

图 A : SX1261 / 2 框图

### 概述

SX1261 和 SX1262 是 1GHz 以下频段无线收发芯片，其非常适合远距离无线应用。这两款芯片的接收电流只需 4.2 毫安，也非常适合要求长电池寿命的应用。SX1261 的最大发射功率可达 +15 dBm，SX1262 的最大发射功率可达 22 dBm。

它们都支持 LoRa®调制和 (G)FSK 调制。这两款芯片可以灵活的配置，以满足全球不同的 LoRaWAN™ 的应用需求标准或专有协议。

芯片的物理层也满足 LoRa 联盟发布的 LoRaWAN™ 协议规格要求。

芯片也可以应用于满足无线电法规的系统中。这些无线电法规包括但不限于 ETSI EN 300220、FCC CFR 47 Part 15，中国的监管要求和日本的 ARIB T-108。从 150MHz 到 960MHz 连续的频率覆盖范围允许支持世界上所有主要的 1GHz 以下的 ISM 频段。

### 应用

SX1261/2 的高集成度和低功耗满足了新一代物联网应用

- 智能仪表
- 供应链和物流
- 楼宇自动化
- 智慧农业
- 智慧城市
- 智慧商店
- 资产追踪
- 街灯
- 智慧停车场
- 环境监测
- 健康
- 安防
- 遥控

## 订单信息

产品编号	交付方式	最小订单量
SX1261IMLTRT	Tape & Reel	3'000 片
SX1262IMLTRT	Tape & Reel	3'000 片

QFN24 封装，无铅，无卤素，符合 RoHS / WEEE 标准的产品

## 版本历史

版本	工程变更单	日期	修订
1.0	039166	2017.10	第一次发布
1.1	040046	2017.12	<p>在 4.1.3XTAL 控制块中添加一个注释“当使用 TCXO”时用 TCXO 解释 XTA 电容值的变化；</p> <p>新增的小节 5.1.5 “关于 DC-DC 电感选择的考虑”；</p> <p>在第 6.1.1.1 节“扩频因子”中加上推荐 12 个 LoRa 前导码以获得最佳性能的注释；</p> <p>在第 9.6 节“接收模式”中添加 SetLoRaSymbNumTimeout 的注释；</p> <p>在第 12.1 节“寄存器表”中随机数 Gen 值的修正；</p>

## 目录

1.	芯片架构 .....	11
2.	引脚图 .....	12
2.1	I/O 口描述 .....	12
2.2	封装视图 .....	13
3.	规格 .....	14
3.1	ESD 注意事项 .....	14
3.2	绝对最大值 .....	14
3.3	工作范围 .....	14
3.4	晶体规格 .....	15
3.5	电气规格 .....	15
3.5.1.	功耗 .....	16
3.5.2.	通用规格 .....	18
3.5.3.	接收模式规格 .....	19
3.5.4.	发射模式规格 .....	21
3.5.5.	数字 I/O 规格 .....	21
4.	电路描述 .....	22
4.1	参考时钟 .....	22
4.1.1.	RC 基准频率 .....	22
4.1.2.	高精度基准频率 .....	22
4.1.3.	XTAL 控制单元 .....	23
4.1.4.	TCXO 控制框图 .....	24
4.2	锁相环(PLL) .....	24
4.3	接收机 .....	25
4.3.1.	中频 .....	25
4.4	发射机 .....	26
4.4.1.	SX1261 功率放大器特性 .....	27
4.4.2.	SX1262 功率放大器特性 .....	29
4.4.3.	功率放大器概要 .....	30
5.	电源分配 .....	32
5.1	选择 DC-DC 或 LDO .....	32
5.1.1.	选项 A: SX1261/ 使用 DC-DC 转换器 .....	33
5.1.2.	选项 B: SX1261/ 使用 LDO 转换器 .....	34
5.1.3.	选项 C: SX1262 /使用 DC-DC 转换器 .....	34
5.1.4.	选项 D: SX1262 使用 LDO 转换器 .....	35
5.1.5.	DC-DC 模式下电感的选型 .....	35
5.2	灵活的 DIO 供电 .....	36
6.	调制解调器 .....	37
6.1	LoRa® 调制解调器 .....	37
6.1.1.	调制参数 .....	37
6.1.2.	LoRa® 包引擎 .....	39
6.1.3.	LoRa® 帧结构 .....	40
6.1.4.	LoRa® 空中时间 .....	41
6.1.5.	LoRa® 信道状态检测 (CAD) .....	41

6.2	FSK 调制解调器 .....	42
6.2.1.	调制参数 .....	42
6.2.2.	FSK 包引擎 .....	43
6.2.3.	FSK 包格式 .....	44
7.	数据缓存 .....	47
7.1	操作原理 .....	47
7.2	接收模式下的数据缓存 .....	48
7.3	发送模式下的数据缓存 .....	48
7.4	数据缓存的使用 .....	48
8.	数字接口和数字控制 .....	49
8.1	复位 .....	49
8.2	SPI 接口 .....	49
8.2.1.	收发器在 active 模式下的 SPI 时序 .....	49
8.2.2.	离开 Sleep 模式的 SPI 时序 .....	50
8.3	多功能数字输入/输出 ( DIO ) .....	51
8.3.1.	BUSY 控制线 .....	51
8.3.2.	数字输入/输出 .....	53
8.4	数字接口状态与芯片模式 .....	53
8.5	IRQ 处理 .....	54
9.	工作模式 .....	55
9.1	Startup .....	55
9.2	校准 .....	55
9.2.1.	指定频段的镜像校准 .....	55
9.3	Sleep 模式 .....	56
9.4	STDBY 模式 .....	56
9.5	FS 模式 .....	57
9.6	RX 模式 .....	57
9.7	TX 模式 .....	58
9.7.1.	PA 启动/上升时间 .....	58
9.8	激活模式的切换时间 .....	58
9.9	芯片电路模式图形说明 .....	59
10.	主机控制接口 .....	60
10.1	命令结构 .....	60
10.2	交互终止 .....	60
11.	命令列表 .....	61
11.1	工作模式命令 .....	61
11.2	寄存器和缓存访问命令 .....	62
11.3	DIO 和 IRQ 控制 .....	62
11.4	射频、调制和数据包命令 .....	62
11.5	状态命令 .....	63
12.	寄存器映射 .....	64
12.1	寄存器列表 .....	64
13.	命令接口 .....	66
13.1	工作模式配置函数 .....	66

13.1.1.	SetSleep .....	66
13.1.2.	SetStandby.....	67
13.1.3.	SetFs.....	67
13.1.4.	SetTx.....	67
13.1.5.	SetRx.....	68
13.1.6.	StopTimerOnPreamble .....	69
13.1.7.	SetRxDutyCycle .....	70
13.1.8.	SetCAD.....	72
13.1.9.	SetTxContinuousWave.....	72
13.1.10.	SetTxInfinitePreamble .....	73
13.1.11.	SetRegulatorMode.....	73
13.1.12.	Calibrate .....	73
13.1.13.	CalibrateImage.....	74
13.1.14.	SetPaConfig.....	75
13.1.15.	SetRXTxFallbackMode .....	76
13.2	寄存器和缓存区访问.....	77
13.2.1.	WriteRegister 函数.....	77
13.2.2.	ReadRegister 函数.....	77
13.2.3.	WriteBuffer 函数.....	77
13.2.4.	ReadBuffer 函数.....	78
13.3	DIO 和 IRQ 控制函数.....	78
13.3.1.	SetDioIrqParams.....	78
13.3.2.	IrqMask.....	78
13.3.3.	GetIrqStatus.....	79
13.3.4.	ClearIrqStatus .....	80
13.3.5.	SetDIO2AsRfSwitchCtrl .....	80
13.3.6.	SetDIO3AsTCXOctrl .....	80
13.4	射频调制和包相关函数.....	82
13.4.1.	SetRfFrequency.....	82
13.4.2.	SetPacketType .....	82
13.4.3.	GetPacketType.....	83
13.4.4.	SetTxParams .....	83
13.4.5.	SetModulationParams .....	84
13.4.6.	SetPacketParams .....	87
13.4.7.	SetCadParams.....	91
13.4.8.	SetBufferBaseAddress .....	93
13.4.9.	SetLoRaSymbNumTimeout .....	93
13.5	通信状态信息.....	94
13.5.1.	GetStatus .....	94
13.5.2.	GetRxBufferStatus .....	95
13.5.3.	GetPacketStatus .....	95
13.5.4.	GetRssiInst .....	96
13.5.5.	GetStats .....	96
13.5.6.	ResetStats .....	96
13.6	其它.....	97
13.6.1.	GetDeviceErrors.....	97
13.6.2.	ClearDeviceErrors .....	97

14.	应用 .....	98
14.1	主机 API 基本读写功能 .....	98
14.2	基本 TX 操作的电路配置 .....	98
14.3	基本 RX 操作的电路配置 .....	99
14.4	按正确的顺序发送命令 .....	99
14.5	应用原理图 .....	100
14.5.1.	SX1261 带射频开关的应用设计 .....	100
14.5.2.	SX1262 带射频开关的应用设计 .....	100
15.	封装信息 .....	101
15.1	封装外形图 .....	101
15.2	包装标识 .....	102
15.3	焊盘布局 .....	102
15.4	回流焊曲线 .....	103

图 2-1: SX1261/2 QFN 4x4 24L 封装的引脚分布俯视图 .....	13
图 4-1: SX1261/2 方框图 .....	22
图 4-2: TCXO 控制图 .....	24
图 4-3: DC-DC 模式下的功率放大器供电原理 .....	27
图 4-4: SX1261 的 VR_PA 电压 VS 输出功率 .....	27
图 4-5: SX1261 在 DC-DC 模式下的电流 VS 输出功率 .....	28
图 4-6: SX1261 在 LDO 模式下的电流 VS 输出功率 .....	28
图 4-7: SX1261 在 LDO 或 DC-DC 模式下的输出功率线性度 .....	29
图 4-8: SX1262 的 VR_PA 电压 VS 输出功率 .....	29
图 4-9: SX1262 的输出功率线性度 .....	30
图 4-10: SX1262 的输出功率 VS 电流 .....	30
图 5-1: SX1261 使用 DC-DC 下的拓扑 .....	33
图 5-2: SX1261 使用 LDO 下的拓扑 .....	34
图 5-3: SX1262 使用 DC-DC 下的拓扑 .....	34
图 5-4: SX1262 使用 LDO 下的拓扑 .....	35
图 5-5: 分离的 DIO 供电 .....	36
图 6-1: LoRa® 信号带宽 .....	38
图 6-2: LoRa® 包格式 .....	40
图 6-3: 定长包格式 .....	44
图 6-4: 可变长包格式 .....	44
图 6-5: 线性反馈移位寄存器的数据白化操作 .....	45
图 7-1: 数据缓存图 .....	47
图 8-1: SPI 时序图 .....	49
图 8-2: SPI 时序转换 .....	50
图 8-3: 切换时间的定义 .....	51
图 8-4: 在激活模式下切换时间的定义 .....	52
图 9-1: 芯片电路模式 .....	59
图 13-1: 检测到前导或包头时停止超时时钟 .....	70
图 13-2: RX DUTY CYCLE 模式下功耗情况 .....	71
图 13-3: 接收时的 RX DUTY CYCLE .....	72
图 14-2: SX1261 带射频开关的应用设计 .....	100
图 14-3: SX1262 带 RF 开关的应用设计 .....	100
图 15-1: QFN 4x4 封装外型图 .....	101
图 15-2: SX1261/2 标识 .....	102
图 15-3: QFN 4x4MM LAND PATTERN .....	102

