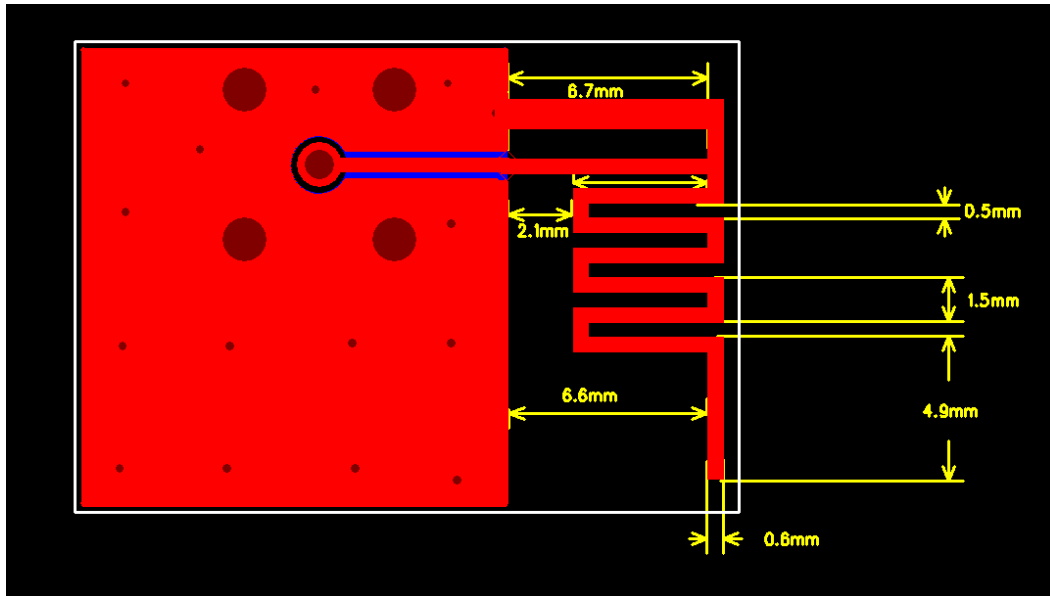


图 2.16 Normal_PIFA 天线的 PCB 尺寸

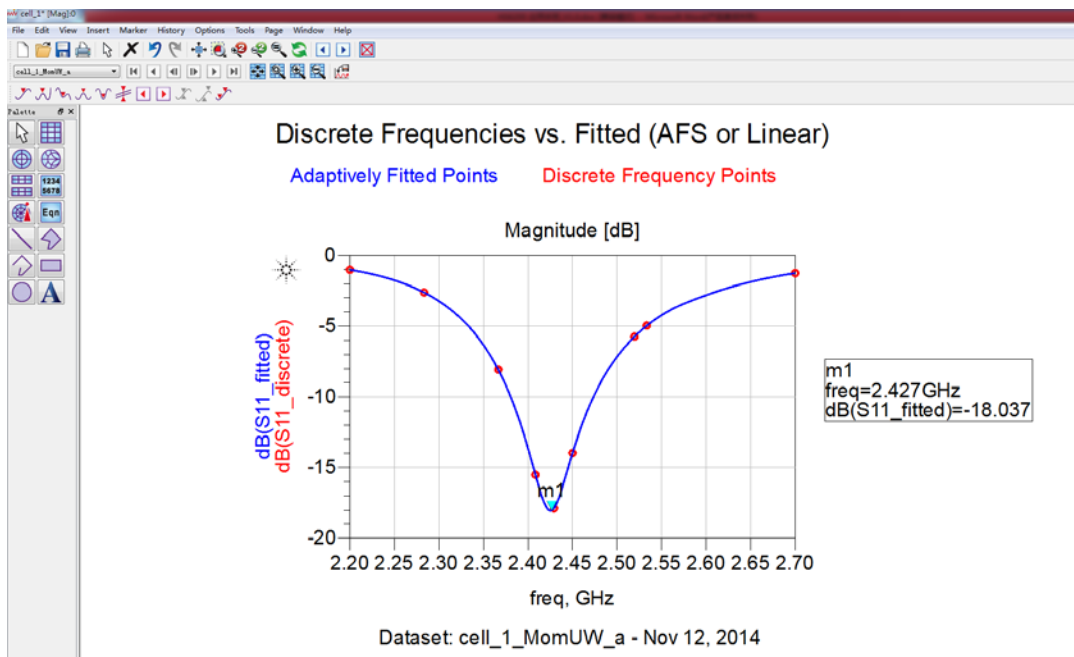


图 2.17 Normal_PIFA 的 S11 图

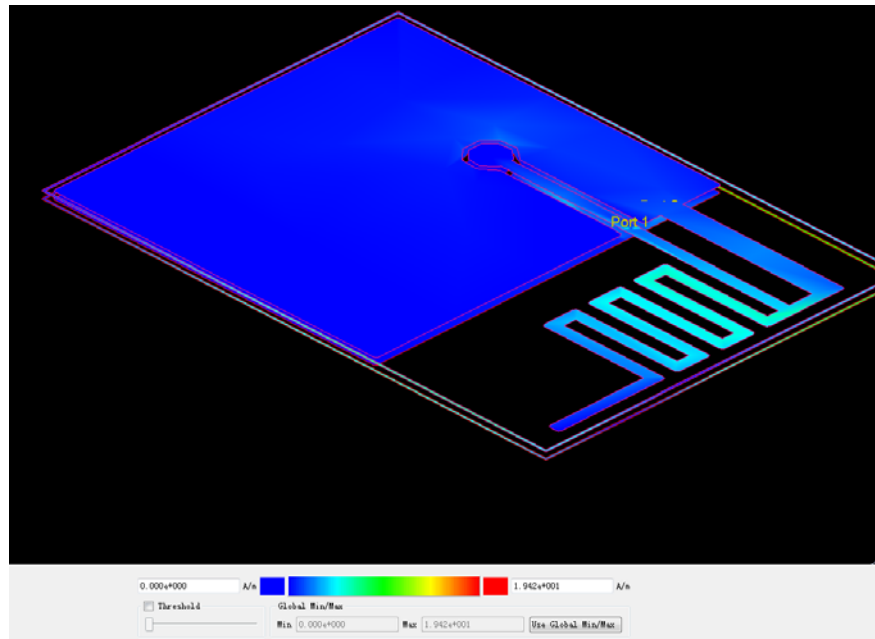


图 2.18 Normal_PIFA 天线的电流图

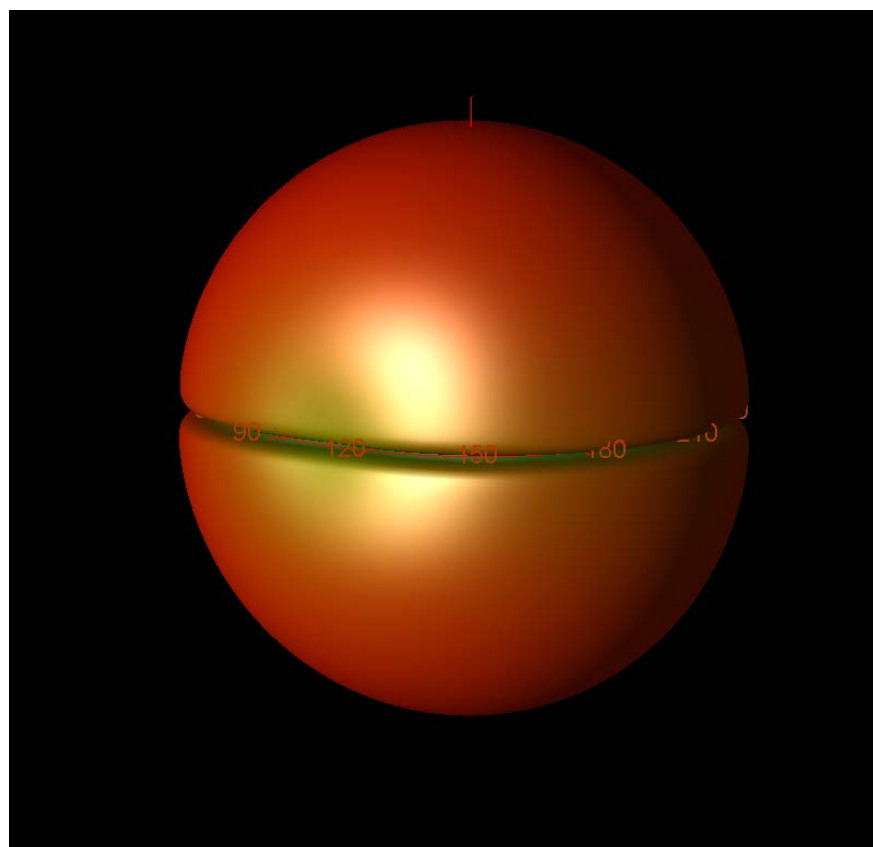


图 2.19 正常 PIFA 天线的辐射图

2.3.4 正常 Wiggle 天线

天线具体尺寸如图 2.20 (板材为两层 FR4, 板厚 1.0mm), 图 2.21 为 S11 图, 图 2.22 为电流图和图 2.23 为辐射图。该天线最大增益只有 0.5dB。