表 2-1: SX1261/2 引脚表 .....	12
表 3-1: ESD 和 LATCH-UP 参数 .....	14
表 3-2: 绝对最大额定值 .....	14
表 3-3: 工作范围 .....	14
表 3-4: 晶体规格 .....	15
表 3-5: 功耗 .....	16
表 3-6: 发射模式的功耗 .....	17
表 3-7: 通用规格 .....	18
表 3-8: 接收模式规格 .....	19
表 3-9: 发射模式规格 .....	21
表 3-10: 数字 I/O 规格 .....	21
表 4-1: 引脚内部电容配置 .....	23
表 4-2: FSK 模式下的中频频率 .....	25
表 4-3: LoRa® 模式下的中频频率 .....	26
表 4-4: 功率放大器参数汇总 .....	31
表 5-1: 电源类型 VS 电路模式 .....	32
表 5-2: 过流保护配置 .....	32
表 5-3: 典型的 15 MH 电感 .....	35
表 6-1: 扩频因子的范围 .....	38
表 6-2: LoRa® 模式下的信号带宽设置 .....	38
表 6-3: 编码率的头 .....	39
表 6-4: FSK 数据包类型下的带宽定义 .....	42
表 6-5: 白化操作初始值 .....	45
表 6-6: CRC 类型配置 .....	46
表 6-7: CRC 初始值 .....	46
表 6-8: CRC 多项式 .....	46
表 8-1: SPI 时序要求 .....	50
表 8-2: 切换时间 .....	52
表 8-3: 每种芯片模式的数字端口配置 .....	53
表 8-4: IRQ 状态寄存器 .....	54
表 9-1: SX1261/2 工作模式 .....	55
表 9-2: 覆盖 ISM 频段的镜像校准 .....	56
表 9-3: RX 增益配置 .....	57
表 10-1: SPI 接口命令序列 .....	60
表 11-1: 选择芯片工作模式的命令 .....	61
表 11-2: 访问芯片寄存器和 FIFO 缓存的命令 .....	62
表 11-3: 控制芯片 IRQs 和 DIOs 的命令 .....	62
表 11-4: 控制射频和数据包设置的命令 .....	62
表 11-5: 返回芯片状态命令 .....	63
表 12-1: 寄存器列表 .....	64
表 13-1: SETSLEEP SPI 交互 .....	66
表 13-2: SLEEP 模式定义 .....	66
表 13-3: STDBY 模式配置 .....	67
表 13-4: STDBY 模式配置 .....	67
<b>表 13-5: SETFS SPI 交互 .....</b>	<b>67</b>
<b>表 13-6: SETTX SPI 交互 .....</b>	<b>67</b>
表 13-7: SETTX 超时时间长度 .....	68
表 13-8: SETRX SPI 交互 .....	68
表 13-9: SETRX 超时时间长度 .....	69

表 13- 10: STOPTIMERONPREAMBLE SPI 交互 .....	69
表 13- 11: STOPONPREAMBPARAM 定义.....	69
表 13- 12: SETRXDUTYCYCLE SPI 交互.....	70
表 13- 13: SETCAD SPI 交互.....	72
表 13- 14: SETTXCONTINUOUSWAVE SPI 交互.....	72
表 13- 15: SENDTXINFINITEPREAMBLE SPI 交互.....	73
表 13- 16: SETREGULATORMODE SPI 交互 .....	73
表 13- 17: CALIBRATE SPI 交互 .....	73
表 13- 18: 校准参数设置 .....	74
表 13- 19: CALIBRATEIMAGE SPI 交互.....	74
表 13- 20: SETPACONFIG SPI 交互 .....	75
表 13- 21: PA 最优配置.....	76
表 13- 22: SETRXTXFALLBACKMODE SPI 交互.....	76
表 13- 23: FALLBACKMODE 定义.....	76
表 13- 24: WRITEREGISTER SPI 交互.....	77
表 13- 25: READREGISTER SPI 交互 .....	77
表 13- 26: WRITEBUFFER SPI 交互.....	77
表 13- 27: READBUFFER SPI 交互.....	78
表 13- 28: SETDIOIRQPARAMS SPI 交互.....	78
表 13- 29: IRQ 寄存器 .....	79
表 13- 30: GETIRQSTATUS SPI 交互.....	79
表 13- 31: CLEARIRQSTATUS SPI 交互.....	80
表 13- 32: SETDIO2ASRFSWITCHCTRL SPI 交互.....	80
表 13- 33: ENABLE 配置项定义.....	80
表 13- 34: SETDIO3ASTCXOCTRL SPI 交互.....	80
表 13- 35: TCXOVOLTAGE 配置定义.....	81
表 13- 36: SETRFFREQUENCY SPI 交互.....	82
表 13- 37: SETPACKETTYPE SPI 交互.....	82
表 13- 38: PACKETTYPE 定义 .....	82
表 13- 39: GETPACKETTYPE SPI 交互.....	83
表 13- 40: SETTXPARAMS SPI 交互 .....	83
表 13- 41: RAMP TIME 定义.....	83
表 13- 42: SETMODULATIONPARAMS SPI 交互.....	84
表 13- 43: GFSK MODPARAM1, MODPARAM2 & MODPARAM3 – BR.....	84
表 13- 44: GFSK MODPARAM4 – PULSESHAPE.....	84
表 13- 45: GFSK MODPARAM5 - BANDWIDTH.....	85
表 13- 46: GFSK MODPARAM6, MODPARAM7 & MODPARAM8 - FDEV.....	85
表 13- 47: LoRa® MODPARAM1- SF.....	86
表 13- 48: LoRa® MODPARAM2 – BW .....	86
表 13- 49: LoRa® MODPARAM3 - CR.....	86
表 13- 50: LoRa® MODPARAM4 – LOWDATARATEOPTIMIZE .....	87
表 13- 51: SETPACKETPARAMS SPI TRANSACTION .....	87
表 13- 52: GFSK PACKETPARAM1 & PACKETPARAM2 - PREAMBLELENGTH .....	87
表 13- 53: GFSK PACKETPARAM3 - PREAMBLEDETECTORLENGTH.....	87
表 13- 54: GFSK PACKETPARAM4 - SYNCWORDLENGTH.....	88
表 13- 55: 同步字编程地址 .....	88
表 13- 56: GFSK PACKETPARAM5 – ADDR COMP.....	88
表 13- 57: 节点地址编程 .....	88
表 13- 58: 广播地址编程 .....	89
表 13- 59: GFSK PACKETPARAM6 - PACKETTYPE .....	89

表 13- 60:	GFSK PACKETPARAM7 - PAYLOADLENGTH .....	89
表 13- 61:	GFSK PACKETPARAM8 - CRCTYPE .....	89
表 13- 62:	CRC 初始值编程.....	90
表 13- 63:	CRC 多项式编程.....	90
表 13- 64:	GFSK PACKETPARAM9 白化 .....	90
表 13- 65:	白化初始值设置 .....	90
表 13- 66:	LORA@PACKETPARAM1 & PACKETPARAM2 - PREAMBLELENGTH.....	90
表 13- 67:	LORA® PACKETPARAM3 – HEADER TYPE .....	91
表 13- 68:	LORA® PACKETPARAM4 – PAYLOADLENGTH .....	91
表 13- 69:	LORA® PACKETPARAM5 - CRCTYPE .....	91
表 13- 70:	LORA® PACKETPARAM6 - INVERTIQ.....	91
表 13- 71:	SETCADPARAMS SPI TRANSACTION .....	91
表 13- 72:	CAD 码元数定义.....	92
表 13- 73:	使用 4 个码元检测 CAD 时， CADDETPEAK 和 CADDETMIN 的推荐设置 .....	92
表 13- 74:	CAD 退出模式定义 .....	92
表 13- 75:	SETBUFFERBASEADDRESS SPI TRANSACTION .....	93
表 13- 76:	SETLORASYMBNUMTIMEOUT SPI TRANSACTION .....	93
表 13- 77:	状态字节定义 .....	94
表 13- 78:	GETSTATUS SPI TRANSACTION .....	94
表 13- 79:	GETRXBUFFERSTATUS SPI TRANSACTION.....	95
表 13- 80:	GETPACKETSTATUS SPI TRANSACTION.....	95
表 13- 81:	STATUS BIT.....	95
表 13- 82:	GETRSSIINST SPI TRANSACTION .....	96
表 13- 83:	GETSTATS SPI TRANSACTION .....	96
表 13- 84:	GETDEVICEERRORS SPI TRANSACTION.....	97
表 13- 85:	OPERROR BITS.....	97
表 13- 86:	CLEARDEVICEERRORS SPI TRANSACTION.....	97

## 1. 芯片架构

SX1261/2 是半双工收发芯片，能够低功耗的工作在 150 – 960MHz ISM 频段。其包括四个主要功能部分：

1. 模拟前端部分：发射和接收链路，以及数据转换接口。SX1261 与 SX1262 的发射链路的最后级电路是不相同的。SX1261 在 DC-DC 或者 LDO 模式下都能够输出最大 +14/15dBm 的功率。SX1262 在电池供电模式下能够输出最大到+22dBm 的功率。
2. 数字调制解调部分：SX1261/2 能够处理以下调制解调方式：
  - LoRa® Rx/Tx，带宽 = 7.8 - 500 kHz, SF5 to SF12, 比特速率 = 0.018 - 62.5 kb/s
  - (G)FSK Rx/Tx，比特速率 = 0.6 - 300 kb/s
3. 数字接口和控制部分：这部分包括所有的有效载荷和协议处理以及通过 SPI 接口对芯片进行配置的功能。
4. 电源部分：包含两种形式的电压转换器：DC-DC 或者 LDO。具体使用根据应用要求设计。

## 2. 引脚图

### 2.1 I/O 口描述

表 2-1: SX1261/2 引脚表

引脚序号	引脚名称	类型 (I = 输入 O = 输出)	描述
0	GND	-	底部外露的接地焊盘
1	VDD_IN	I	功率放大器的稳压源输入电压, VR_PA SX1261: 连接到引脚 7 SX1262: 连接到引脚 10
2	GND	-	接地
3	XTA	-	晶体振荡器连接, 可用与输入外部参考时钟
4	XTB	-	晶体振荡器连接
5	GND	-	接地
6	DIO3	I/O	多功能数字 I/O 口 - 外部 TCXO 电源电压
7	VREG	O	内部稳压源 LDO/DC-DC 的输出电压
8	GND	-	接地
9	DCC_SW	O	DC-DC 开关输出
10	VBAT	I	芯片电源
11	VBAT_IO	I	数字 I/O 接口引脚的电源 (不包括 DIO3)
12	DIO2	I/O	多功能数字 I/O 口/ 射频开关控制
13	DIO1	I/O	多功能数字 I/O 口
14	BUSY	I/O	Busy 指示引脚
15	NRESET	I/O	复位信号, 低有效
16	MISO	O	SPI 从输出
17	MOSI	I	SPI 从输入
18	SCK	I	SPI 时钟
19	NSS	I	SPI 从片选
20	GND	-	接地
21	RFI_P	I	RF 接收输入
22	RFI_N	I	射频接收输入
23	RFO	O	射频发射输出(SX1261 低功率放大器 或者 SX1262 高功率 放大器)
24	VR_PA	-	转换后的功率放大器的电源

2.2 封装视图

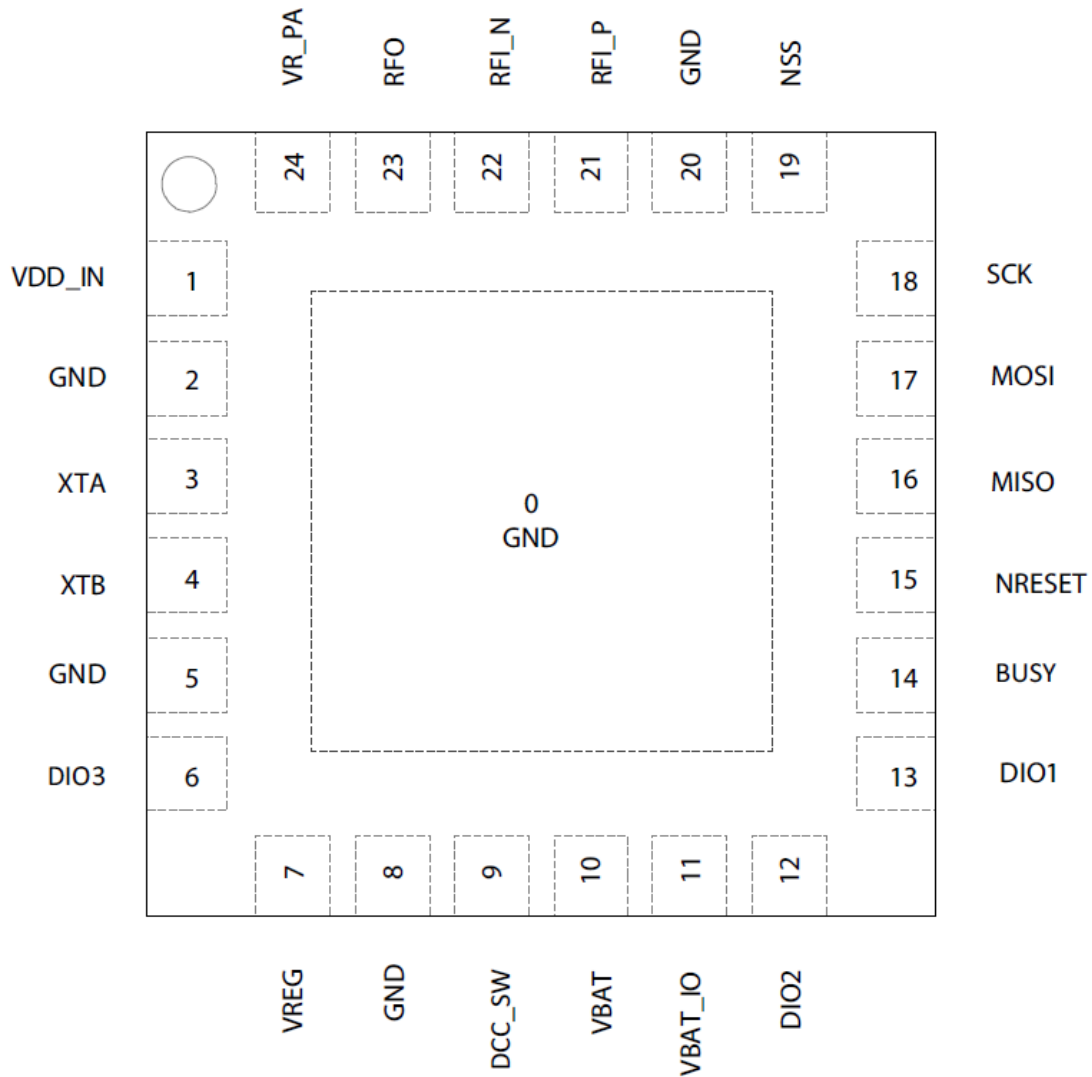


图 2-1: SX1261/2 QFN 4x4 24L 封装的引脚分布俯视图

### 3. 规格

#### 3.1 ESD 注意事项

SX1261/2 是一款高性能的射频器件，具有高 ESD 和门锁电阻。为避免器件永久性损坏，应采取所有必要的 ESD 防范措施。

表 3-1: ESD 和门锁参数

符号	描述	最小值	典型值	最大值	单位
ESD_HBM	ANSI/ESDA/JEDEC标准JS - 1001-2014第2类(人体模型)	-	-	2.0	kV
ESD_HBM	ESD充电装置模型, JEDEC标准 JESD22-C101D, III类	-	-	1000	V
LU	门锁 JEDEC 标准 JESD78 B, I类 A 级	-	-	100	mA

#### 3.2 绝对最大值

如果超出下表所列的绝对最大值，可能会导致器件永久性损坏；而长期在绝对最大值条件下工作会影响器件的可靠性。

表 3-2: 绝对最大值

符号	描述	最小值	典型值	最大值	单位
V <sub>DDmr</sub>	输入到 VBAT 和 VBAT_IO 的电源电压	-0.5	-	3.9	V
T <sub>mr</sub>	温度	-55	-	125	°C
P <sub>mr</sub>	射频输入电平	-	-	10	dBm

#### 3.3 工作范围

工作范围定义了芯片的正常工作时参数的限值。如果超出了这些限值，不能保证芯片的功能。

表 3-3: 工作范围

符号	描述	最小值	典型值	最大值	单位
V <sub>DDop</sub>	输入到 VBAT 和 VBAT_IO 的电源电压	1.8	-	3.7	V
Top	工作温度	-40	-	85	°C
C <sub>lop</sub>	数字端口的负载电容	-	-	20	pF
ML	射频输入功率	-	-	0	dBm
VSWR	电压驻波比	-	-	10 : 1	-

### 3.4 晶体规格

表 3-4: 晶体规格

符号	描述	最小值	典型值	最大值	单位
FXOSC	晶体振荡器频率	-	32	-	MHz
CLOAD	晶体负载电容	-	10	-	pF
COXTAL	晶体等效并联电容	0.3	0.6	2	pF
RSXTAL	晶体等效串联电阻	-	30	60	Ω
CMXTAL	晶体动态电容	1.3	1.89	2.5	fF
DRIVE	驱动电平	-	-	100	uW

参考频率精度由发射机和接收机的精度、以及环境参数，比如极限温度等整个系统来确定。在 LoRaWAN™ 系统中，所有工作条件下，期望的终端产品参考频率精度大约是 +/-30ppm。这些条件包括初始误差、温度漂移和老化。

### 3.5 电气规格

电气规格基于以下的条件测试所得，除非另有说明：

- VBAT\_IO = VBAT = 3.3 V, 所有的电流消耗都是基于 VBAT 连接至 VBAT\_IO 所得；
- 温度 = 25°C；
- FXOSC = 32 MHz, 用特定的晶体；
- FRF = 434/490/868/915 MHz；
- 所有的 RF 阻抗都已匹配；
- 发射输出功率是指在50 ohm负载下的输出功率；
- FSK BER = 0.1%, 不带前置滤波的 2FSK 调制, BR = 4.8 kb/s, FDA = ± 5 kHz, BW\_F = 20 kHz 双边带；
- LoRa® PER = 1%, 64 字节包长, 8 个码元的前导, CR = 4/5, 有效载荷 CRC 功能打开, 显式包头模式；
- RX/TX 的指标是基于默认的 RX 增益设置以及 RX 和 TX 匹配是共用匹配连接下测试所得；
- 抗阻塞性，邻信道抑制，同信道抑制是基于单音干扰信号以及有用信号电平在灵敏度+3dB 的情况下测试所得；
- TCXO 和射频开关的功耗不包括在以下指标中；

#### 注意!

在整个文档中，无论是 LoRa® 调制还是 FSK 调制，所有的接收机带宽都是指双边带带宽。

### 3.5.1. 功耗

表 3-5: 功耗

符号	模式	描述	最小值	典型值	最大值	单位
IDDOFF	OFF 模式 (冷启动的 sleep 模式 1)	所有功能块关闭	-	160	-	nA
IDDSL	SLEEP 模式 (热启动的 sleep 模式 2)	保留配置 保留配置+RC64K	-	600 1.2	-	nA uA
IDDSBR	STDBY_RC 模式	RC13M, XOSC 关闭	-	0.6	-	mA
IDDSBX	STDBY_XOSC 模式	XOSC 开启	-	0.8	-	mA
IDDFS	FS 模式	DC-DC 模式 LDO 模式	-	2.1 3.55	-	mA mA
IDDRX	用 DC-DC 的接收模式	FSK 4.8 kb/s	-	4.2	-	mA
		LoRa® 125 kHz	-	4.6	-	mA
		Rx Boosted <sup>3</sup> , FSK 4.8 kb/s	-	4.8	-	mA
		Rx Boosted, LoRa® 125 kHz	-	5.3	-	mA
		LoRa® 125 kHz, VBAT = 1.8 V	-	8.2	-	mA
	用 LDO 的接收模式	FSK 4.8 kb/s	-	8	-	mA
		LoRa® 125 kHz	-	8.8	-	mA
		Rx Boosted <sup>3</sup> , FSK 4.8 kb/s	-	9.3	-	mA
		Rx Boosted, LoRa® 125 kHz	-	10.1	-	mA

1. 冷启动相当于器件在上电复位(POR), 或者器件在所有功能模块都关闭的 sleep 模式下被唤醒, 详见第 66 页的 [13.1.1 节 “SetSleep”](#) ;
2. 热启动是器件在保留配置的 sleep 模式下被唤醒, 详见第 66 页的 [13.1.1 节 “SetSleep”](#) ;
3. 对于怎样设置器件进入 Rx Boosted 增益 gain 模式, 详见第 57 页的 [9.6 节 “RX 模式”](#) ;

表 3-6: 发射模式的功耗

符号	频段	PA匹配/条件	输出功率	典型值	单位
IDDTX SX1261 <sup>1</sup>	868/915MHz	+14dBm	+14 dBm, VBAT = 3.3 V	25.5	mA
			+10 dBm VBAT = 3.3 V	18	mA
			+14 dBm, VBAT = 1.8 V	48	mA
			+10 dBm, VBAT = 1.8 V	34	mA
		+14dBm/optimal settings <sup>2</sup>	+15 dBm, VBAT = 3.3 V	32.5	mA
			+10 dBm VBAT = 3.3 V	15	mA
			+15 dBm, VBAT = 1.8 V	60	mA
			+10 dBm, VBAT = 1.8 V	29	mA
	434/490MHz	+14 dBm	+15 dBm, VBAT = 3.3 V	25.5	mA
			+14 dBm, VBAT = 3.3 V	21	mA
+10 dBm, VBAT = 3.3 V			14.5	mA	
+15 dBm, VBAT = 1.8 V			46.5	mA	
+14 dBm, VBAT = 1.8 V			39	mA	
+10 dBm, VBAT = 1.8 V			26	mA	
IDDTX SX1262 <sup>3</sup>	868/915MHz	+22dBm	+22 dBm	118	mA
			+20 dBm	102	mA
			+17 dBm	95	mA
			+14 dBm	90	mA
		+20 dBm/optimal settings <sup>4</sup>	+20 dBm	84	mA
			+17 dBm/optimal settings <sup>4</sup>	58	mA
			+14 dBm/optimal settings <sup>4</sup>	45	mA
			434/490MHz	+22dBm	+22 dBm
	+20 dBm	90			mA
	+17 dBm	75			mA
+14 dBm	63	mA			
+20 dBm/optimal settings <sup>4</sup>	65	mA			
+17 dBm/optimal settings <sup>4</sup>	42	mA			
+14 dBm/optimal settings <sup>4</sup>	32	mA			

1. 对于 SX1261, DC-DC 模式应用于整个 IC。详见 32 页 [5.1 节 "选择 DC-DC 或 LDO"](#)。

2. 关于最优设置, 详见 75 页 [13.1.14.1 节"PA 最优设置"](#)。

3. 对于 SX1262, DC-DC 模式应用于 IC 内核, PA 由 VBAT 供电。详见 32 页 [5.1 节 "选择 DC-DC 或 LDO"](#)。