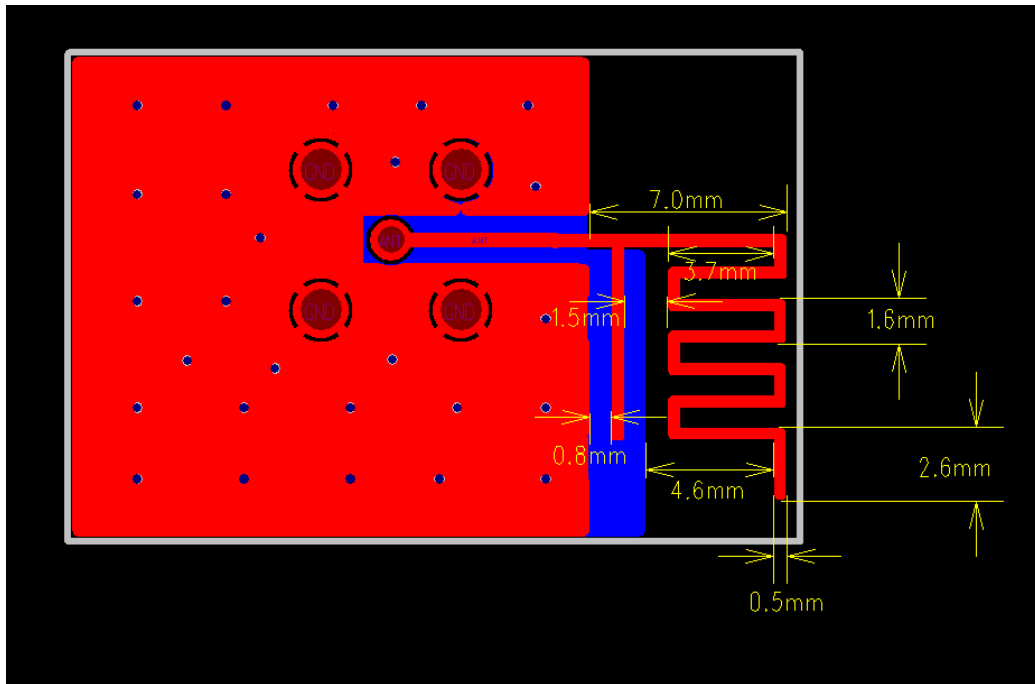


图 2.20 正常 wiggle 天线的 PCB 尺寸

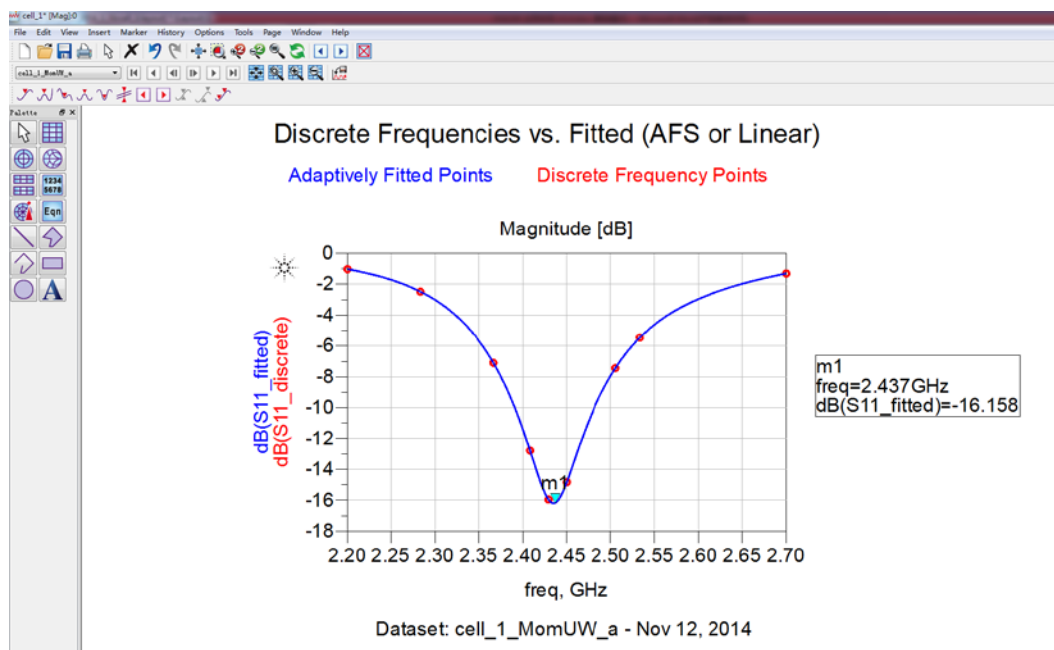


图 2.21 正常 wiggle 天线的 S11 图

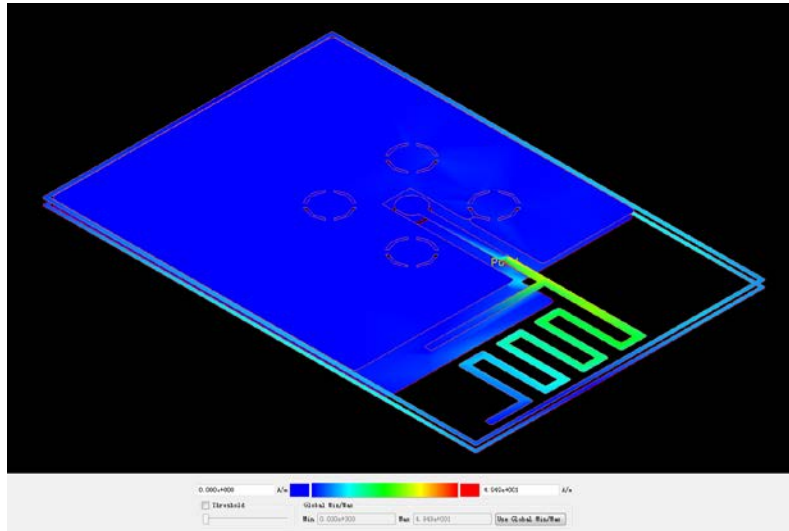


图 2.22 正常 wiggle 天线的电流图

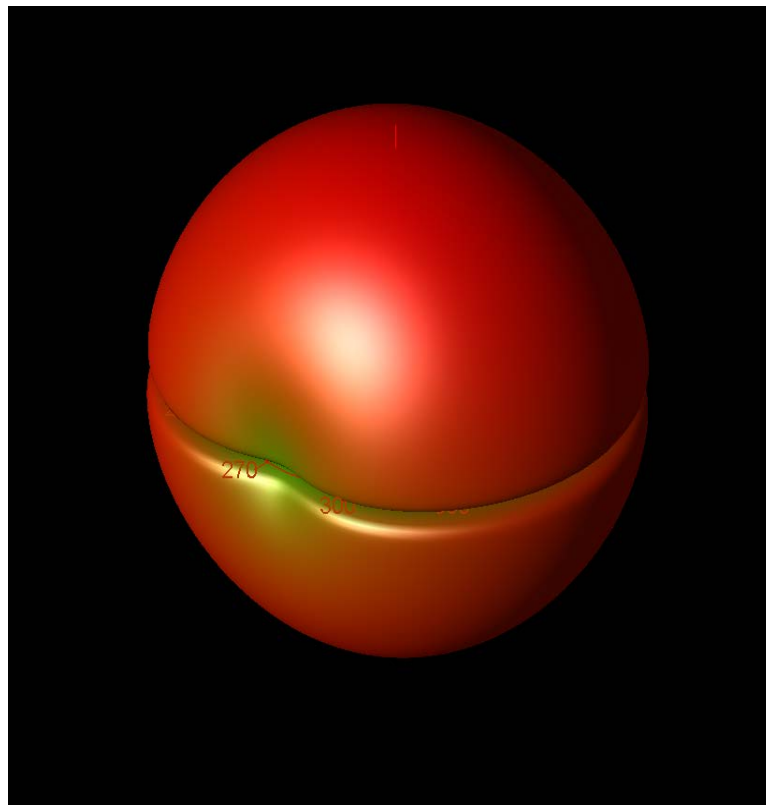


图 2.23 正常 wiggle 天线的辐射图

2.3.5 Mini_PIFA 天线

天线具体尺寸如图 2.24 (板材为两层 FR4, 板厚 0.6mm)。图 2.25 为 S11 图, 图 2.26 为电流图和图 2.27 为辐射图。该天线最大增益只有 -5dB 左右。

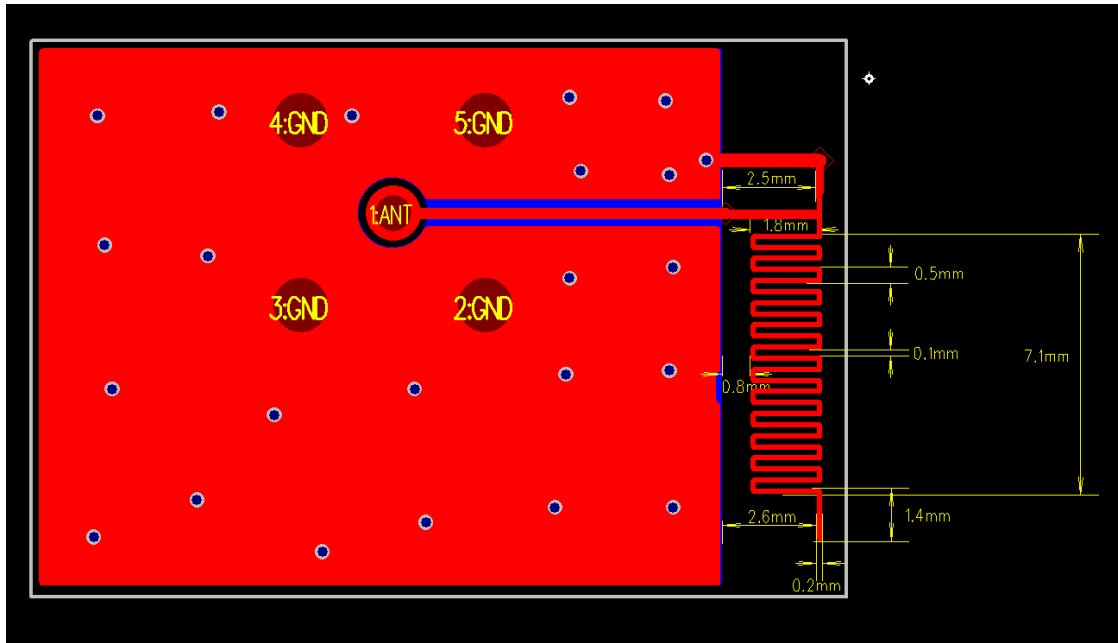


图 2.24 Mini_PIFA 天线的 PCB 尺寸

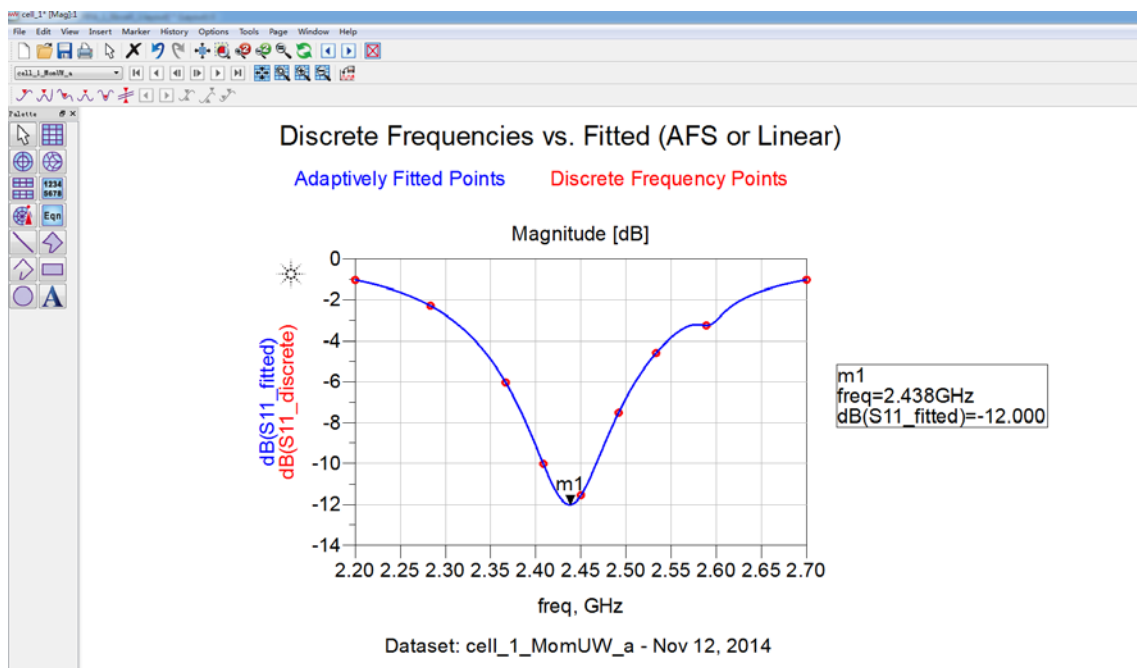


图 2.25 Mini_PIFA 天线的 S11 图

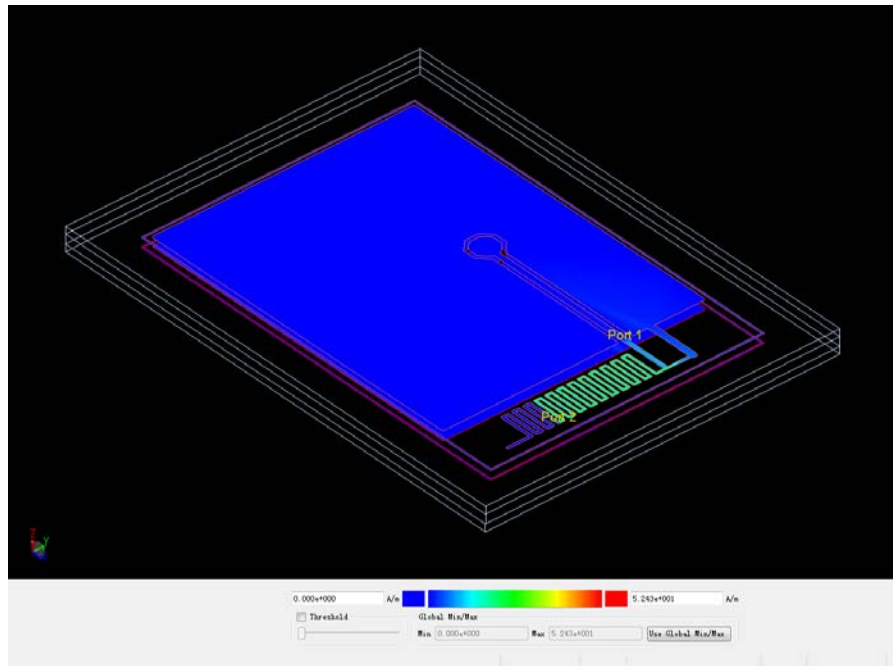


图 2.26 Mini_PIFA 天线的电流图

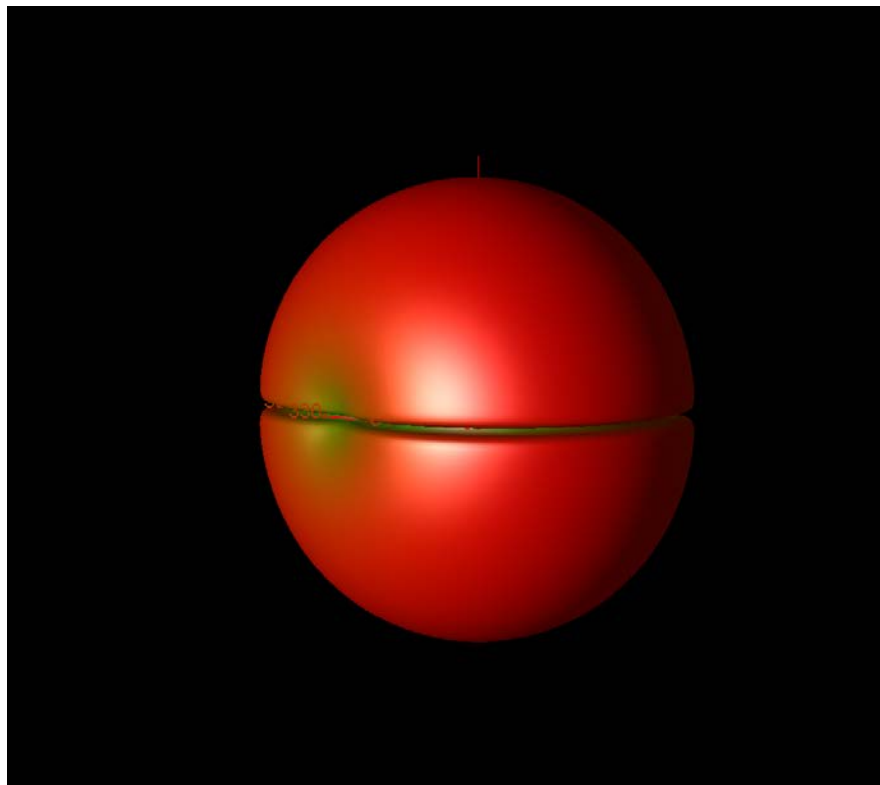


图 2.27 Mini_PIFA 天线的辐射图

2.3.6 天线设计总结

由于天线性能受板材（板厚、介质等）与周围环境（外壳、人体、天线离 PCB 地平面距离等）诸多因素的影响，所以天线需选择经过验证的结构。推荐的 PIFA（Printed Inverted FAntenna）天线和 Wiggle 天线都经过仿真和实测（考虑机壳等影响），实际使用时，可以根据板材介质等因素变化，进行微调。