4. 适用指定输出功率的最优设置, 详见 75 页 [13.1.14.1 节"PA 最优设置"](#)。

## 3.5.2. 通用规格

表 3-7: 通用规格

符号	描述	条件	最小值	典型值	最大值	单位
FR	合成器频率范围	SX1261	150		960	MHz
FSTEP	合成器频率步进	-	-	0.95	-	Hz
PHN <sup>1,2</sup>	合成器噪声 (868/915MHz)	偏移 1 kHz 偏移 10 kHz 偏移 100 kHz 偏移 1MHz 偏移 10 MHz	-	-75 -95 -100 -120 -135	-	dBc/Hz dBc/Hz dBc/Hz dBc/Hz dBc/Hz
TS_FS	合成器唤醒时间	从 STDBY_XOSC 模式	-	40	-	us
TS_HOP	合成器跳频时间	10MHz 步进	-	30	-	us
TS_OSC	晶体振荡器唤醒时间	从 STDBY_RC <sup>3</sup> 模式	-	150	-	us
OSC_TRM	晶体频率误差补偿的调节范围 <sup>4</sup>	最小/最大晶体规格	+/- 15	+/- 30	-	ppm
BR_F	比特速率, FSK	可编程 最小调制指数是 0.5	0.6	-	300 <sup>5</sup>	Kb/s
FDA	频偏, FSK	可编程 FDA+BR_F/2=<250KHz	0.6	-	200	KHz
BR_L	比特速率, LoRa	最小: SF12, BW_L=7.8KHz 最大: SF5, BW_L=7.8KHz	0.018	-	62.5 <sup>6</sup>	Kb/s
BW_L	信号带宽, LoRa	可编程	7.8	-	500 <sup>6</sup>	KHz
SF	扩频系数, LoRa	可编程, 码片/码元=2^SF	5	-	12	-
VTCXO	TCXO 电源变换器的电压范围	典型条件下最小/最大值 默认设置下的典型值 VDDop>VTCXO+200mV	1.6	1.7	3.3	V
ILTCXO	TCXO 电源变换器的负载电流		-	1.5	4	mA
TSVTCXO	TCXO 电源变换器的启动时间	从使能到升至偏离目标电压 25mV 以内	-	-	100	us
IDDTXO	TCXO 电源变换器的电流消耗	静态电流 相对于负载电流	- -	- 1	70 2	uA %
ATCXO	外部 TCXO 加载到 XTA 引脚的电压幅度	通过 220Ω电阻串联 10 PF 电容, 详见 24 页 4.2 节 “锁相环”	0.4	0.6	1.2	Vpk-pk

- 给出了用于特定调制/比特速率的推荐锁相环带宽的相位噪声规格, 最优设置可用于特定应用。
- 由于 PLL 的拓扑结构, 相位噪声在频率上不是固定的, 对于两个彼此接近的频率, 相位噪声可能有明显的变化;
- 唤醒时间是指直到晶体振荡器频率稳定在 ±10 ppm 以内;
- OSCSRIM 是可以用来补偿晶体初始频率误差和实现晶体温度补偿的有效调节范围; 整个可用调节范围更高, 并允许补偿所有 IC 工艺变化;
- 最大比特速率与 RF 频率成比例; 例如在 869/915 MHz 频段, 最大比特速率可以到 300kb/s, 而在 150MHz, 最大比特速率只可以到 50kb/s;
- 对于 400MHz 以下的频率, 频率和支持的带宽有一定的比例关系, 有一些带宽在 400MHz 以下是不可使用的;

### 3.5.3. 接收模式规格

表 3- 8: 接收模式规格

符号	描述	条件	最小值	典型值	最大值	单位
RXS_2FB	2-FSK 灵敏度, RX boosted 增益, 参见 57 页 9.6 节 “接收模式”, RX 和 TX 的射频通路分离, 未考虑射频开关插损	BR_F = 0.6 kb/s, FDA = 0.8 kHz, BW_F= 4 kHz	-	-125	-	dBm
		BR_F = 1.2 kb/s, FDA = 5 kHz, BW_F= 20 kHz	-	-123	-	dBm
		BR_F = 4.8 kb/s, FDA = 5 kHz, BW_F= 20 kHz	-	-118	-	dBm
		BR_F=38.4 kb/s, FDA = 40 kHz, BW_F=160kHz	-	-109	-	dBm
		BR_F=250 kb/s, FDA =125 kHz, BW_F=500 kHz	-	-104	-	dBm
RXS_LB	LoRa 灵敏度, RX boosted 增益, 参见 57 页 9.6 节 “接收模式”, RX 和 TX 的射频通路分离, 未考虑射频开关插损	BW_L = 10.4 kHz, SF = 7	-	-134	-	dBm
		BW_L = 10.4 kHz, SF = 12	-	-148	-	dBm
		BW_L = 125 kHz, SF = 7	-	-124	-	dBm
		BW_L = 125 kHz, SF = 12	-	-137	-	dBm
		BW_L = 250 kHz, SF = 7	-	-121	-	dBm
		BW_L = 250 kHz, SF = 12	-	-134	-	dBm
		BW_L = 500 kHz, SF = 7	-	-117	-	dBm
BW_L = 500 kHz, SF = 12	-	-129	-	dBm		
RXS_2F	2FSK 灵敏度, 节电模式增益, RX 和 Tx 的射频通路直接连接	BR_F=4.8 kb/s,FDA=5kHz,BW_F=20 kHz	-	-115	-	dBm
RXS_L	LoRa 灵敏度, 节电模式增益, RX 和 Tx 的射频通路直接连接	BW_L = 125 kHz, SF = 12	-	-133	-	dBm
CCR_F	同信道抑制, FSK		-	-9	-	dB
CCR_L	同信道抑制, LoRa	SF=7	-	5	-	dB
		SF=12	-	19	-	dB
ACR_F	邻信道抑制, FSK	偏移频率= +/- 50KHz	-	45	-	dB
ACR_L	邻信道抑制, LoRa	偏移频率= +/- 1.5 x BW_L		60		dB
		BW_L = 125 kHz, SF = 7 BW_L = 125 kHz, SF = 12		72		dB
BI_F	阻塞性, FSK	BR_F=4.8 kb/s,FDA=5kHz,BW_F=20 kHz		68		dB
		偏移频率= +/- 1MHz	-	70	-	dB
		偏移频率= +/- 2MHz	-	80	-	dB
BI_L	阻塞性, LoRa	BW_L = 125 kHz, SF = 12		88		dB
		偏移频率= +/- 1MHz	-	90	-	dB
		偏移频率= +/- 2MHz 偏移频率= +/- 10MHz	-	99	-	dB
IIP3	三阶输入交调截止点	偏移本振 1MHz 和 1.96MHz 的无用单音信号	-	-5	-	dBm
IMA	镜像衰减	无 IQ 校准	-	35	-	dB
		有 IQ 校准	-	54	-	dB

BW_F	双边带带宽, FSK	可编程, 典型值	4.8	-	467	KHz
TS_RX	接收机唤醒时间	FS 到 RX	-	41	-	us
FERR_L	发射机和接收机之间最大容许的频率偏移量(SF5-SF12, 灵敏度不恶化)	所有带宽, BW 的±25% 更严格的限制, 参看下表		±25%		BW
	发射机和接收机之间最大容许的频率偏移量(SF10-SF12, 灵敏度不恶化)	SF12	-50	-	50	ppm
		SF11	-100	-	100	ppm
		SF10	-200	-	200	ppm

### 3.5.4. 发射模式规格

表 3-9: 发射模式规格

符号	描述	条件	最小值	典型值	最大值	单位
TXOP	最大射频输出功率	最高功率设置				
		SX1261		+14/15 <sup>1</sup>	-	dBm
		SX1262		+22	-	dBm
TXDRP	射频输出功率 vs 电源电压	SX1261, DC-DC 或者 LDO		0.5		dB
		VDD 电压范围从 1.8V 到 3.7 V				
		SX1262, +22 dBm, VBAT = 2.7 V		2	-	dB
		SX1262, +22 dBm, VBAT = 2.4 V		3		dB
		SX1262, +22 dBm, VBAT = 1.8 V		6		dB
TXPRNG	射频输出功率范围	31 级步进, 可编程, 典型值	TXOP-31	-	TXOP	dBm
TXACC	射频输出功率步进精度		-	+/-2	-	dB
TXRMP	功率放大器 Ramp 时间	可编程	10	-	3400	us
TS_TX	TX 唤醒时间	启用频率合成器	-	36+PA Ramping	-	us

1. 对于 SX1261, 用特定的设置可以输出 +15dBm 的射频功率, 参见第 75 页 [13.1.14.1 节 “PA 最优设置”](#) ;

### 3.5.5. 数字 I/O 规格

表 3-10: 数字 I/O 规格

符号	描述	条件	最小值	典型值	最大值	单位
VIH	输入高电平	-	0.7*VBAT_IO <sup>1</sup>	-	VBAT_IO <sup>1</sup> +0.3	V
VIL	输入低电平	-	-0.3	-	0.3* VBAT_IO <sup>1</sup>	V
VIL_N	引脚 NRESET 的输入低电平	-	-0.3	-	0.2*VBAT	V
VOH	输出高电平	I <sub>max</sub> =-2.5mA	0.9*VBAT_IO <sup>1</sup>	-	VBAT_IO <sup>1</sup>	V
VOL	输出低电平	I <sub>max</sub> =2.5mA	0	-	0.1* VBAT_IO <sup>1</sup>	V
I <sub>leak</sub>	数字输入漏电流 (NSS, MOSI, SCK)		-1	-	1	uA

1. 不包括下列引脚: NRESET 和 DIO3, 它们参考 VBAT。

## 4. 电路描述

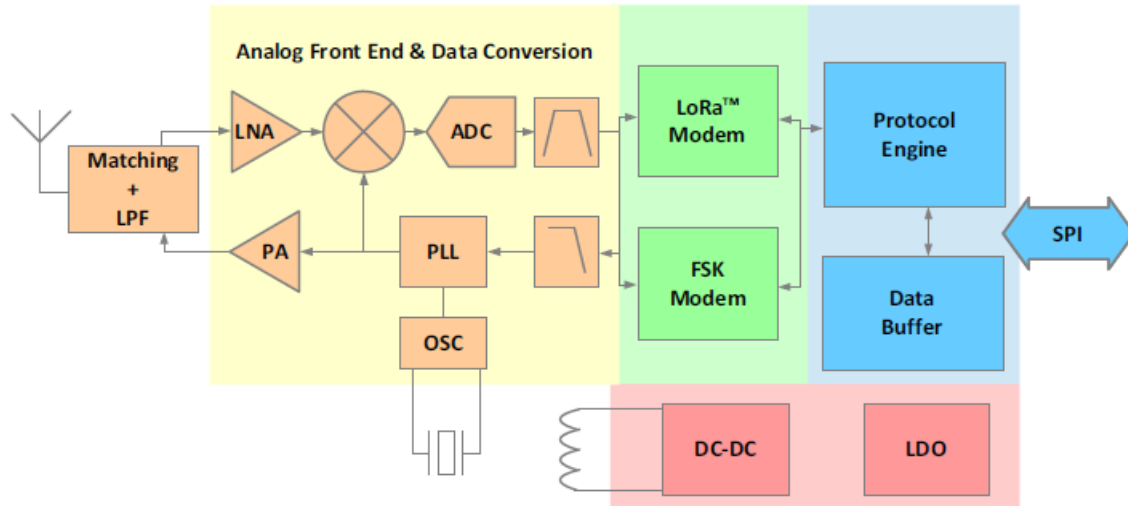


图 4-1: SX1261/2 方框图

SX1261 和 SX1262 是工作在 1GHz 以下频段的半双工射频收发器，能够处理类似 LORA 或者 FSK 的恒包络调制。

### 4.1 参考时钟

#### 4.1.1. RC 基准频率

芯片内有 2 个 RC 振荡器：64KHz 和 13MHz。当收发器在休眠模式下执行周期性的工作时，内部电路选用 64KHz 振荡器（RC64K）来唤醒收发器。有几个命令利用内部 64KHz 振荡器来产生基于时间的事件。内部 13 MHz RC 振荡器（RC13M）用来为 SPI 通信提供时钟从而对器件进行配置，这样不需要启动外部晶体振荡器。两种 RC 振荡器都是由电池直接供电。