天线 Layout 时需要注意:

- (1) 天线 top 部分及对应 bottom 层都需要净空, 不能铺铜;
- (2) PIFA 天线短路臂连到地以后, 至少有一个对地过孔直接到 bottom 层;
- (3) 天线与 PCB 铺铜地之间的距离应大于 0.5mm, 靠近天线沿着地平面应该打一排对地过孔。

3 HS6200 软件设计

3.1 基本描述

本节将介绍 HS6200 的基本特性以方便编程。包括 HS6200 的传输方式，地址要求，对外控制接口，中断形式，包结构等，开发者应仔细阅读本节内容。

3.1.1 SPI 接口

HS6200 和 MCU 的所有数据交互都是采用 4-wire SPI 通信，SPI 接口最大通讯速率为 10Mbps。

1. SPI 时序

SPI 操作和时序如图 3.1 和图 3.2 所示，通过 SPI 操作 HS6200 寄存器过程中，应将 CE 拉低。

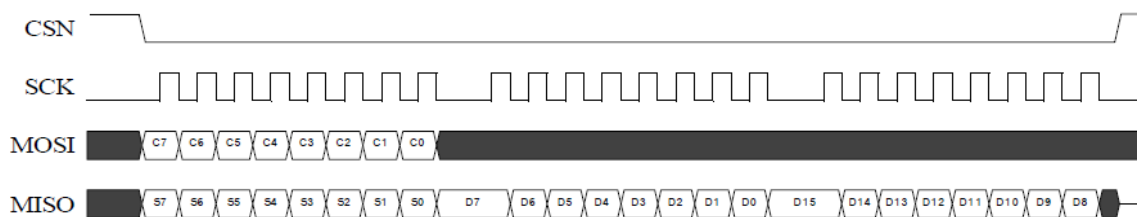


图 3.1 SPI 读操作

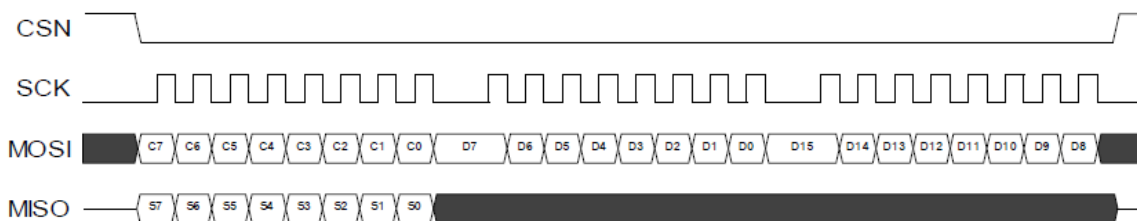


图 3.2 SPI 写操作

2. SPI 命令字

SPI 接口用到的命令表 4.1 有所说明。CSN 为低后 SPI 接口等待执行命令。每一条命令的执行都必须通过一次 CSN 由高到低的变化。

SPI 命令格式：

<命令字：由高位到低位（每字节）>

<数据字节：低字节到高字节，每一字节高位在前>

表 3.1 SPI 控制命令表

命令名称	命令字 (二进制)	数据字节	操作
R_RESISTER	000A AAAA	1 to 5 低字节在前	读命令寄存器和状态寄存器, AAAAA=5bit 寄存器地址
W_RESISTER	001A AAAA	1 to 5 低字节在前	写命令寄存器和状态寄存器, AAAAA=5bit 寄存器地址 仅在 power_down 和 standby 模式写可执行
R_RX_PAYLOAD	0110 0001	1 to 32 低字节在前	读接收数据, 读操作通常由第 0 字节开始。读完过后数据将被从 FIFO 中删除。接收模式可用
W_TX_PAYLOAD	1010 0000	1 to 32 低字节在前	写发射数据, 写操作通常由 0 字节开始。发射模式中可用
FLUSH_TX	1110 0001	0	清 Tx_FIFO, Tx 模式中可用
FLUSH_RX	1110 0010	0	清 Rx_FIFO, Rx 模式中可用。在回传应答时不可执行, 否则会使应答数据不完整
REUSE_TX_PL	1110 0011	0	用在 PTX 模式芯片, 重用最后传送的数据。当 CE 为高时数据将被一直重复使用直到 W_REGISTER, FLUSH_TX 命令执行后。数据传输时该命令不能被执行
R_RX_PL_WID	0110 0000	1	读 Rx_FIFO 最顶部 RX-payload 数据宽度
W_ACK_PAYLOAD	1010 1PPP	1 to 32 低字节在前	Rx 模式可用, 写 PIPE PPP (PPP 的值从 000 到 101) 响应 ACK 时同时回传的数据。最多可设置 3 个 ACK 数据包。同 PIPE 的数据将以先进先出的原则发送。写操作通常从 0 字节开始。
W_TX_PAYLOAD_NO_ACK	1011 0000	1 to 32 低字节在前	Tx 模式中可用, 使用该命令发送数据将使 AUTOACK 不可用
NOP	1111 1111	0	无操作。可用于读状态寄存器

R_REGISTER 和 W_REGISTER 寄存器可能操作单字节或多字节寄存器。当访问多字寄存器时首先要读

/写的是最低字节的高位。在所有多字节寄存器被写完之前可以结束写 SPI 操作，在这种情况下没有写完的高字节保持原有内容不变。例如：RX_ADDR_P0 寄存器的最低字节可以通过写一个字节给寄存器 RX_ADDR_P0 来改变。在 CSN 状态由高变低后可以通过 MISO 来读取状态寄存器的内容。

3.1.2 CE 控制

HS6200 的 CE 管脚为芯片 TX 或 RX 的使能，高有效。当 CE 置高，HS6200 开始 TX 或 RX 流程。当 CE 管脚内部结构简要描述如下图所示：

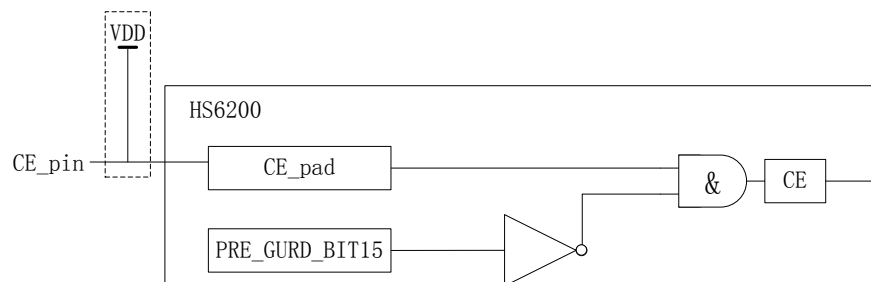


图 3.3 CE 管脚内部结构

可以看出 $CE = CE_PAD \& (\sim PRE_GURD_BIT15)$ 。

CE 管脚有两种外部接线形式，对应于两种控制形式。一种接法为 CE 管脚直接接到 MCU 的 GPIO 上，通过 GPIO 的拉高拉低来操作 CE。此种接法适合 MCU 的 GPIO 管脚充足的方案。另一种接法为 CE 直接到 VDD 如图 4.3 中虚线部分，通过 SPI 操作 HS6200 的寄存器 PRE_GURD 的 bit15 实现 CE 在 HS6200 内部的置高置低操作。此种接法适合 MCU 的 GPIO 管脚数量紧张的方案。相比于第一种接法，第二种接法节省了 MCU 的 IO 口，但通过寄存器控制 CE 会比直接用 IO 控制 CE 耗费更多操作时间。

用 GPIO 控制 CE 直接操作 IO 口即可，这里主要对第二种操作方式进行说明。PRE_GURD 寄存器位于 HS6200 的 BANK0，地址为 0x1F，因此在操作 PRE_GURD 寄存器控制 CE 的时候应确保 HS6200 已经处于 BANK0 模式。

当 CE 外部上拉到 VDD，由 CE 管脚内部结构可知，此时 CE_pad 将被置 1，CE 的最终状态将有 PRE_GURD 的 bit15 决定。即，当此位置 1 时，CE 置低，当此 bit 置 0 时，CE 置高。

CE_PIN 外挂到 VDD 的 CE 操作应按如下流程进行：

- 1、读 PRE_GURD 寄存器，数据长度为 2Bytes；
- 2、若执行 CE 置高，则将读出数据的 bit15 置为 0，反之，置 1；
- 3、将修改后的 2Bytes 数据写回 PRE_GURD 寄存器。

注意：1、PRE_GURD_BIT15 的值一旦更改成功，CE 操作则立即生效。2、整个 CE 操作的延时由 SPI 口速率决定。

3.1.3 IRQ 描述

HS6200 共有 3 种中断类型，分别是 TX_DS、RX_DR 和 MAX_RT。三种中断产生的条件见下表所示：

表 3.2 中断描述

中断类型	中断条件	中断清除
TX_DS	数据发送中断，TX 状态下，TX FIFO 中的数据发送完毕后将产生此中断， 特殊的，若使用 AUTO_ACK 自动应答模式，当发送端收到接收端返回的 ACK 方能产生此中断	向 BANK0 或 BANK1 寄存器 STATUS 的 TX_DS 位写 1
RX_DR	收据接收中断，RX 状态下，接收到数据包会产生此中断	向 BANK0 或 BANK1 寄存器 STATUS 的 RX_DR 的位写 1
MAX_RT	最大重传中断，TX 状态下，若使用 ART 自从重传模式，当所设的重传次数执行完毕仍没有收到接收端返回的 ACK，则会产生此中断。	向 BANK0 寄存器 STATUS 的 MAX_RT 位写 1， 同时执行 FLUSH_TX 指令清除 TX_FIFO，以防止当 CE 再次拉高，TX_FIFO 中有数据的情况下 HS6200 会继续发送进而继续产生 MAX_RT 中断。

当 HS6200 产生相应中断时，中断状态将被反映在 BANK0 寄存器 STATUS 相应位中，通过读取 STATUS 寄存器值，应用程序可以判断所发生的中断类型，进而进行相应的处理流程。

读 **STATUS** 应按如下步骤操作：向 SPI 写 NOP 指令，此时 SPI 返回的状态即为 HS6200 的 STATUS 值。

此外，HS6200 的中断状态也可通过 HS6200 的 IRQ_PIN 指示，以便 MCU 以外部中断的方式响应 HS6200 中断。当发生中断时，IRQ_PIN 电平将被拉低，中断状态被清除以后，IRQ_PIN 恢复高电平。不过，默认状态下，IRQ_PIN 中断指示不是能，用户需修改 BANK0 的 CONFIG 寄存器中 MASK_RX_DR、MASK_TX_DS 或 MASK_MAX_RT 来使能相应中断的 IRQ_PIN 指示。

3.1.4 BANK 切换

HS6200 共有两个 BANK，BANK0 和 BANK1，BANK0 和 BANK1 下的寄存器公用相同的地址空间，通过 SPI 操作 HS6200 时需要确保对应的 BANK 正确，BANK 不对应时要切换 BANK 后再进行操作。

HS6200 的 BANK 切换时通过 SPI 写寄存器实现的，通过 SPI 向 HS6200 写 0x50,0x53 将实现 BANK 切换。当前 BANK 状态应用程序可以通过读取 BANK0 或 BANK1 的寄存器 STATUS 获得，STATUS 的 bit7 为 1 时，说明 SPI 的读写操作都将针对 BANK1 的寄存器进行，STATUS 的 bit7 为 0 时，说明 SPI 的读写操作都将针对 BANK0 的寄存器进行。

切换 **BANK** 的操作应按如下流程进行：

读 STATUS 寄存器，数据长度 1 个 Byte，通过读出数据的 bit7 判断是否需要切换 BANK，若需要，则

向 SPI 写 0x50,0x53, 若不需要切换 BANK, 则不操作。

3.1.5 静态和动态数据包长功能

通信协议提供了两种数据长度：静态和动态。

默认为静态数据长度，在此模式下，用户通过设置 BANK0 的寄存器 RX_PW_Px 的 RX_PW_Px 域配置接收端的数据长度，接收机根据 RX_PW_Px 的配置值长度接收数据。发射机计算 TX_FIFO 锁定的数据长度发射。应注意，发射数据长度应和接收机设置的数据包长度一致。

静态数据长度(DPL)模式下，发射机发送给接收机的数据长度可变，接收机可以自动解码收到数据包的有效数据长度而不再依赖 RX_PW_Px 寄存器的值。这意味着发射端每次发射可以使用不同的包长度。

数据包动态长度功能配置：1、设置 FEATURE 寄存器中的 EN_DPL 位，使能 DPL 功能。2、配置 BANK0 的 DYNPD 寄存器相应 bit 使能相应管道 DPL。注：PTX 端的 BANK0 的 DYNPD 寄存器 DPL_P0 必须设置为 1，PRX 可根据需要设定。

3.1.6 自动回复 ACK 功能(auto ACK)

Auto ACK 功能为接收端收到数据包并判断数据包无误后自动向发送端回复 ACK 的功能。默认条件下，接收端回复的 ACK 为空包，特殊地，当 PRX 收到的数据包 Packet control field 域的 NO_ACK 位为 1 时，PRX 将不回复 ACK。

Auto ACK 功能配置：配置 BANK0 的寄存器 EN_AA 相应 bit 可开启相应管道的此功能。

PRX 自动回复的 ACK 也可以携带 payload, ACK payload 需使用 W_ACK_PAYLOAD 指令写入 PRX 的 TX_FIFO 中，将按照先进先出的原则被使用。接收端的 TX_FIFO 最多可同时保持 3 个 ACK payload，如下图所示：

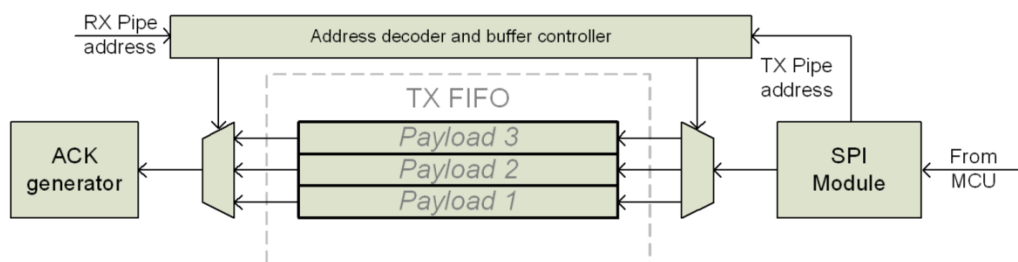


图 3.4 Auto Ack TX_FIFO

ACK with payload 配置：1、设置 FEATURE 寄存器中的 EN_ACK_PAY 位，使能 ACK with payload 功能。2、将 ACK payload 内容使用 W_TX_PAYLOAD 指令写入 PRX 的 TX_FIFO 中。

3.1.7 自动重传功能（ART）

自动重传是指在 **Auto ACK** 模式下，发射机在发送数据包后会转接收，然后接收 **ACK**，若在一定时间内发射机未收到 **ACK** 包，会再次自动发上一次数据包的操作，直到发射次数超出重发设置。此操作不需要 MCU 参与，可节省 MCU 资源。

自动重发配置：在 **Auto_ACK** 模式下，通过 **BANK0** 的寄存器 **SETUP_RETR** 设置。注意：**SETUP_RETR** 设置的次数是在正常发送一次后重传的次数。比如 **SETUP_RETR** 设置为 10，实际上发出去的最大数据包数为 11 次。

当发射机发射次数达到 **SETUP_RETR** 设置值，即发生最大重传中断 **MAX_RT**。若在发生 **MAX_RT** 中断前发射机收到了接收机发来的 **ACK**，则重发流程会终止并产生 **TX_DS** 中断。实际应用中，为保证所设重发次数合理，强烈建议单独运行 **TX**，查看是否可产生 **MAX_RT**(最大重传)中断。

自动重传时的空口时隙如下图所示：

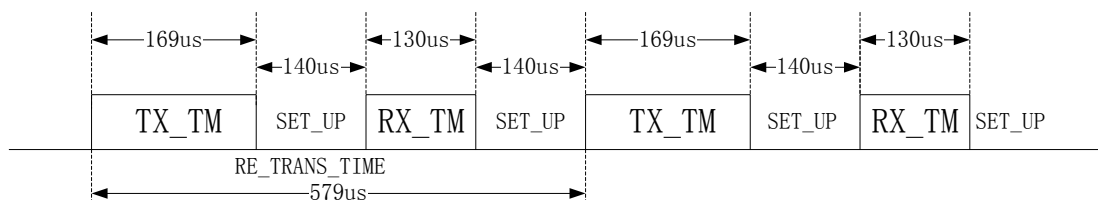


图 3.5 ART 时隙描述

其中，

TX_TM：为空口发数据一次时间，HS6200 包结构见章节 4.1.8 描述。

以 1Mbps，5byte 地址，10 byte payload，2byte CRC 为例，HS6200 发送一包数据时间为：

$$(8+40+16+9+80+16)\text{bit} \times 1\text{us} = 169\text{us}$$

SET_UP：收发建立时间，140us。

RX_TM：HS6200 发包后接收 **ACK** 的时间。

3.1.8 packet 描述

应用程序使用 **W_TX_PAYLOAD** 指令将待发送数据写入 **TX_FIFO** 后，HS6200 将按照 **BANK0** 中的相应配置组包并进行空口数据发送，一个典型的 HS6200 空口数据包如下图所示，其中参数可通过 **BANK0** 相应寄存器修改：

表 3.3 HS6200 空口 packet 描述

Preamble	Address	Guard	Packet control	Payload	CRC
1 byte	4-5 byte	2byte	field 9 bit	0-32 bytes	1-2 bytes

Preamble: 前导码，接收方同步使用，长度是 1 字节，此部分不需配置，HS6200 会根据地址的第一位是 1 还是 0 决定 preamble 的值是 10101010（二进制）还是 01010101（二进制）。

Address: 地址，长度可通过 BANK0 的 SETUP_AW 寄存器配置，可配置为 4 或 5 个字节，内容可由 BANK0 的 RX_ADDR_P0 和 TX_ADDR 寄存器配置，收发双方配置地址应相同，接收方收到数据包会匹配地址，发现地址不一致时，会放弃接收，因此地址的使用能够防止同频点不同地址多个系统间的交扰。

HS6200 所使用的地址不可随意选择，应符合以下原则：

- 地址中不能有超过六个的连续的“零”或者“一”；
- 地址字节不能都一样；
- 不能有超过 2 4 个“0”和“1”的转换；

Guard: 长度 2 字节，主要作用是为了接收机更好的同步。

Packet control field: 包控制域，有 9 个比特，包括 6 比特的 payload length(数据长度)、2 比特的 PID（数据包识别）和一个比特的 NO_ACK 标志，具体如表 3.4 所示。

表 3.4 HS6200 Packet control field 描述

Payload length	PID	NO_ACK
6bit	2bit	1bit

6 比特的 Payload length（数据长度）标识数据的字节数。数据字节数为 0~32 字节。000000 表示 0 字节（只用在空 ACK 数据包），数据长度为 0 可以用 R_RX_PAYLOAD 读出无数据。100000 表示数据长度为 32 字节。100001 未被使用。此功能只有在动态数据长度下才被使能。

2 比特的 PID 号用于检测接收的数据包是新的还是重传的。PID 防止 PRX 送给 MCU 相同的数据包。在 PTX 端，每发一个新的数据包，PID 就自动加 1。当发射的数据包丢失时，PID 会保持上一个数据包的 PID 号。

NO_ACK 位在 选择自动应答功能（EN_DYN_ACK 位使能）下使用。在发射机端，通过 W_TX_PAYLOAD_NOACK 命令设置 NO_ACK 位，设置为“1”时，告知接收机不需要回自动应答。

Payload: 是用户需要发送的内容，由应用程序用 W_TX_PAYLOAD 或 W_TX_PAYLOAD_NOACK 指令将待发送数据写入 TX_FIFO。此部分长度可以是 0-32 字节，用户至少需发送 1 字节数据，在 ART 模式下，当接收方应答是空包时，此部分长度为 0。

数据是用户定义的数据包内容，长度为 0~32 字节。

CRC: CRC 是数据包的错误检测机制，它可以是 1 或 2 字节，计算地址、数据标识和数据。CRC 字节数可通过设置 BANK0 的 CONFIG 寄存器的 CRCO 位来选择。如 CRC 校验错误，则数据包不被接收。

1 字节 CRC 多项式: $X^8 + X^2 + X + 1$ ，初始值为 0xFF。

2 字节 CRC 多项式: $X^{16} + X^{12} + X^5 + 1$ ，初始值为 0xFFFF。

如果收到数据包 PID 号一样，PRX 进行 CRC 校验，如果 CRC 校验也相同，此数据包为重传数据包，将被丢掉。

3.1.9 常用操作时间计算

1. CE 操作时间:

若通过 GPIO 控制 CE，此部分时间可以忽略，若通过寄存器控制 CE，则需要 6 个 SPI byte 写时间，包括 3Byte 时间将 HS6200_BANK0_PRE_GURD 寄存器内容读出(1 个 Byte 地址 2 个 Byte 数据)，3Byte 时间将 HS6200_BANK0_PRE_GURD 寄存器内容写入(1 个 Byte 地址 2 个 Byte 数据)。

以 200KHz 的 SPI 计算，寄存器操作一次 CE 的时间为 240us。

2. 切换信道操作时间:

2 个 SPI byte 写时间，包括 1byte 的地址，1byte 的数据。

以 200KHz 的 SPI 计算，更改一次信道所需时间为 80us。

3. 清中断操作时间:

2 个 SPI byte 写时间，包括 1byte 的地址，1byte 的数据。

以 200KHz 的 SPI 计算，更改一次信道所需时间为 80us。

4. 读 STATUS 寄存器时间:

2 个 SPI byte 写时间，包括 1byte 的地址，1byte 的数据。

以 200KHz 的 SPI 计算，更改一次信道所需时间为 80us。

5. 写 payload 时间:

以 200KHz 的 SPI 计算，写 10byte payload 需要 400us。

3.2 工作流程

HS6200 工作分为 2 个步骤：1、初始化；2、收发数据。工作流程如下图所示：

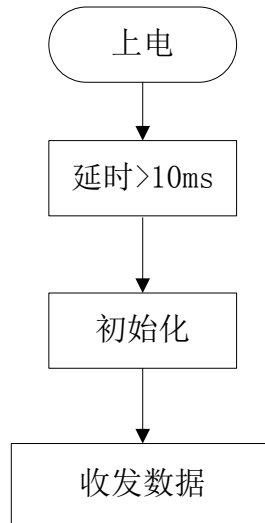


图 3.6 HS6200 工作流程

校准与初始化需先后进行，顺序不可颠倒，初始化之后方能进行正常的收发操作。注意：HS6200 上电后需要 10ms 的延时用来完成 power on reset，之后才能通过 SPI 对 HS6200 进行操作。

3.2.1 HS6200 初始化流程及配置

HS6200 上电power on reset后就可以开始校准流程，校准流程及配置如表 4.6 所示，其中切换BANK、CE 操作、中断操作的方法请参照 4.1 节相关内容。

表 4.6 所列数据为一组 HS6200 的典型配置，配置后 HS6200 的工作方式为：发射速率 1Mbps，Auto ACK，ART，5dBm 发射功率，如需其他配置详见 DataSheet 相关说明。

初始化操作请严格按照表格所述顺序进行。

表 3.5 HS6200 初始化与配置

1、切换至 BANK0		
2、CE 拉低		
3、配置 BANK0 寄存器 (配置如下)		
寄存器地址	配置值	备注
0x00	0x03	HS6200_BANK0_CONFIG
0x05	0x32	HS6200_BANK0_RF_CH
0x06	0x40	HS6200_BANK0_RF_SETUP
4、切换至 BANK1		
5、配置 BANK1 寄存器 (配置如下)		
寄存器地址	配置值	备注

0x01	1M:0x40,0x01,0x10,0xE5 2M: 0x40,0x01,0x10,0xE6	HS6200_BANK1_PLL_CTL0
0x03	0x20,0x08,0x50,0x40,0x50	HS6200_BANK1_CAL_CTL
0x0A	1M:0x00,0x00,0x1F 2M: 0x00,0x00,0x1E	HS6200_BANK1_IF_FREQ
0x0C	1M:0x25 2M:0x30	HS6200_BANK1_FDEV
0x18	0x7F	HS6200_BANK1_DAC_CAL_HI
0x1E	0x1F,0x64,0x00,0x81	HS6200_BANK1_RF_IVGEN
6、切换至 BANK0		
7、延时>=2ms(等待晶体起振)		
8、CE 拉高		
9、20us<延时<= 2ms(CE pulse，以使 HS6200 开始校准。)		
10、CE 拉低		
11、延时>=40ms(CE pulse 后大约 40ms 后，校准完成)		
12、CE 拉低(需确保此时 HS6200 处于 BANK0)		
13、切换至 BANK1		
14、配置 B ANK1 寄存器(配置如下)		
寄存器地址	配置值	备注
0x01	1M:0x40,0x01,0x30,0xE1 2M: 0x40,0x01,0x30,0xE2	HS6200_BANK1_PLL_CTL0
0x02	0x00,0x42,0x10,0x07	HS6200_BANK1_PLL_CTL1
0x03	0x29,0x89,0x75,0x28,0x50	HS6200_BANK1_CAL_CTL
0x11	0x52,0xC2,0x09,0xAC	HS6200_BANK1_RX_CTRL
0x13	0x80,0x14,0x08,0x29	HS6200_BANK1_FAGC_CTRL_1
0x1E	0x1F,0x64,0x00,0x11	HS6200_BANK1_RF_IVGEN
15、切换至 BANK0		
16、配置 BANK0 寄存器(配置如下)		
寄存器地址	配置值	备注
0x00	TX:0x0F RX: 0x0E	HS6200_BANK0_CONFIG
0x11	0x0A	HS6200_BANK0_RX_PW_P0
0x1c	0x3F	HS6200_BANK0_DYNPD
0x1D	0x07	HS6200_BANK0_FEATURE
0x1E	0x40,0x5A,0x80,0x06,0x00	HS6200_BANK0_SETUP_VALUE
0x1F	0x77	HS6200_BANK0_PRE_GURD
0x01	0x01	HS6200_BANK0_EN_AA
0x02	0x01	HS6200_BANK0_EN_RXADDR
0x03	0x03	HS6200_BANK0_SETUP_AW
0x04	0x0F	HS6200_BANK0_SETUP_RETR
0x05	0x50	HS6200_BANK0_RF_CH
0x06	1M,8dBm:0x47 2M,8dBm: 0x4F	HS6200_BANK0_RF_SETUP
0x0A	0x55,0x42,0x9C,0x8F,0xC9	HS6200_BANK0_RX_ADDR_P0