#### 4.1.2. 高精度基准频率

在 SX1261/2 中，高精度频率基准有两个来源：一是来自晶体振荡器(OSC)，二也可以来自 TCXO。

SX1261/2 是 4 x 4 mm QFN 封装，而且 SX1262 能输出高达 22dBm 的功率。当在发射模式时，由于输出功率和电流消耗，会使电路发热。因此为了避免把热量传导至外部晶体谐振器，在 PCB 设计时要注意隔离电路和晶体谐振器。

在 LoRa 调制模式下，把 LowDataRateOptimize 配置为 0x00 时(参见 87 页表 13-50：“LoRa® ModParam4 - LowDataRateOptimize”)，应该使包传输时间内的频率漂移降至最小并且低于 Freq\_drift\_max。

$$Freq\_drift\_max = \frac{BW\_L}{3 * 2^{SF}}$$

如果可能，把 LowDataRateOptimize 配置为 0x01，则可以使包传输时间内的频率漂移的要求放松至  $16 \times \text{Freq\_drift\_max}$ 。

**注意:**

在应用文档 AN1200.37 “Recommendations for Best Performance”中，详细描述了在 PCB 设计中的散热技术。

在散热技术不能够使用的紧凑型设计中或不支持 LowDataRateOptimize 功能的应用中，需要使用 TCXO 来提供更稳定的时钟基准。

### 4.1.3. XTAL 控制单元

SX1261/2 不要求用户焊接 32MHz 晶体的外部负载电容。实际上，芯片的 XTA 和 XTB 引脚上都连接有可调节的内部电容。每个电容都可独立配置，步进是 0.47pF。

表 4-1: 引脚内部电容配置

引脚	寄存器地址	典型值
XTA	0X0911	每个电容可以独立控制，步进是 0.47pF: 0X00 设置调节电容到 11.3pF (最小值) 0X2F 设置调节电容到 33.4pF (最大值)
XTB	0X0912	

使用 XTAL 时的注意事项:

在上电复位 (POR) 或者从睡眠模式下冷启动唤醒后，可调电容寄存器的初始化值 0x05 (13.6 pF)。一旦芯片配置为 STDBY\_XOSC 模式，内部状态器将把这两个寄存器值设置为 0x12 (19.7pF)。因此，用户在改变可调电容值前，必须确保芯片已经在 STDBY\_XOSC 模式中，以免寄存器被状态机改写。

使用 TCXO 时的注意事项:

一旦 SetDIO3AsTCXOctrl(...)命令被执行，控制 XTA 引脚上内部电容的寄存器将自动配置为 0x2F (33.4 pF) 以便滤除可能传导到 PLL 的杂散频率。

#### 4.1.4. TCXO 控制框图

在某些情况下，散热不好的小型设计中或者极端温度变化的环境中，需要使用TCXO来获得更好的频率精度。这取决于整个系统，发射机和接收机。在第19页[表3-8：“接收模式规格”](#)中的FERR\_L 指标列出了最大可容许频率偏移量。

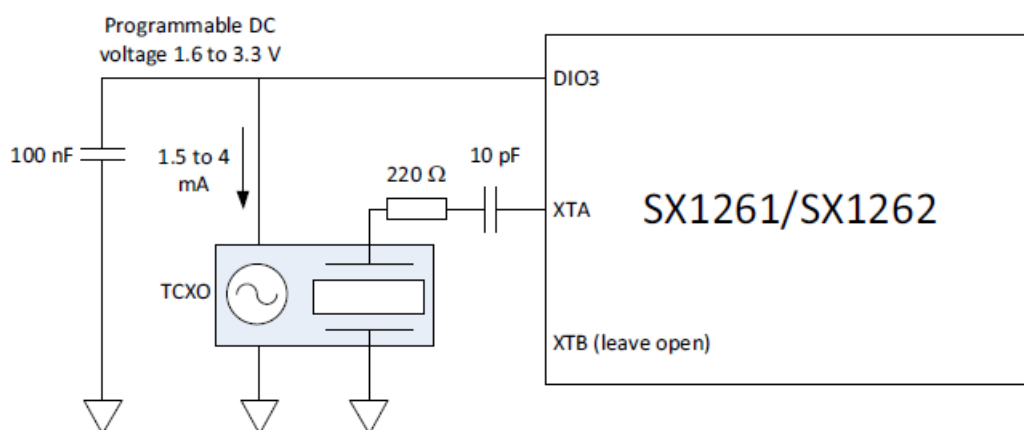


图 4- 2: TCXO 控制图

当使用TCXO时，应该通过一个220ohm电阻和一个10pF的隔直电容连接到引脚3 (XTA)。引脚4 (XTB) 悬空。引脚6 (DIO3) 可以提供稳定的直流电源给TCXO，其电源电压是从1.6V 到 3.3V 可调节的。为了确保正常工作，VBAT应该比设定电压高200mV。

额定电流是1.5mA，但是电源变换器可以提供4mA的电流。TCXO的正弦输出幅度不能超过1.2V峰峰值。启用TCXO模式的命令在第80页[13.3.6节“SetDIO3AsTCXOctrl”](#)中描述，也包括了DC电压和时序信息。

注意：

在芯片已被SetDIO3AsTCXOctrl设置为TCXO模式之后，为了返回到正常的XOSC操作，需要执行一个完整的芯片复位过程，如第49页[第8.1节“Reset”](#)中描述。

## 4.2 锁相环(PLL)

N分数三阶 $\Sigma$ - $\Delta$ 锁相环(PLL)用作接收机和发射机链路的本振(LO)的频率合成器。SX1261 / 2能够连续覆盖所有的1GHz以下频率范围：150 ~ 960MHz。锁相环(PLL)能够自动校准，并且具有低切换或跳频时间。频率调制在PLL带宽内进行。锁相环(PLL)频率来源于使用外部32 MHz晶体基准的晶体振荡器电路。

### 4.3 接收机

收到的射频信号先被一个差分的低噪声放大器放大，然后降频到低中频信号，再经混频器正交得到I/Q信号，该I/Q信号经过低通滤波器后被一个连续反馈型的 $\Sigma$ - $\Delta$  ADC量化，其动态范围大于80dB。一旦进入数字域中，信号被抽取，再次下变频，再次抽取，信道滤波，最后由所选择的解调器解调：FSK调制解调器或者LoRa® 调制解调器。

#### 4.3.1. 中频

SX1261/2 的接收机主要工作在低中频（除了特定的高带宽配置），其具体的带宽配置如下表：

表 4- 2: FSK 模式下的中频频率

带宽配置名	带宽 [kHz 双边带]	中频 [kHz]
RX_BW_467	467.0	250
RX_BW_234	234.3	250
RX_BW_117	117.3	250
RX_BW_58	58.6	250
RX_BW_29	29.3	250
RX_BW_14	14.6	250
RX_BW_7	7.3	250
RX_BW_373	373.6	200
RX_BW_187	187.2	200
RX_BW_93	93.8	200
RX_BW_46	46.9	200
RX_BW_23	23.4	200
RX_BW_11	11.7	200
RX_BW_5	5.8	200
RX_BW_312	312.0	167
RX_BW_156	156.2	167
RX_BW_78	78.2	167
RX_BW_39	39.0	167
RX_BW_19	19.5	167
RX_BW_9	9.7	167
RX_BW_4	4.8	167

表 4- 3: LoRa® 模式下的中频频率

带宽配置名	带宽 [kHz 双边带]	中频 [kHz]
LORA_BW_500	500	0
LORA_BW_250	250	250
LORA_BW_125	125	250
LORA_BW_62	62.5	250
LORA_BW_41	41.67	167
LORA_BW_31	31.25	250
LORA_BW_20	20.83	167
LORA_BW_15	15.63	250
LORA_BW_10	10.42	167
LORA_BW_7	7.81	250

#### 4.4 发射机

发射链路由N-分数锁相环直接调制输出。为了减小泄漏到邻信道的功率，可以选择一个可选的比特流前置滤波器，这也依赖于所选择的调制类型。

SX1261 的最大输出功率是+14/15dBm，SX1262的是+22dBm。输出功率可以在32dB的动态范围内由软件进行配置，步进为1dB；功率放大器Ramping时间也是可配置的，以满足法规要求。

芯片的功率放大器由内部的电压转换器通过 VR\_PA 管脚进行供电，在使用时VR\_PA 和 RFO 在芯片外部直接相连。如4-3的图示：“DC-DC模式下功率放大器供电原理”，两种芯片之间VR\_PA 的供电方式是不同的。

- SX1261：VR\_PA端口, 由VDD\_IN来进行供电，而其电源来自内部的电压转换器(DC-DC 或 LDO)，这样就由于电源电压的变化引起的输出功率的变化非常小。
- SX1262：VR\_PA端口, 由VDD\_IN来进行供电，而其电源直接来自电池；这种情况下，输出的最大功率将受限于VDD\_IN管脚电源电压。

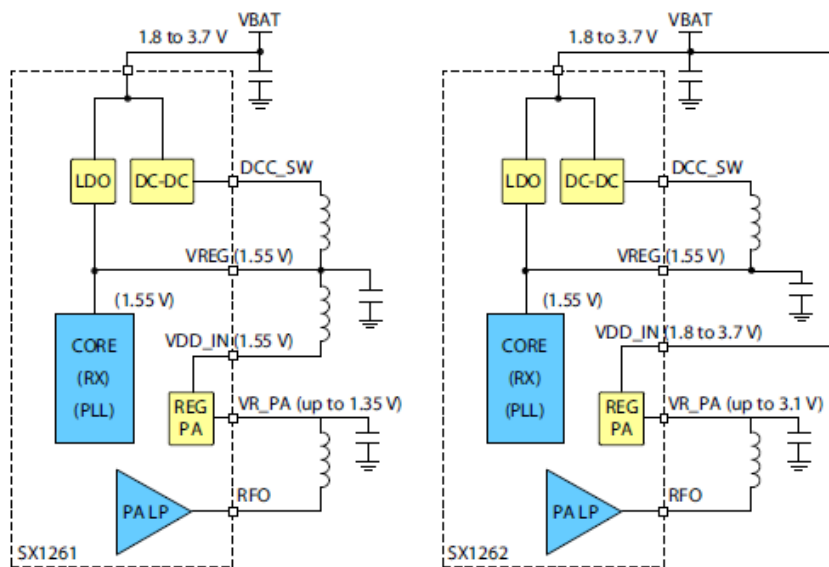


图 4-3: DC-DC 模式下的功率放大器供电原理

#### 4.4.1. SX1261 功率放大器特性

##### 注意!

本章中的所有图表都是指示性的和典型的值，而不是规格。所有图表只突显了功率放大器的电压-电流特征。

对于SX1261芯片，当采用内部DC-DC供电时，发射机的功率效率将会达到最大。其VR\_PA端口的电压会依据软件配置的输出功率在20mV 到 1.35V 内进行变化。