0x10	0x55,0x42,0x9C,0x8F,0xC9	HS6200_BANK0_TX_ADDR
17、清 STATUS 寄存器所有中断标志		
18、Flush_TX		
19、Flush_RX		
20、CE 拉高		

3.2.2 HS6200 收发数据流程

本节介绍 HS6200 收发数据的流程,用户可根据本节介绍及实际使用情况设计应用程序。其中 CE 操作、中断操作的方法请参照 3.1 节相关内容。

3.2.2.1 发送数据

HS6200 推荐的数据发送步骤如下:

- 1、拉低 CE。
- 2、使用 SPI FLUSH_TX 指令清 TX_FIFO。
- 4、将数据写入 TX_FIFO。
- 5、CE Pulse(CE 拉高再拉低,中间间隔要大于 20uS)。
- 6、查询 STATUS 状态,直到 HS6200 中断产生。
- 7、写 STATUS 清中断标志位。
- 8、结束。

以上步骤补充说明如下:

步骤 4,当使用的是 Auto ACK 模式,若希望 PRX 回 ACK,则使用 W_TX_PAYLOAD 指令写入 TX_FIFO,若不需 PRX 回 ACK,则使用 W_TX_PAYLOAD_NO_ACK 指令写入 TX_FIFO。

步骤 6,若使用的不是 ART 方式,此处等来的中断将会是 RX_DR。若 ART 方式,则此处可能等来的中断为 TX_DS 或 MAX_RT,即收到 ACK 产生 TX_DS,超时没收到 ACK 产生 MAX_RT。当使用 HS6200 IRQ_pin 指示中断时,可查询 IRQ_pin 状态,中断产生后再读 STATUS 寄存器。

HS6200 接收数据流程图如下:

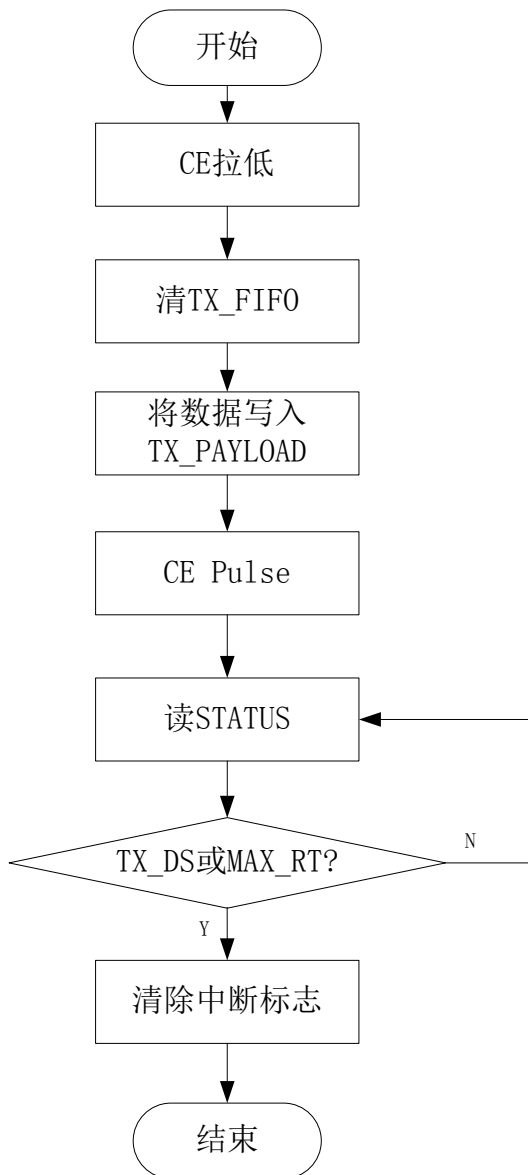


图 3.7 HS6200 发送数据流程

3.2.2.2 接收数据

HS6200 推荐的数据接收步骤如下：

- 1、查询 STATUS 寄存器，判断是否产生 RX_DR 中断。
- 2、若产生 RX_DR 使用 R_RX_PL_WID 命令读 RX_FIFO 中数据长度。
- 3、数据长 length 不为 0 则使用 R_RX_PAYLOAD 命令将数据读出。
- 4、重复步骤 3，直到读出的 length 为 0。
- 5、使用 SPI_FLUSH_RX 指令清 RX_FIFO。
- 6、写 STATUS 清中断标志位。

7、处理接收数据。

8、结束。

以上步骤补充说明如下：

步骤 1，当使用 HS6200 IRQ_pin 指示中断时，可查询 IRQ_pin 状态，中断产生后再读 STATUS 寄存器。

具体流程如下图所示：

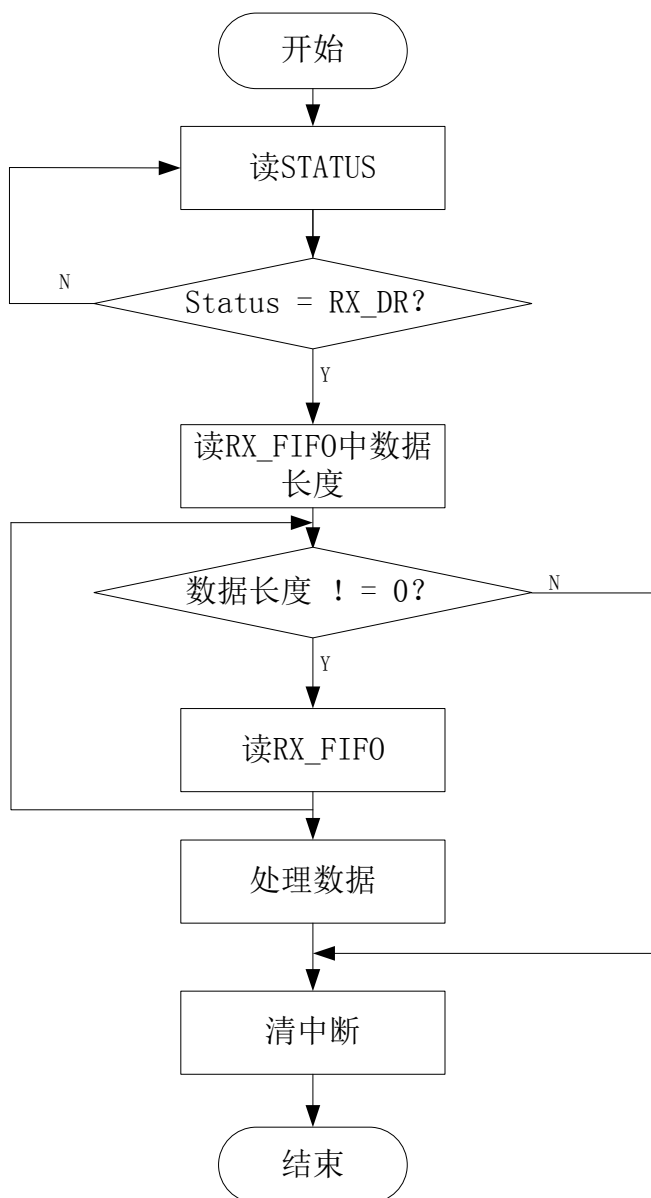


图 3.8 HS6200 接收数据流程

3.3 驱动程序

3.3.1 介绍

- 1、本驱动由 c 语言编写，汇编语言可参照修改，驱动中所用流程及配置请勿轻易修改。
- 2、本驱动软件驱动包括 HS6200.c 和 HS6200.h 两个文件，文件位置为本 SDK 文件目录..\SW\src\HS6200_driver\HS6200，在移植驱动前，请确定所使用的 HS6200 芯片版本，及 HS6200 外部的器件是否严格按照硬件参考设计实现。
- 3、本驱动在 Keil C51 uVision4 环境下编译调试通过。

3.3.2 函数表

表 3.6 HS6200 驱动函数表

函数名称	函数功能
HS6200_spi_wrd	SPI 驱动
HS6200_write_byte	通过 SPI 向目标地址写一个 byte
HS6200_read_byte	通过 SPI 从目标地址读一个 byte
HS6200_wr_buffer	通过 SPI 向目标地址连续写多个 byte
HS6200_read_buffer	通过 SPI 从目标地址连续读多个 byte
HS6200_wr_cmd	通过 SPI 向 6200 写 CMD
HS6200_Operation	HS6200 执行操作
HS6200_bank_Switch	激活 HS6200 工作寄存器 BANK0 或 BANK1
HS6200_CE_High	使能 HS6200 收发
HS6200_CE_Low	不使能 HS6200 收发
HS6200_Read_Status	读 HS6200 状态寄存器
HS6200_ChangeAddr_Reg	更改 HS6200 地址，包括发射和接收地址
HS6200_Change_CH	更改 HS6200 信道
HS6200_Get_Chip_ID	读芯片版本号
HS6200_ModeSwitch	切换 HS6200 工作模式，可切换至 TX,RX 或者载波模式
HS6200_SendPack	发送数据包
HS6200_RecivePack	接收数据包
HS6200_Write_Ack_Payload	写 auto_ACK 模式下写 ACK 到 TX_FIFO
HS6200_Clear_All_Irq	清除所有中断
HS6200_Flush_Tx	清除 TX_FIFO
HS6200_Flush_Rx	清除 RX_FIFO
HS6200_Init	初始化 HS6200，包括校准及初始化两部分

3.3.3 函数描述

1、 unsigned char HS6200_spi_wrd(unsigned char Data)

描述: SPI 驱动, 向 HS6200 写一个 Byte 并返回一个 Byte

参数: Data: MOSI 输出

返回值: MISO 输入

2、 void HS6200_write_byte(unsigned char addr,unsigned char D)

描述: 通过 SPI 向目标地址写一个 byte

参数: addr: 寄存器地址

D: 待写入数据

返回值: 无

3、 unsigned char HS6200_read_byte(unsigned char addr)

描述: 通过 SPI 从目标地址读一个 byte

参数: addr: 寄存器地址

返回值: 读出的数据

4、 void HS6200_wr_buffer(unsigned char addr,unsigned char* buf,unsigned char len)

描述: 通过 SPI 向目标地址连续写多个 byte

参数: addr: 寄存器地址

buf: 待写入数据首地址

len: 写入数据长度

返回值: 无

5、 void HS6200_read_buffer(unsigned char addr,unsigned char* buf,unsigned char len)

描述: 通过 SPI 从目标地址连续读多个 byte

参数: addr: 寄存器地址

buf: 读出数据的存放地址

len: 读出数据长度

返回值：无

6、void HS6200_wr_cmd(unsigned char cmd,unsigned char D)

描述：通过 SPI 向 6200 写 CMD

参数：cmd: 指令字节 1

D: 指令字节 2

返回值：无

7、unsigned char HS6200_Operation(unsigned char opt)

描述：HS6200 执行操作

参数：opt:

HS6200_READ_STATUS: 读 STATUS 状态

HS6200_FLUSH_TX: 清 TX_FIFO

HS6200_FLUSH_RX: 清 RX_FIFO

返回值：STATUS 状态

8、void HS6200_bank_Switch(HS6200_Bank_TypeDef bank)

描述：激活 HS6200 工作寄存器 BANK0 或 BANK1

参数：bank:

HS6200_Bank0: 切换至 BANK0

HS6200_Bank1: 切换至 BANK1

返回值：无

9、void HS6200_CE_High(void)

描述：HS6200 内部 CE 置高，使能 HS6200 收发

参数：无

返回值：无

10、void HS6200_CE_Low(void)

描述：HS6200 内部 CE 置低，不使能 HS6200 收发

参数：无

返回值： 无

11、 unsigned char HS6200_Read_Status(void)

描述：读 HS6200 状态寄存器 STATUS

参数：无

返回值： STATUS 状态

12、 void HS6200_ChangeAddr_Reg(unsigned char* AddrBuf,unsigned char len)

描述：更改 HS6200 地址，包括发射和接收地址

参数： AddrBuf：保存待写入地址 buffer 的首地址

len：地址长度

返回值：无

13、 void HS6200_Change_CH(unsigned char ch_index)

描述：更改 HS6200 地址，包括发射和接收地址

参数： AddrBuf：保存待写入地址 buffer 的首地址

len：地址长度

返回值：无

14、 unsigned char HS6200_Get_Chip_ID(void)

描述：读芯片版本号

参数：无

返回值： HS6200 批次号

15、 void HS6200_ModeSwitch(HS6200_ModeTypeDef mod)

描述：切换 HS6200 工作模式，可切换至 TX,RX 或者载波模式

参数： mod：

Rf_PRX_Mode：接收模式

Rf_PTX_Mode：发射模式

Rf_Carrier_Mode：载波模式

返回值：无

16、void HS6200_SendPack(unsigned char cmd, unsigned char* buf, unsigned char len)

描述：切换 HS6200 工作模式，可切换至 TX,RX 或者载波模式

参数：mod:

Rf_PRX_Mode: 接收模式

Rf_PTX_Mode: 发射模式

Rf_Carrier_Mode: 载波模式

返回值：无

17、void HS6200_SendPack(unsigned char cmd, unsigned char* buf, unsigned char len)

描述：发送数据包

参数：buf: 存放待发送数据 buffer 的首地址

Len: 发送数据长度

返回值：无

18、unsigned char HS6200_RecivePack(unsigned char* buf)

描述：接收数据包

参数：buf: 保存接收数据包 buffer 的首地址

返回值：无

19、void HS6200_Write_Ack_Payload(unsigned char PipeNum, unsigned char *pBuf, unsigned char bytes)

描述：写 auto_ACK 模式下写 ACK 到 TX_FIFO

参数：PipeNum: 管道号

Buf: 存储 ACK 内容的 buffer 首地址

Bytes: 数据长度

返回值：无

20、void HS6200_Flush_Tx(void)

描述：清除 TX_FIFO

参数：无

返回值：无

21、void HS6200_Flush_Rx(void)

描述：清除 RX_FIFO

参数：无

返回值：无

22、void HS6200_Init(void)

描述：初始化 HS6200，包括校准及初始化两部分

参数：无

返回值：无

3.3.4 驱动移植

1、 将 HS6200.c 和 HS6200.h 两个文件包含到客户的工程中，并在需要调用 HS6200 接口的文件中添加头文件引用说明 `#include "HS6200.h"`。

2、 CE 控制修改：

HS6200 的 CE 控制通过调用接口 HS6200_CE_High()和 HS6200_CE_Low()来实现。

若：客户方案使用 GPIO 作为 CE 管脚的控制管脚，则打开头文件 HS6200.h 中的宏 `__HS6200_USE_CE_CTRL`，并将 CE 高和 CE 低的操作(HS6200_CE_HIGH、HS6200_CE_LOW)定义为客户所使用的相应管脚。

若：客户方案中没有多余 GPIO 作为 CE 管脚的控制，则首先应确保客户已将 HS6200 的 CE 管连接到高电平，并一直保持拉高，然后关闭头文件 HS6200.h 中的宏 `__HS6200_USE_CE_CTRL`。

3、 SPI 驱动移植：

首先，配置 SPI 的 CSN 为客户方案中的 CSN 控制，即修改 HS6200.h 中

```
#define HS6200_CSN_HIGH
```

```
#define HS6200_CSN_LOW
```

为客户方案中相应操作。然后，修改 HS6200.c 中单 byte 写入操作接口 HS6200_spi_wrd()内容为客户方案中单 byte 写入操作。

4、 修改 HS6200.c 中 CLK_SysTickDelay()为客户方案中相应延时函数。

5、 替换客户方案中调用 transceiver 的函数接口为 HS6200 接口，如初始化，查状态，查接收 FIFO，写发送 FIFO 等。一般情况下，客户原使用的接口我们的新驱动中都会包含，特殊情况下所需驱动不全的，

可联系研发人员为客户添加功能函数接口。

6、注意：由于客户的某些变量定义在了原驱动文件中，在使用 HS6200 驱动替换原驱动的时候，请保留原驱动中的这些定义。

3.4 跳频方式

跳频方式多种多样，本部分主要介绍一种主动跳频和一种被动跳频方法来说明 HS6200 的跳频工作方式。实际使用中的跳频方式可根据具体应用选择及修改，并不限于本章节所讲述的方法。本章节介绍的两种跳频方法实现可参考本 SDK 文件目录下：

..\SW\src\HS6200 Demo\ HS_6200_demo_1(主动跳频)

..\SW\src\HS6200 Demo\ HS_6200_demo_2(被动跳频)

3.4.1 频率表

PRX 和 PTX 需要按照预先定义的频点序列进行跳频，这个频点序列就是频率表。HS6200 建议使用的频点推荐如表 4.5 所示，如所需频率表频点少于 12 个，可顺序选择一些频点使用，若所需频点大于 12 个，当以连续两个频点间隔较远及尽量避开 WIFI 常用频点为原则。

表 3.7 频率表

Index	Channel	Frequency(MHz)
0	2	2402
1	34	2434
2	61	2461
3	8	2408
4	40	2440
5	65	2465
6	13	2413
7	46	2446
8	73	2473
9	25	2425
10	56	2456
11	78	2478

3.4.2 被动跳频方式

被动跳频是一种当链路丢失，收发双方才会按照约定频率表及规则跳频的一种通信方式。相对主动跳频，这种方式较为简单。

被动跳频方法描述如下：

Rev: 1.4 /16-Mar.-2016

46 / 60

TX: 每 TX_T(10ms)查收一次 ACK, 若收到 ACK, 则继续发送一包数据, 若没收到 ACK, 则跳频后发送一包数据。

RX: 每 RX_T(10ms)查收一次数据, 若收到数据, 则信道不变, 若收不到数据, 则跳频, 同时, RX_F 加 1。当 RX_F 等于 3, 则此后 RX 每隔 RX_T_SCAN(30ms)查收一次数据, 若收到数据, 则信道不变, RX_F 清零, RX 再次以 RX_T(10ms)为间隔查收数据, 以后将重复上述流程。若收不到数据, 则跳频, 直到 RX 再次收到数据。

本方法中, TX_T, RX_T 可根据实际需要设置, 原则上,

TX_T=RX_T, TX_T<RX_T_OUT 以使 TX 和 RX 在是去链路有限时间内重新回到同一频点上。

正常通信过程中, TX 和 RX 均不跳频, 如下图所示:

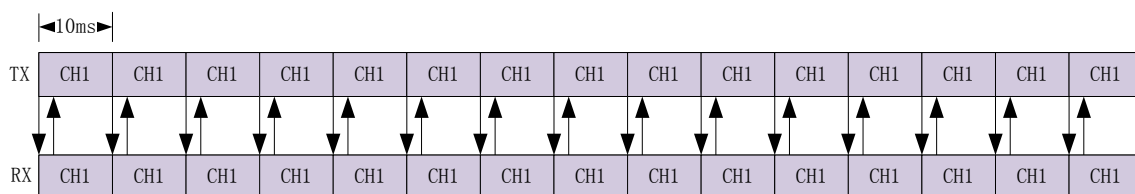


图 3.9 被动调频通信正常时隙图

当通信中有数据包丢失, RX_F 小于 3, 则 TX 以 TX_T 时间跳频, RX 以 RX_T 时间跳频, 如下图所示:

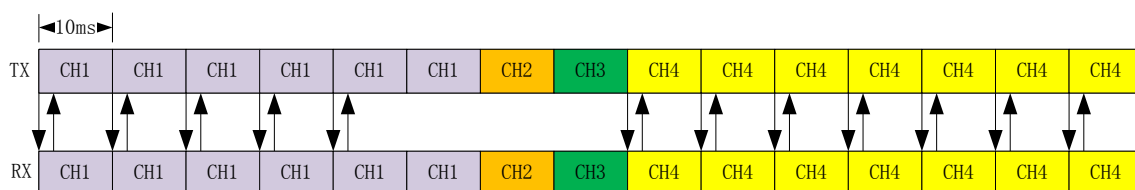


图 3.10 被动调频通信中断时隙图

当 RX_F 大于等于 3, 即链路丢失, 则 TX 以 TX_T 时间跳频, RX 以 RX_T_SCAN 时间跳频, 如下图所示:

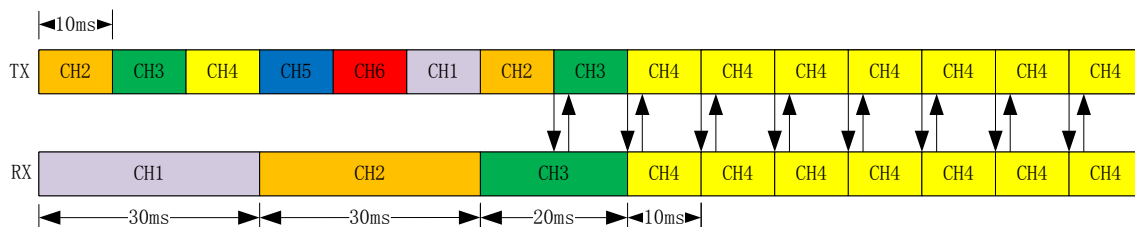


图 3.11 被动调频通信失联时隙图

3.4.3 主动跳频方式

主动跳频是一种无论链路是否丢失，收发双方都会按照约定频率表及规则跳频的一种通信方式。相对于被动跳频，主动跳频实现较为复杂，但由于主动跳频 TX 不需关心 RX 所回 ACK，因此，理论上主动跳频将会提供更远的通信距离。本节介绍的主动跳频方式为 TX 发送带有序号的数据包，RX 通过收到的数据包中的序号判断是否跳频以及多长时间后跳频。

主动跳频方法描述如下：

TX 发送的数据包长度为 10bytes， byte9 为跳频序号(M0、M1、M2)，数据包结构如下：

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8	Byte9
用户数据	用户数据	用户数据	用户数据	用户数据	用户数据	用户数据	用户数据	用户数据	M0-M2

TX：每 TX_T(10ms)发一包数据，TX_HP_T(30ms)一个周期(M1,M2,M3 各发一次)，每个周期跳频一次。

RX：每 RX_T(10ms)查收一次数据，当收到 M1 消息，RX_HP_T1(25ms)后跳频，当收到 M2 消息，RX_HP_T2(15ms)后跳频，当收到 M3 消息，RX_HP_T3(5ms)后跳频。若没收到数据，则认为数据中断，RX 以 RX_HP_T_F(30ms)跳频一次。若超过 RX_HP_T_OUT(200ms)没收到数据，则认为链路中断，RX 以 RX_HP_T_SCAN(10ms)跳频一次(此时发射端仍以 30ms 为周期跳频)。

本方法中所述的 TX_T，TX_HP_T，RX_T 可以根据实际需要修改，相应的 RX_HP_T1，RX_HP_T2，RX_HP_T3，RX_HP_T_F,RX_HP_T_OUT 也需做相应修改，原则上，

$$RX_T=TX_T$$

$$RX_HP_T1=TX_T/2$$

$$RX_HP_T2=TX_T/2+TX_T$$

$$RX_HP_T3=TX_T/2+TX_T*2$$

$$RX_HP_T_F=TX_HP_T$$

$$RX_HP_T_OUT=通信终端可容忍时长$$

$$RX_HP_T_SCAN=TX_T$$

正常通信过程中，TX 和 RX 跳频如下图所示：

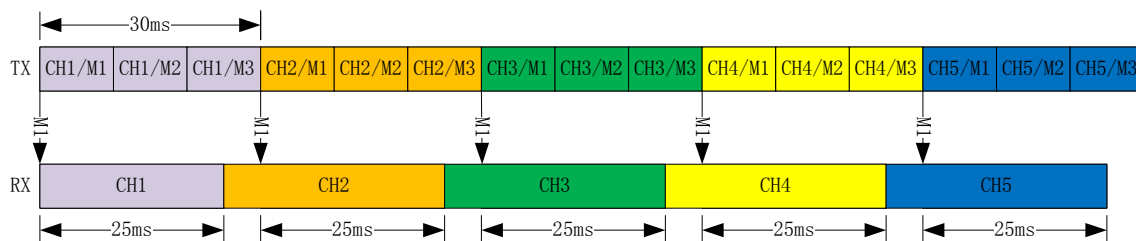


图 3.12 主动调频通信正常时隙图

当通信中有数据包丢失，TX 和 RX 跳频如下图所示：

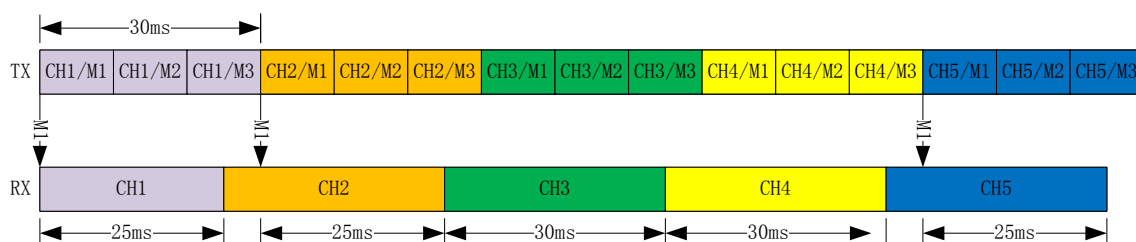


图 3.13 主动调频通信中断时隙图

当通信中链路丢失，TX 和 RX 跳频如下图所示：

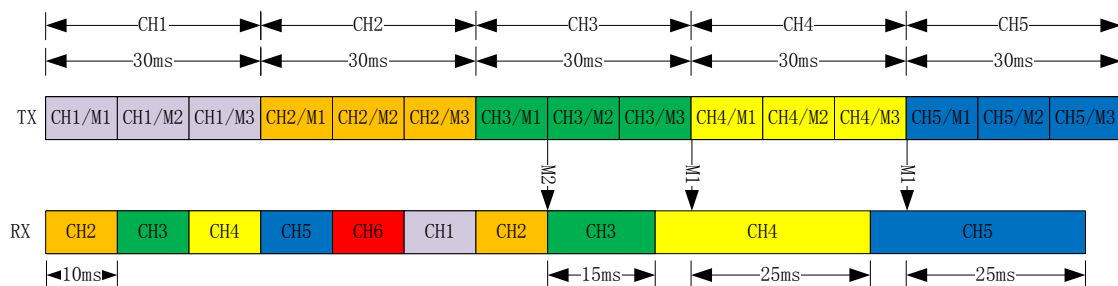


图 3.14 主动调频通信失联时隙图

4 HS6200 Module 射频测试

HS6200 Module 的射频性能测试主要包括发射功率（Power）频率（Frequency）和接收灵敏度（Sensitivity）测试，而 FCC/CE 测试主要 FAIL 在谐波功率高于标准要求，本部分将详细介绍测试方法和模块测试结果。