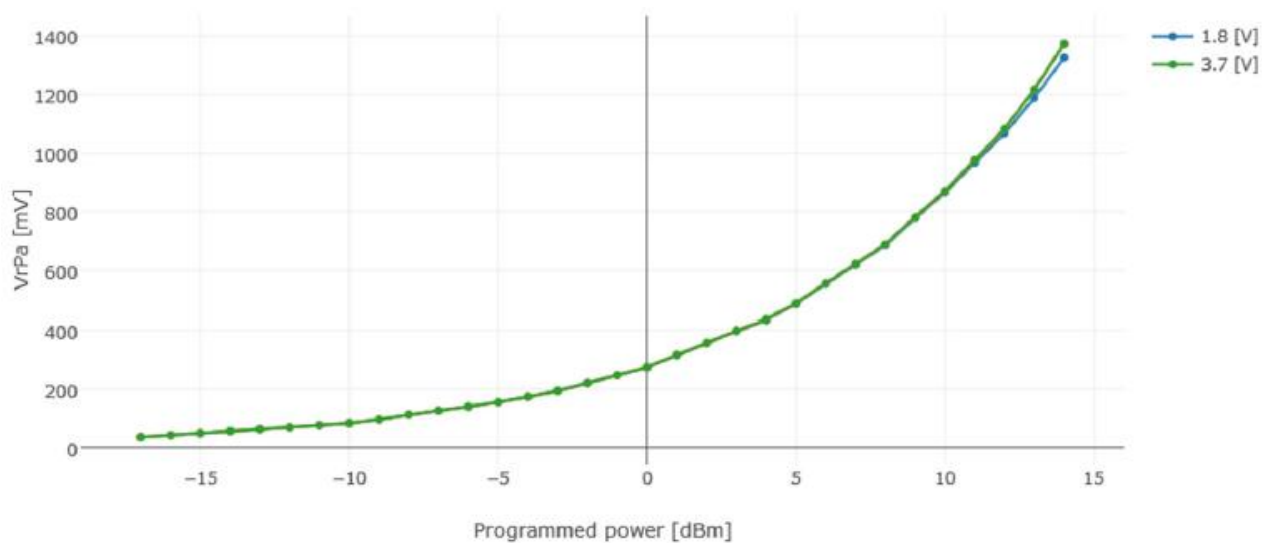


图 4-4: SX1261 的 VR\_PA 电压 vs 输出功率

在 DC-DC 模式下，当电源电压在 1.8V 到 3.7V 范围内变化时，输出功率几乎保持恒定。

在 DC-DC 模式下，总功耗将直接受供电电压的影响。例如，供电电压为 3.7V 时，10dBm 的输出功率需要消耗电流 17mA；同样输出功率的情况下，供电电压在 1.8V 时，需要消耗电流 34mA。

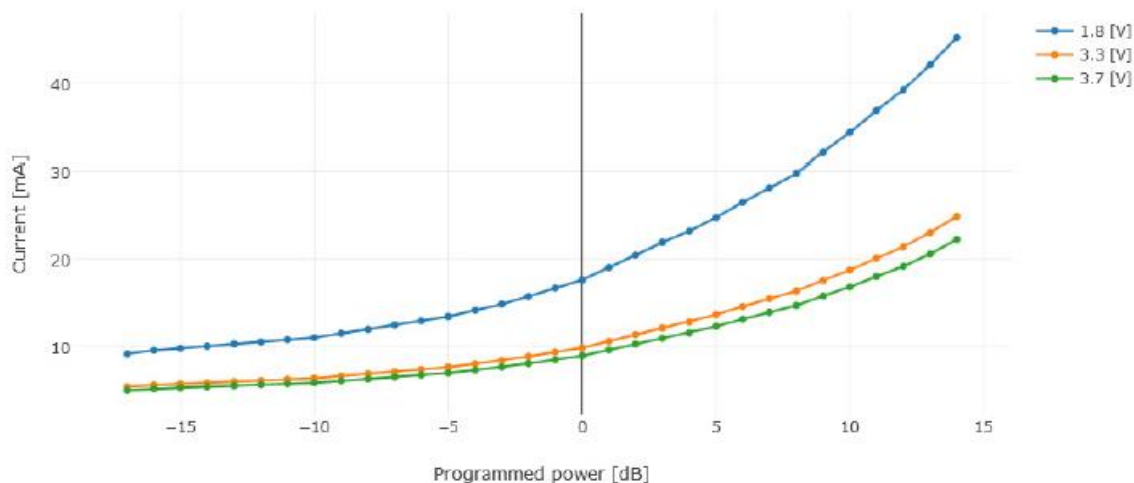


图 4-5: SX1261 在 DC-DC 模式下的电流 vs 输出功率

然而，当选择 LDO 模式，并且输出功率相同的条件下，当供电电压在 1.8V 到 3.7V 范围内变化时，消耗电流将保持不变，但是降低了电源效率。

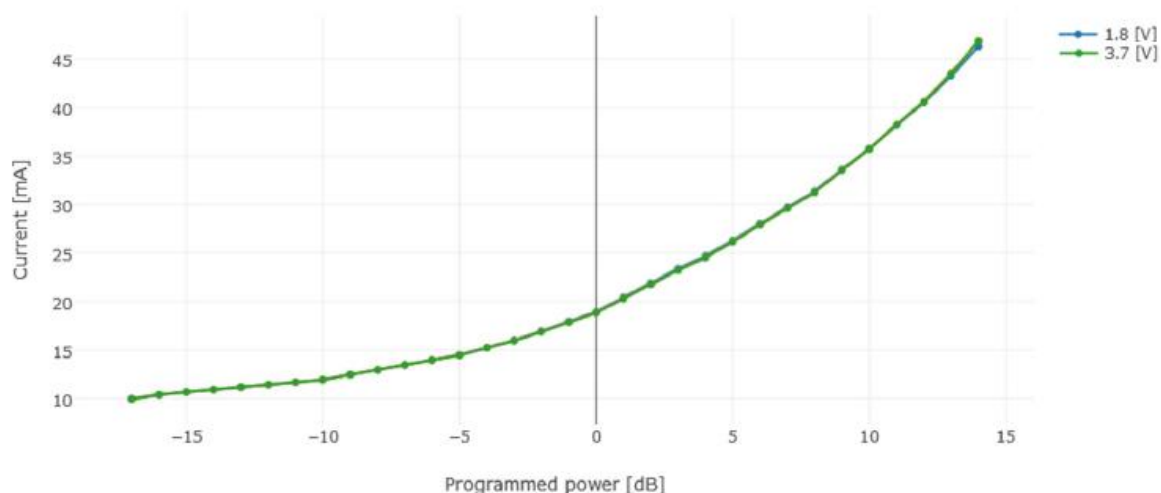


图 4-6: SX1261 在 LDO 模式下的电流 vs 输出功率

下图所示在额定电压和极端电压条件下输出功率的线性度：

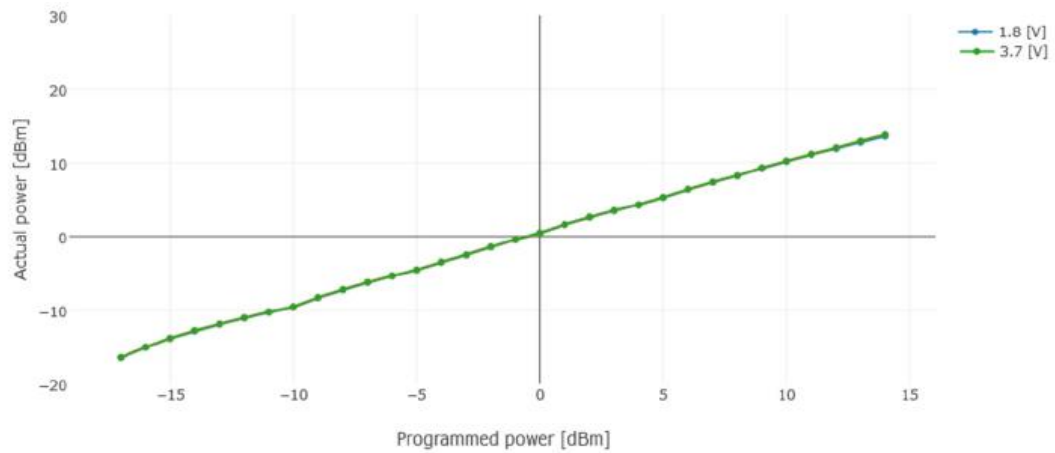


图 4- 7: SX1261 在 LDO 或 DC-DC 模式下的输出功率线性度

#### 4.4.2. SX1262 功率放大器特性

##### 注意!

所有以下图表都是指示性的和典型的值，而不是规格。所有图表只突显了功率放大器的电压-电流特征。

下列图表基于SX1262工作在DC-DC模式，此时DC-DC只负责为内核供电。

对于SX1262，功率放大器优化了最大输出功率，同时最大限度的提高了效率，这导致如果要使SX1262保持较高的输出功率，就必须为其功率放大器提供相当高的电压。总结如下：

- 当输出功率最大时，功率放大器的电流效率最高；
- 输出功率受限于供电电压；

如下图表所示：

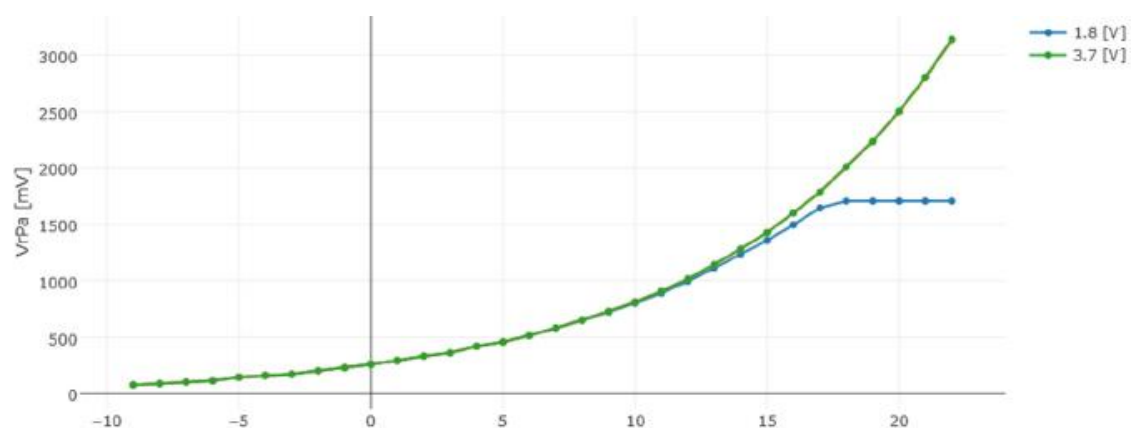


图 4- 8: SX1262 的 VR\_PA 电压 VS 输出功率

芯片内部的 VR\_PA 电压变换器有一个小于 200mV 的压降，这意味着供电电压必须比 VR\_PA 的输出电压至少高出 200mV 才能满足相应的输出功率要求。例如，设置输出功率为+20dBm 时，需要的

VR\_PA 输出电压为 2.5V，那么此时的供电电压必须在 2.7V 到 3.7V 的范围内，才能保证 SX1262 输出 +20 dBm 的功率。当电压低于 2.7V，此时的输出功率将会随着供电电压的降低而降低。

如图 4-8：“SX1262 的 VR\_PA 电压 vs 输出功率”中蓝色图线所示，当供电电压在 1.8V 时，VR\_PA 可以提供 1.7V 的输出电压；对于所有高于 17dBm 的功率设置，这个电压将使输出功率稳定在 +17dBm。

只要供电电压足够高以满足 VR\_PA 所需求的电压，则输出功率是线性的，下图确认了输出功率的线性度。

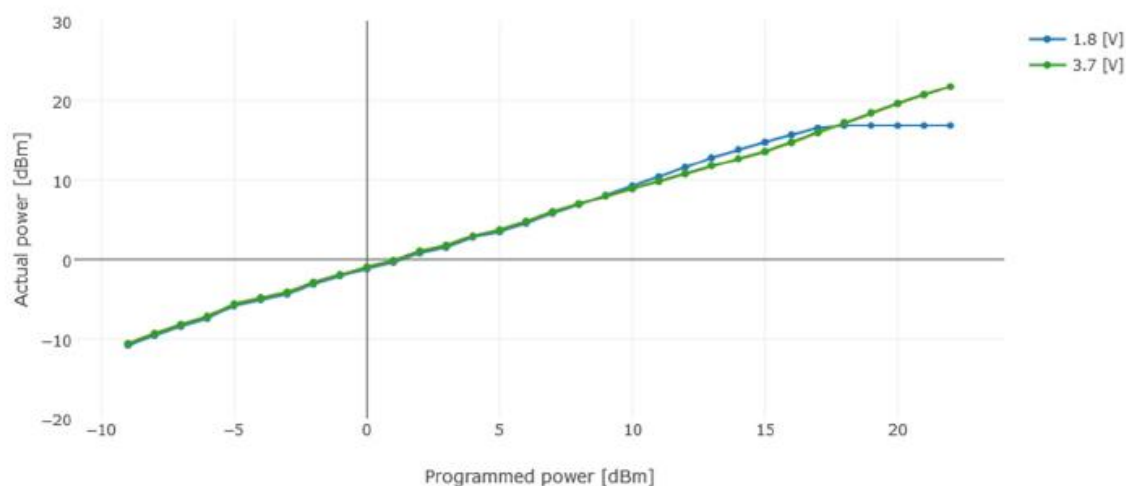


图 4-9: SX1262 的输出功率线性度

输出功率与对应功耗的关系如下图所示（DC-DC 模式）：

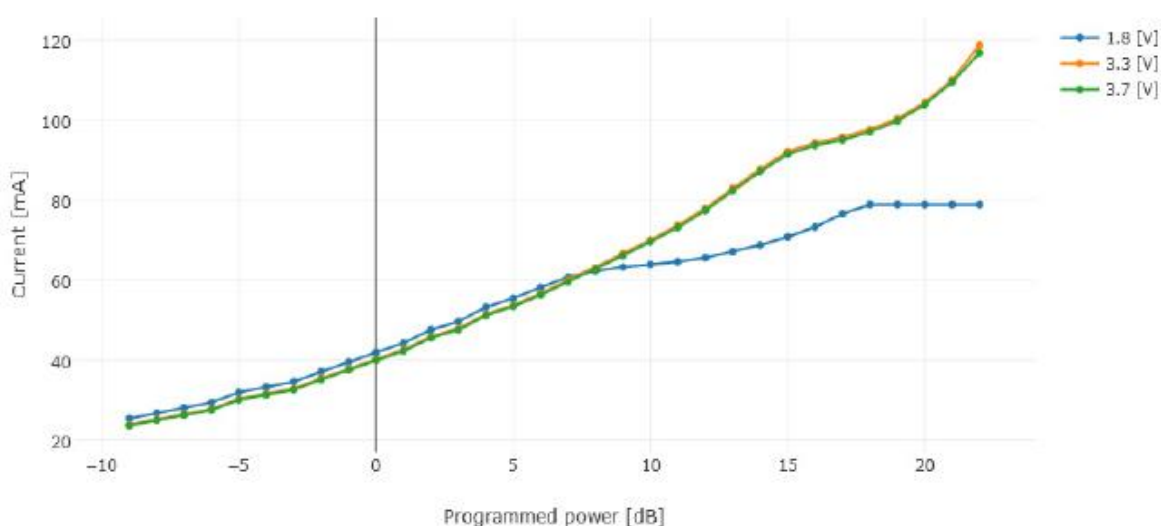


图 4-10: SX1262 的输出功率 vs 电流

#### 4.4.3. 功率放大器概要

下列表格汇总了 SX1261 和 SX1262 收发器在功率放大器最优情况下的相关数据：

表 4- 4: 功率放大器参数汇总

功率放大器概要	条件	SX1261	SX1262
最大输出功率	相应的阻抗匹配 和参数配置	+14 /15 dBm	+22 dBm
发射电流	+22 dBm +14 dBm	- 25.5 mA	118 mA 90/45 <sup>1</sup> mA
输出功率 vs 电压		电压 : 1.8 V~3.7 V	电压 : 3.3 V~3.7 V +22 dBm@3.1V +20 dBm@2.7V +16 dBm@1.8V
发射电流 vs 电压		与电压成反比 用 DC-DC 模式为功率放大器供电	-

1. 详见 第75页[13.1.14.1节 “PA最优设置”](#)。

## 5. 电源分配

### 5.1 选择 DC-DC 或 LDO

根据应用的优先级要求，有两种形式的电源变换器可以选择（DC-DC 或 LDO）。其中，LDO存在于所有模式中，但DC-DC只在被选择后才可使用。DC-DC电源可以在FS、RX和TX 模式中使用。

DC-DC电源可以由两种时钟驱动：

- 在STDBY\_XOSC模式下：内部的一个13MHz的RC时钟作为其时钟源，频率为13MHz/4，所以DC-DC的开关频率是3.25MHz。
- 在FS、RX、TX模式下：DC-DC的时钟源由内部的锁相环提供，其频率是~5MHz；每次调用命令 *SetRFFrequency(...)* 时，会重新计算分频比，使得开关频率尽可能接近5MHz。

除非特殊说明，收发器的所有规格都是基于DC-DC使能的状态。对于成本和产品尺寸受限的应用，可以使用LDO模式供电。相较于DC-DC，其不需要Pin\_1的47nH电感，同时不需要在Pin\_7 和Pin\_9之间加入15uH电感，这样可以降低BOM成本，同时减小板子的尺寸。下列的表格表述了不同模式和配置下的电源使用情况。

表 5-1: 电源类型 vs 电路模式

电路模式	Sleep	STDBY_RC	STDBY_XOSC	FS	RX	TX
Type= 0	-	LDO	LDO	LDO	LDO	LDO
Type = 1	-	LDO	DC-DC+LDO	DC-DC+LDO	DC-DC+LDO	DC-DC+LDO

用户可以通过命令 *SetRegulatorMode(...)* 指定使用 DC-DC，该操作必须在 STDBY\_RC 模式下被执行。

当启用 DC-DC 时，LDO 将保持打开，其电压将被设置在低于 DC-DC 输出电压 50mV 的地方，以确保在大电流峰值时电压保持恒定。如果由于大电流峰值导致 DC-DC 输出电压跌出此范围，LDO 将会作为补充对电流予以弥补，但是此时能耗将增加。

然而，为了避免过多的能量消耗，用户可以通过手动设置过流保护来加以限制。每次复位以后，该过流保护门限默认在 60mA。

表 5-2: 过流保护配置

芯片类型	寄存器地址	过流保护默认值	最大电流
SX1261	0x08E7	0x18	60 mA
SX1262	0x08E7	0x38	140 mA

过流保护设置的步进为 2.5mA，每次调用命令 *SetPaConfig(...)* 时，默认值会被自动重新配置。如果客户想修改该过流保护参数，可以在执行命令 *SetPaConfig(...)* 后的再进行配置。

**注意:**

当用户在使用 SX1261 的 DC-DC 模式时，应当注意电压与电流的所属关系。由于电压与电流是成反比的关系（例如当输出功率为 +14dBm 时，3.3V 时电流为 25.5mA，而 1.8V 时电流则为 48mA），因此过流保护的门限应该设置的足够高以便适应电流的增加或者动态的配置它。