4.1 测试准备

一般实验室射频测试都为传导(Conducted) 测试，即通过射频线 SMA 头连到外部仪器测试，对于本 Module 需要焊接 R5(0ohm)，并且与 PIFA 天线断开(去掉 R6)下图为传导测试的焊接图片：

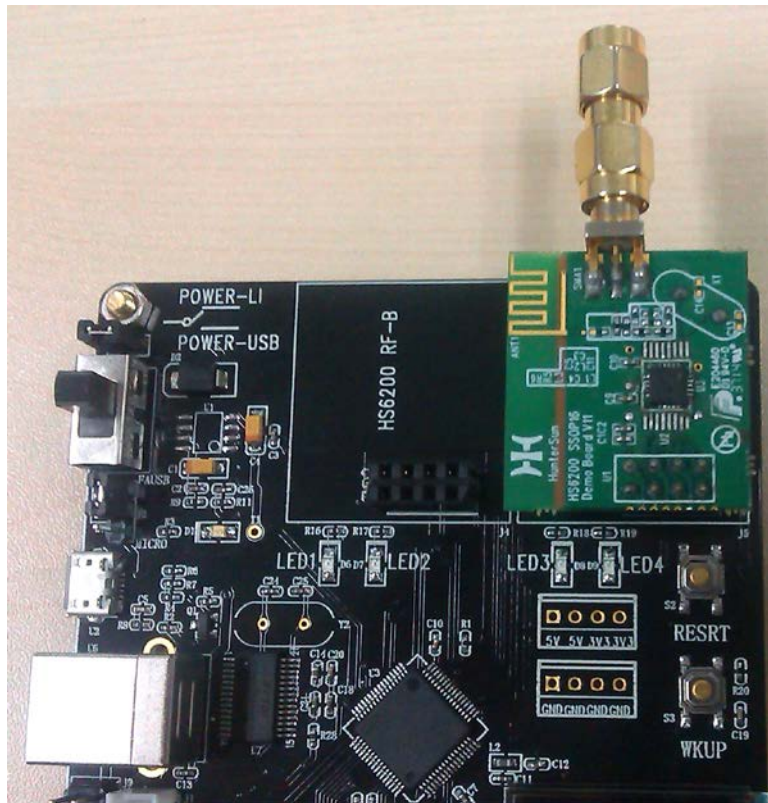


图 4.1 传导测试焊接图

射频测试需要准备以下仪器：

射频测试线：需要专门的射频线进行测试，并且需要知道测试频率下的插入损耗（IL），一般 Cable 线的插损为 0.5dB~1.5dB。

频谱分析仪（用于测发射性能）：测 2.4G 载波信号，最高频率至少 3GHz，要测试五次谐波，最高频率至少 13GHz，具体仪器型号如 Agilent PSA、ESA 系列频谱分析仪。

矢量信号发生器（用于测接收性能）：最高频率至少 3GHz，要有 Custom (I/Q) setup modulation option，

具体仪器型号如 Agilent E4438C。

4.2 发射功率测试

频谱分析仪通过射频线与 HS6200_DEMO 板的 SMA 口相连。用户可以通过配置 HS6200 使其处于发射载波模式来测量发射功率，发射功率频率测试步骤如下：

- 1、初始化 HS6200，具体参考 4.2.1 描述。
- 2、通过配置 HS6200 的 HS6200_BANK0_RF_CH 设置要测量的频点(可精确到 100kHz)。
- 3、通过配置 HS6200 的 HS6200_BANK0_RF_SETUP 设置发射功率，具体可参考 7.5 节描述。
- 4、通过配置 HS6200 的 HS6200_BANK0_RF_SETUP 寄存器 CONT_WAVE(bit7)位为 1,配置成载波模式。
- 5、配置 HS6200 的 CE 为高，此时将开始发送载波。
- 6、频谱分析仪设置: Frequency center=Frequency at Max Amplitude, such as 2440MHz at channel 40; Span<10MHz ; RBW<100kHz , 可以设为 Auto 。
- 7、记录该频率下的功率值和对应频率值。

4.3 接收灵敏度测试

4.3.1 接收灵敏度测试方法

接收灵敏度(Sensitivity) 是指当接收误比特率(BER: Bit Error Rate) 为 0.1%时的输入信号强度。

HS6200 接收灵敏度的测试方法为使用信号发生器产生包含伪随机数 PAYLOAD 的数据包，循环发送，接收端模块接收数据包，统计误码率，当接收到的数据包误码率为 0.1%时，此时信号发生器的发射功率为接收灵敏度。为了测试准确，建议发送的 bits 数至少为 1000 个。

4.3.2 接收灵敏度测试步骤

1、 配置接收端为:No-Ack , 8bit preamble, 5bytes 地址域, 32bytes 包长度, No CRC, 并配置成相应数据速率。RX 读取收到的包后，与实际已知的包比较，计算出错误 bits 数目，从而计算出 BER 。

2、 配置信号发生器：

(1) 信号发生器发送数据应配置为 Preamble+ Address+ Guard+ Packet control field+ Payload。具体数据如下：

Preamble: 0x55

Address: 0x55,0x42,0x9C,0x8F,0xC9

Guard:0x8F,0Xc9

Packet control field:010000001

Payload:256bits 伪随机数序列

(2) Input Frequency: 实际工作频率, 如 2.4GHz。

(3) Filter : Gaussian

(4) Filter Bbt : 1

(5) Symbol Rate : 250ksps、1Msps or 2Msps

(6) Modulation Type : 2-Lvl FSK

(7) Freq Dev : 160kHz for 500ksps; 160kHz for 1Msps; 320kHz for 2Msps

5 HS6200 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）。由于 FCC 标准比 CE 标准要求严格，所以此部分以 FCC 为例说明，FCC 关于 2.4G ISM 频段的规范主要参考 FCC section 15.247 和 15.249，ISM 工作频段为 2400MHz~2483.5MHz。在 FCC 标准中，最容易 FAIL 的指标是发射杂散特别是高次谐波的幅度容易超过标准范围，要过 FCC 标准最关键的是传导（连线）测试结果与 FCC 标准相比需要有足够的余量（至少 5dB），如果由于测试条件所限，至少需要满足以下几点：

射频匹配电路部分的原理图和版图完全等同于本文档中匹配电路的布局和布线，如果不同则需要重新调试元器件值；

发射功率最大为 0dBm：写 Bank0_RF_SETUP=0XC0；

发射测试模式下,需要关掉接收, 否则 RX 本振泄漏会超过标准：Bank0_CONFIG [0]=0；

由于各个客户所使用的板材和元器件都不一样，所以正确的方法是在以上几点满足的情况下，调试传导测试结果与 FCC 标准相比有足够余量，并且在量产时用相同的元器件焊接（同品牌同值）。**射频匹配电路部分的原理图和版图详见参考文档[2]。**

5.1 传导杂散测试结果

传导杂散测试需要焊接带 SMA 头的射频线到频谱分析仪来测试发射功率，具体请参考前面的测试准备和发射功率频率测试。测试之前需要校准各个频率的线损，以下是低、中、高信道下的传导杂散测试结果示例：

表 5.1 传导杂散测试结果

载波频率	谐波频率	FCC 要求: <-41.2dBm		CE 要求: <-30dBm	
(MHz)	(MHz)	实测 (dBm)	余量 (dB)	实测 (dBm)	余量 (dB)
2400(低信道)	4800	-48	7	-48	18
	7200	-65	24	-65	35
	9600	-75	34	-75	45
	12000	-75	34	-75	45
2440(中信道)	4880	-50	9	-50	20
	7320	-68	26	-68	38
	9760	-75	34	-75	45

	12200	-70	29	-70	40
2480(高信道)	4960	-51	10	-51	21
	7440	-66	25	-66	36
	9920	-75	34	-75	45
	12400	-67	26	-67	37

5.2 辐射杂散测试结果

由于 HS6200 的天线为 PCB 板上天线，所以一般 FCC 测试都为辐射情况下（空间辐射 3 米，见下图）的测试，而天线增益已经考虑在内，所以不需要另外关于天线的说明。由于谐波中辐射最强的为二次和三次谐波，所以本示例辐射杂散测试中主要的测试内容为二、三次谐波的辐射强度。

FCC 测试中对于功率信号有两种检测方式：Peak Detector（峰值检测）和 Average Detector（平均值检测），Average 为实际工作情况下的平均信号辐射功率，如无线鼠标中假设每 8ms 发送一次信号，则此值即为在 8ms 内发送的功率在 8ms 时间内的平均值，而 Peak 为发射 burst 功率的最大值，以上值在测试中都会在全方向不同高度下 MAX Hold 取最大值。

通常在测试标准中要求 Peak 功率值大于 Average 功率值为 20dB，而实际产品当中 Peak 功率值大于 Average 功率值为：

$$Pincrease=20*\log((Ton+Toff)/Ton)$$

Ton 为功率发射时间，Toff 为发射中间间隔

所以在 FCC 测试中，需要用实际产品在正常发射模式下发数据作为测试模式进行测试，而不是发单载波作为 Peak 和 Average 的测试模式；由于正常发射下，发射 PA 会 ON/OFF 导致发射频谱变宽，所以在低、中、高信道频率选择上不要太靠近 ISM 频段边缘频率，即低信道选择大于 2402MHz，而高信道小于 2480MHz，以防止在 ISM 频段以外有泄漏边带超过标准。由于 Average 值跟实际产品有关，所以下面示例中的测试结果为单载波发射下的值，即对应于 Peak 标准下的值，从结果可以看出即便是 HS6200 Module 发射单载波下的功率值在 Average 下都有相当的余量。

表 5.2 辐射杂散测试结果

载波频率 (MHz)	谐波频率 (MHz)	FCC 要求: <74dBuV/m@3m (Peak Detector) <54dBuV/m@3m (Average Detector)		
		实测 (dBuV)	Peak Detector 余量(dB)	Average Detector 余量 (dB)
2400 (低信)	4800	48	26	6

道)	7200	42	32	12
2440(中信道)	4880	51	23	3
	7320	40	34	14
2480(高信道)	4960	50	24	4
	7440	39	35	15

5.3 FCC 测试前 Checklist

当准备去过 FCC 测试之前，请检查产品是否工作在如下模式：

1. 发射功率最大为 0dBm：写 Bank0_RF_SETUP=0XC0;
2. 发射测试模式下,需要关掉接收,否则 RX 本振泄漏会超过标准：Bank0_CONFIG [0]=0;

3. 产品的测试模式需要正常工作模式下，即发数据的方式以模拟真实情况，而不是只发单载波作为测试模式；建议采用单次周期性调制发射，发射周期尽量 $\leq 1\text{ms}$ （认证实验室的扫描时间为 1ms，周期建议 1ms，如果周期大于 1ms，需要建议测试工程师加大扫描时间），发射时间尽量短（尽量发一个字节，时间为 110us），空闲时间处于 Standby-I 或 Standby-II。

4. FCC 需要测试低、中、高三个信道，产品中需要通过按钮可以直接切换不同信道，信道频率选择建议至少离开 ISM 频率边缘 2MHz 以上；

5. 测试产品需要用本产品供电电源（如电池），而不能用其它产品（如笔记本）作为电源。

如果以上设置已经完成，在自己实验室请先确保传导测试没问题；

如果测试条件允许，传导测试结果与 FCC 标准相比有足够的余量（至少 5dB）；

如没有测试条件，则至少要求射频匹配电路部分的原理图和版图完全等同于本文档匹配电路中的布局和布线，如果不相同则一定要到有条件的实验室进行预测试。

6 HS6200 常见问题总结

6.1 故障排查流程

在确定了使用的晶振正常，双方晶体外的负载电容形式一样后，请按如下步骤排查：

1、分别将收发两端配成载波模式，查看是否能正确发出载波，以确定模块没有问题，若能发出载波请执行步骤 2，不能发出载波请执行步骤 3。

2、请确定载波频偏是否符合要求($\pm 120\text{KHz}$)，若不符合，则执行步骤 4。然后查看发射功率是否与配置相符，确定原程序中收发双方模式配置是否正确，收发双方使用地址是否符合算法要求且相同，收发两端所处信道是否相同。

3、读取芯片版本号确定芯片版本正确，检查寄存器配置是否正确(有条件的话可借助逻辑分析仪查看 SPI 口收发数据)，能发出载波后执行步骤 2。

4、收发两端 bank1 的 RF_IVGEN 寄存器里的 Xtal_cc 配置是否统一，范围是否为 0pF~19pF。或者更换晶体。

6.2 如何控制 HS6200 发送载波

初始化后，配置 BANK0 寄存器 RF_SETUP 的 CONT_WAVE 位为 1 后置高 CE 即可发出载波，也可在初始化完成后，调用 HS6200.c 中接口 HS6200_ModeSwitch(Rf_Carrier_Mode)控制 6200 发送载波。

6.3 怎样使 Power down 模式下电流消耗最低

需要将 HS6200 CSN 置高，CE 置低，如果线上有上拉电阻，CLK 及 MOSI 需置高，如果线上没有上拉电阻，CLK 及 MOSI 需置低。请另外注意 MCU 空闲引脚的处理。

6.4 如何更改发射速率

HS6200 支持 500Kbps、1Mbps 和 2Mbps、三档发射速率，不同发射速率需在初始化过程中修改 HS6200_BANK1_PLL_CTL0、HS6200_BANK1_IF_FREQ、HS6200_BANK1_FDEV 和 HS6200_BANK0_RF_SETUP 四个寄存器。HS6200_BANK1_PLL_CTL0、HS6200_BANK1_IF_FREQ、HS6200_BANK1_FDEV 三个寄存器配置可参考 3.2.1 节相关说明，HS6200_BANK0_RF_SETUP 可参考 6.5 节说明，更多配置也可参照 regmap。

6.5 如何更改发射功率

发射功率由 HS6200_BANK0_RF_SETUP 和 HS6200_BANK1_RF_IVGEN 进行配置，默认为最大发射功率 8dBm，若想更换其他功率，只需在初始化阶段将 HS6200_BANK0_RF_SETUP 和 HS6200_BANK1_RF_IVGEN 寄存器的值更改成下表对应的值即可，更多配置也可参照 regmap。

	500K bps	
	HS6200_BANK0_RF_SETUP	HS6200_BANK1_RF_IVGEN
6 dbm	0x6F	0x1F,0x64,0x00,0x01
5 dbm	0x68	0x1F,0x64,0x00,0x01
4 dbm	0x2F	0x9F,0x64,0x00,0x01
0 dbm	0x2B	0x1F,0x64,0x00,0x01
-6 dbm	0x29	0x1F,0x64,0x00,0x01

-12 dbm	0x29	0x9F,0x64,0x00,0x01
-16 dbm	0x28	0x1F,0x64,0x00,0x01
-43 dbm	0x28	0x9F,0x64,0x00,0x01

	1M bps	
	HS6200_BANK0_RF_SETUP	HS6200_BANK1_RF_IVGEN
6 dbm	0x47	0x1F,0x64,0x00,0x01
5 dbm	0x40	0x1F,0x64,0x00,0x01
4 dbm	0x07	0x9F,0x64,0x00,0x01
0 dbm	0x03	0x1F,0x64,0x00,0x01
-6 dbm	0x01	0x1F,0x64,0x00,0x01
-12 dbm	0x01	0x9F,0x64,0x00,0x01
-16 dbm	0x00	0x1F,0x64,0x00,0x01
-43 dbm	0x00	0x9F,0x64,0x00,0x01

	2M bps	
	HS6200_BANK0_RF_SETUP	HS6200_BANK1_RF_IVGEN
6 dbm	0x4f	0x1F,0x64,0x00,0x01
5 dbm	0x48	0x1F,0x64,0x00,0x01
4 dbm	0x0f	0x9F,0x64,0x00,0x01
0 dbm	0x0b	0x1F,0x64,0x00,0x01
-6 dbm	0x09	0x1F,0x64,0x00,0x01
-12 dbm	0x09	0x9F,0x64,0x00,0x01
-16 dbm	0x08	0x1F,0x64,0x00,0x01
-43 dbm	0x08	0x9F,0x64,0x00,0x01

6.6 为什么我设置的自动重发次数 ARC 为非 0，系统却没有自动重发

自动重发是在在 Auto_ACK 模式下才能生效的，因此，收发双方都必须将 HS6200_BANK0_EN_AA 对应 Pipe 的寄存器使能才能使用自动重发功能。

6.7 在跳频通信中，偶尔能收到数据，而定频却没有问题

这种情况通常是由跳频机制引起的，请检查跳频机制。将收发双方设为固定频点也是我们调试中排查问题常用的手段之一。

6.8 HS6200 的可以软复位操作么

可以通过 bank1 的 PLL_CTL1 寄存器里的 soft_rst 对 HS6200 进行软复位，但尽量不要在 CE 为高时进

行软复位。

6.9 用 HS6200 的裸 die 直接邦定(Bonding)在 PCB 上需要注意什么?

HS6200 提供裸 die 供客户直接邦定在 PCB 板上, HS6200 所有的 GND 都采用 Down Bonding 方式, 具体如图 6.1 所示。

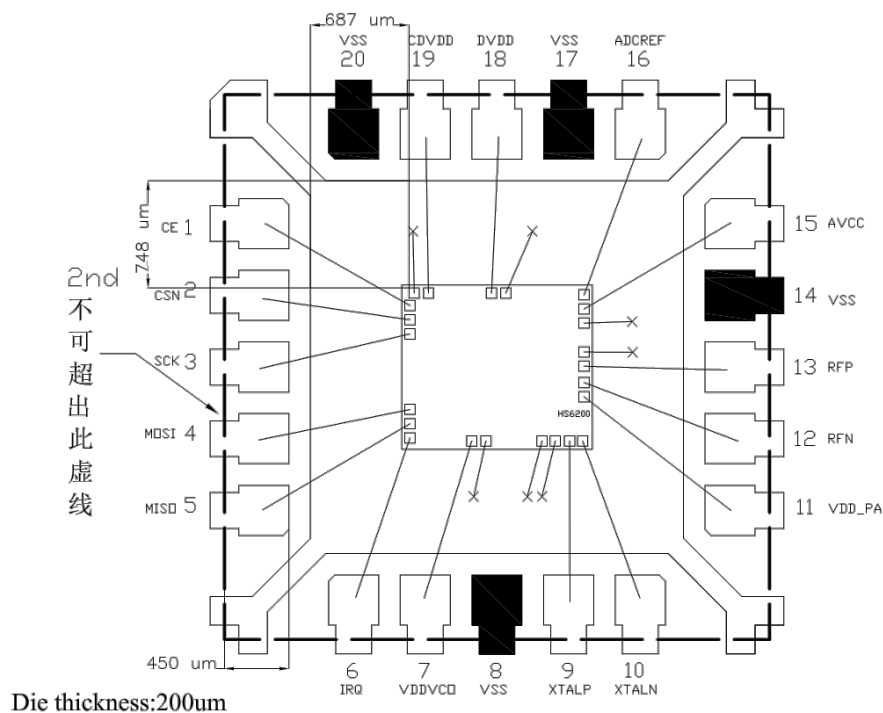


图 6.1 HS6200_Die 打线图

6.10 晶体需要外接电容么

请检查收发双方晶振外围电路器件形式及参数是否一致, 是否存在一方焊了电容一端没焊的情况。

由于电容在芯片内部, 所以电容的绝对值会有 10%左右的偏差, HS6200 的接收机可以把小于 200KHz 的载波频率偏差纠正回来。

6.11 如何增强 HS6200 的 ESD 能力

可以在片外天线上并联一个 10nH 的电感加强 ESD 保护。

6.12 HS6200 过 CE 认证需要注意的问题？

建议使用认证实验室的电源供电（直流稳压电源）或锂电池供电，使用移动电源供电时，低频杂散会超限值（Limit）。

CE 辐射测试时，建议采用连续载波发射，以减小测试误差，得到的测试结果更可靠。

6.13 HS6200 过 FCC 认证需要注意的问题？

建议采用单次周期性调制发射，发射周期尽量 $\leq 1\text{mS}$ （认证实验室的扫描时间为 1mS ，周期建议 1mS ，如果周期大于 1mS ，需要建议测试工程师加大扫描时间），发射时间尽量短（尽量发一个字节，时间为 $110\mu\text{S}$ ），空闲时间处于 Standby-I 或 Standby-II。

7 版本变更记录 Revision history

时间	内容	作者	版本号
2014/09/28	创建	肖金红 李殿君	V1.0
2014/12/15	增加射频 LAYOUT 注意事项。	肖金红	V1.1
2015/05/21	增加过 FCC/CE 认证的匹配网络。 增加 TX 发送 CE pulse 描述,修改包结构说明中 Guard 描述	肖金红 李殿君	V1.2
2015/12/23	更新 3.2.1 章节寄存器配置	李殿君	V1.3
2016/3/16	修改“表 3.8 HS6200 初始化与配置”第 9 项 CE_pulse 延时说明	李殿君	V1.4

重要声明

中科汉天下电子有限公司（以下简称汉天下电子）有权在不事先通知的情况下，随时对所提供的产品和服务进行更正、修改、增强、改进或其它更改，并有权随时中止提供任何产品和服务。客户在下订单前应获取最新的相关信息，并验证这些信息是否完整且是最新的。所有产品的销售都遵循在订单确认时所提供的汉天下电子销售条款与条件。保证其所销售的硬件产品的性能符合汉天下电子标准保修的适用规范。仅在汉天下电子保证的范围内，且汉天下电子认为有必要时才会使用测试或其它质量控制技术。除非政府做出了硬性规定，否则没有必要对每种产品的所有参数进行测试。

汉天下电子对应用帮助或客户产品设计不承担任何义务。客户应对其使用汉天下电子组件的产品和应用自行负责。为尽量减小与客户产品和应用相关的风险，客户应提供充分的设计与操作安全措施。汉天下电子不对任何汉天下电子专利权、版权、屏蔽作品权或其它与使用了汉天下电子产品或服务的组合设备、机器、流程相关的汉天下电子知识产权中授予的直接或隐含权限作出任何保证或解释。汉天下电子所发布的与第三方产品或服务有关的信息，不能构成从汉天下电子获得使用这些产品或服务的许可、授权、或认可。使用此类信息可能需要获得第三方的专利权或其它知识产权方面的许可，或是汉天下电子的专利权或其它知识产权方面的许可。

对于汉天下电子的产品手册或数据表，仅在没有对内容进行任何篡改且带有相关授权、条件、限制和声明的情况下才允许进行复制。在复制信息的过程中对内容的篡改属于非法的、欺诈性商业行为。汉天下电子对此类篡改过的文件不承担任何责任。在转售汉天下电子产品或服务时，如果存在对产品或服务参数的虚假陈述，则会失去相关汉天下电子产品或服务的明示或暗示授权，且这是非法的、欺诈性商业行为。汉天下电子对此类虚假陈述不承担任何责任。

汉天下电子产品未获得用于关键的安全应用中的授权，例如生命支持应用（在该类应用中一旦汉天下电子产品故障将预计造成重大的人员伤亡），除非各方官员已经达成了专门管控此类使用的协议。购买者的购买行为即表示，他们具备有关其应用安全以及规章衍生所需的所有专业技术和知识，并且认可和同意，尽管任何应用相关信息或支持仍可能由汉天下电子提供，但他们将独力负责满足在关键安全应用中使用其产品以及汉天下电子产品所需的所有法律、法规和安全相关要求。

此外，购买者必须全额赔偿因在此类关键安全应用中使用汉天下电子产品而对汉天下电子及其代表造成的损失。汉天下电子产品并非设计或专门用于军事/航空应用，以及环境方面的产品，除非汉天下特别注明该产品属于“军用”或“增强型塑料”产品。只有汉天下电子产品指定的军用产品才满足军用规格。购买者认可并同意，对汉天下电子未指定军用的产品进行军事方面的应用，风险由购买者单独承担，并且独力负责在此类相关使用中满足所有法律和法规要求。

汉天下电子产品并非设计或专门用于汽车应用以及环境方面的产品，除非汉天下电子特别注明该产品符合ISO/TS 16949 要求。购买者认可并同意，如果他们在汽车应用中使用任何未被指定的产品，汉天下电子对未能满足应用所要求不承担任何责任。