另一种方法是将过流保护设置在一个特定的门限，当过流保护开始限制当前的电流消耗时，接受该器件的输出功率的下降。

### 5.1.1. 选项 A: SX1261/ 使用 DC-DC 转换器

使用 DC-DC 转换器时，对于芯片内核和功率放大器来说，可以达到 90% 的效率，

**优点：**

当电压为 3.3V 时的功耗值将大大减少，  
当电压在 1.8V 到 3.7V 之间变化时，输出功率保持不变。

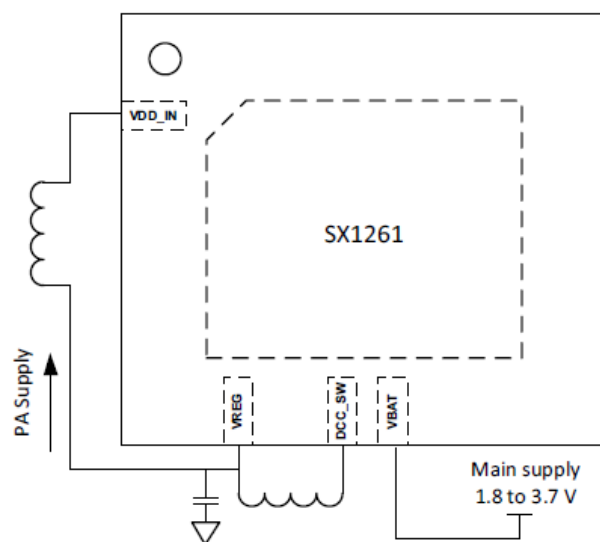


图 5-1: SX1261 使用 DC-DC 下的拓扑

### 5.1.2. 选项 B: SX1261/ 使用 LDO 转换器

LDO转换器用于为芯片内核和功率放大器的供电。

优点：

没有外置的15uH 和 47nH电感， BOM成本下降，尺寸减小。

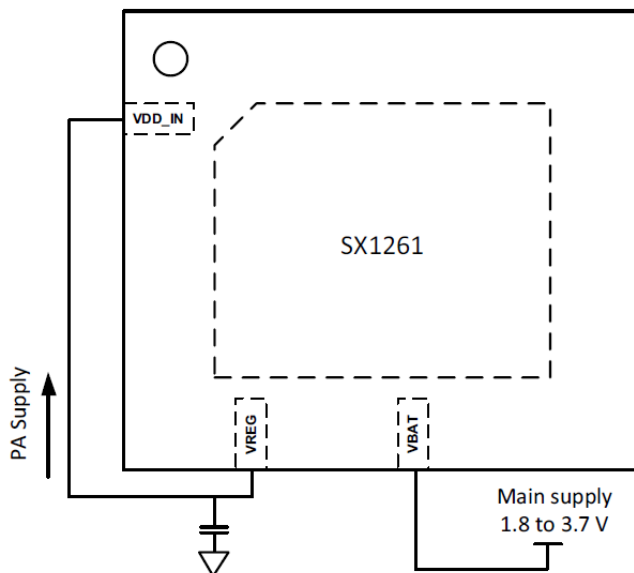


图 5- 2: SX1261 使用 LDO 下的拓扑

### 5.1.3. 选项 C: SX1262 /使用 DC-DC 转换器

DC-DC转换器效率可以达到90%，其只对芯片的内核供电，功率放大器采用外部电源供电。

优点：

内核功耗下降。

\*输出+22dBm，VBAT 最小值为 3.3V

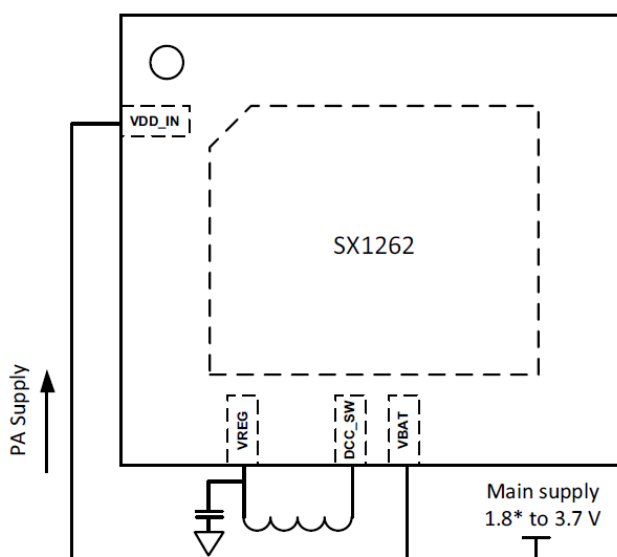


图 5- 3: SX1262 使用 DC-DC 下的拓扑

#### 5.1.4. 选项 D: SX1262 使用 LDO 转换器

使用LDO转换器，该种用法下内核的功耗较选项C的略微偏高。

优点：

没有外置的15uH电感，BOM成本下降，尺寸减小。

\* 输出+22dBm，VBAT 最小值为 3.3V

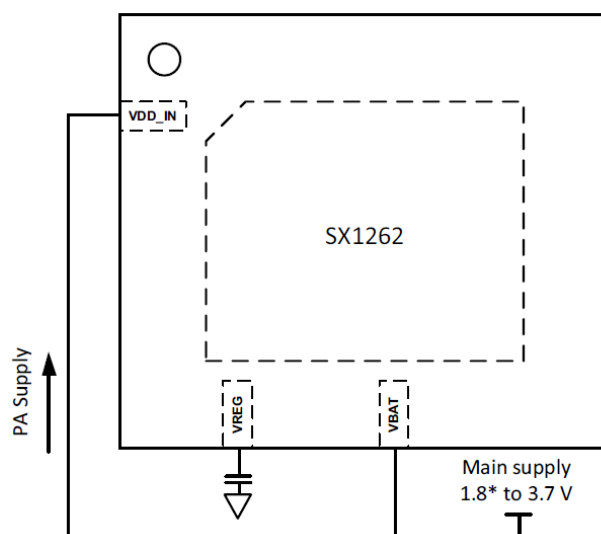


图 5-4: SX1262 使用 LDO 下的拓扑

#### 5.1.5. DC-DC 模式下电感的选型

对于电感的选择，其关键是确保DC-DC工作在最佳状态，电感选用不恰当将会产生各种不良的影响，从DC-DC输出的纹波电流，到器件的老化，以及DC-DC转换器效率的恶化。

对于SX1261/2，首选的电感是其本身带有屏蔽性能，具有低的内部串联电阻和比DC-DC开关频率高很多的谐振频率。当选用15uH电感时，以下几个方面应该被考虑：

- 直流电阻 (最大值) = 2 ohms
- 直流电流 (最小值) = 100 mA
- 频率 (最小值) = 20 MHz

表 5-3: 典型的 15 uH 电感

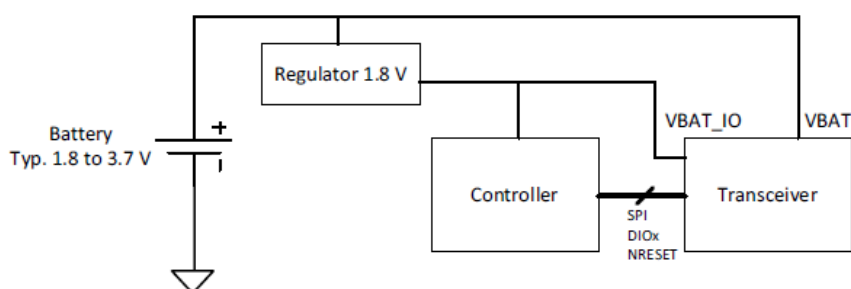
型号	生产厂家	电感量 ( $\mu\text{H}$ )	直流电流 (最大) (mA)	频率 (MHz)	直流电阻 (ohm)	封装形式 (L x W x H in mm)
LPS3010-153	Coilcraft	15	370	43	0.95	2.95 x 2.95 x 0.9
MLZ2012N150L	TDK	15	90	40	0.47	2 x 1.25 x 1.25
MLZ2012M150W	TDK	15	120	40	0.95	2 x 1.25 x 1.25
VLS2010ET-150M	TDK	15	440	40	1.476	2 x 2 x 1
VLS2012ET-150M	TDK	15	440	40	1.062	2 x 2 x 1.2

## 5.2 灵活的 DIO 供电

该芯片有两种电源引脚，一种名为VBAT的是用于芯片内核供电的，另一种名为VBAT\_IO的是用于主机控制接口（SPI,DIO,BUSY）的，使用时两种电源管脚可以直接相连。如果低电压的微控制器（IO口典型电压为1.8V）用于控制该芯片，用户可以如下操作：

- VBAT使用3.3V 电源以便射频达到最佳性能
- VBAT\_IO电源引脚与微控制器的电源引脚直接相连
- 芯片的数字IO引脚与微控制器的IO引脚直接相连

任何时候，VBAT\_IO 的电压必须不高于 VBAT 的电压。



要求:  $VBAT \geq VBAT\_IO$

图 5- 5: 分离的 DIO 供电

## 6. 调制解调器

SX1261/2包含两种不同类型的调制解调器，分别处理LoRa和FSK调制。两种调制类型都有各自的帧结构和调制解调器。

- LoRa® 调制解调器 ↔ LoRa® 帧结构
- FSK 调制解调器 ↔ FSK 帧结构

用户通过命令 *SetPacketType(...)* 来选择调制解调器类型和使用的帧结构，该命令指定了所需要的帧结构，以及调制解调器类型。

在选择调制和包参数，以及在进入RX或TX之前，该函数要首先被调用。命令 *GetPacketType()* 用于返回芯片当前的协议类型。

### 6.1 LoRa®调制解调器

与传统的FSK调制相比，LoRa®调制使用扩频调制和前向纠错的技术来增加信号的覆盖范围和无线通信链路的鲁棒性。

LoRa®调制解调器的一个重要特性就是增强了抗干扰能力。其可以对相同信道的GMSK信号达到19dB的抑制度。无论是在频谱被严重占用的频段上，还是在传统的调制方案失败的时候使用LoRa®扩展的混合通信网络中，该抗干扰性能允许LoRa®调制系统的简单共存。

#### 6.1.1. 调制参数

对于一个给定的应用，在LoRa®调制方式下，可以通过给出的4个参数对其性能进行优化，以便在链路预算，抗干扰性能，频谱占用和传输速率之间做出平衡。这些参数如下：

- 调制带宽 (BW\_L)
- 扩频因子 (SF)
- 编码率 (CR)
- 低速率优化 (LDRO)

这些参数通过命令 *SetModulationParams(...)* 进行设置，在确定协议之后该操作必须被执行。

#### 扩频因子

LoRa®扩频调制是通过把有效载荷信息中每比特数据用多位码片信息表示来实现的。发送扩展信息的速率称为码元速率 (Rs)，该符号速率和码片速率的比值就是扩频因子，它表示每比特信息发送的码元数。

#### SF5 and SF6 的差异

较之早前版本的芯片，在SX1261/2里增加了两个新的扩频因子：SF5和SF6。这两个新增加的扩频因子已经在SX1261/2中被稍微改进，以便在隐式和显式两种包头模式下可以被使用。然而这些修改导致了新增的扩频因子和前一版本的芯片不兼容，尤其是SX1262中的SF6不能后向兼容SX1276中的SF6。

此外，由于码元速率较高，相比于其他的扩频因子，推荐的最小前导码的长度有所增加，以便确保该前导码从接收信号中被正确的侦测和解调。对于SF5和SF6，建议用户使用12个码元的前导码长度以在动态范围或接收机上具有最佳性能。

#### 注意:

由于不同的扩频因子之间两两正交，所以在一组收发链路中，扩频因子必须提前预知。除此之外，还要注意在接收机输入端所需的信噪比（SNR）。

在LoRa® 接收机中，由于能够接收负信噪比的信号，因此增加了接收机灵敏度，链路预算以及覆盖距离。

表 6-1: 扩频因子的范围

扩频因子 (SF)	5	6	7	8	9	10	11	12
$2^{\wedge}SF$ (码片/ 码元)	32	64	128	256	512	1024	2048	4096
典型的 LoRa 解调器的 SNR [dB]	-2.5	-5	-7.5	-10	-12.5	-15	-17.5	-20

较高的扩频因子提供更好的接收机灵敏度，但是以更长的空中传输时间为代价。

#### 6.1.1.2 带宽

信号带宽的增加允许使用更高的有效载荷速率，因此在减小了发射时间的同时也带来的接收机灵敏度的下降。

LoRa® 模式下，带宽 (BW\_L) 可以用软件设置，位于中心频率（f<sub>RF</sub>）附近。

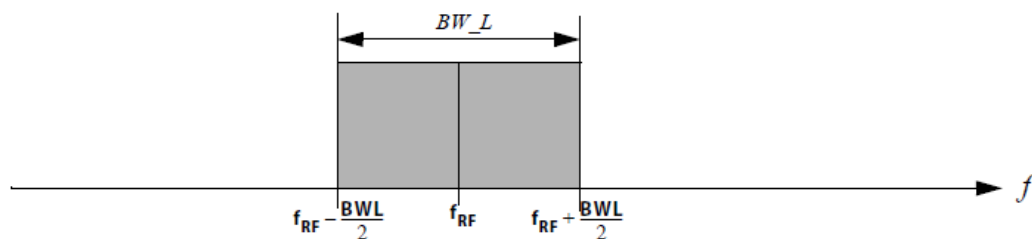


图 6-1: LoRa® 信号带宽

信号带宽的增加允许使用更高的有效载荷速率，因此在减小了发射时间的同时也带来的接收机灵敏度的下降。很多国家对占用带宽都有限制。LoRa® 调制带宽指的是双边带带宽。其可用的带宽范围如下表所示：

表 6-2: LoRa® 模式下的信号带宽设置

信号带宽	0	1	2	3	4	5	6	7	8	9
BW_L [kHz]	7.81	10.42	15.63	20.83	31.25	41.67	62.5	125	250 <sup>1</sup>	500 <sup>1</sup>

1. 对于400MHz以下的无线电频率，在频率和带宽之间有一个比例关系，部分带宽不适用于400MHz 以下的频段。

对于小于或等于250KHz的带宽(BW\_L)，接收机需要做二次变频。第一次在射频电路内变频到低中频，第二次变频到基带频率，以便基带用作解调。在RF链路内执行第一次下变频到低中频，在基带调制解调器内部进行第二次变频到基带。当使用500KHz带宽时，在RF链路内一次性直接变频至零中频（zero-IF）。

### 6.1.1.3 前向纠错编码率

为了进一步提高链路的鲁棒性，LoRa调制解调器采用循环误差编码来执行前向误差检测和校正。

前向纠错(FEC)对于提高存在干扰的链路的可靠性特别有效。因此，可以改变编码速率和抗干扰的鲁棒性来应对不同的信道条件。在发射机端选择的编码速率通过包头(当存在时)传递给接收机。

表 6-3 : 编码率的包头

编码率	循环编码的 CR [有用位数 / 总位数]	开销率
1	4/5	1.25
2	4/6	1.5
3	4/7	1.75
4	4/8	2

更高的编码率提供了更好的抗扰性，但需要更长的传输时间。在正常情况下，4/5 的因子提供了最佳的折中；在强干扰的情况下，可以使用更高的编码率。错误校正码不需要被接收器预先知道，因为它被编码在包头部分。

### 6.1.1.4 低数据率优化

对于低的数据率(通常是高的扩频因子或低的带宽)和一个需要在空中持续传输几秒钟的有效载荷，此时可以打开低速率优化(LDRO)。这样每个码元对应的比特位数减少为SF-2 (详见第 41页 [6.1.4 节 "LoRa® 空中时间"](#))，以便接收机更好地追踪 LoRa 信号。根据有效载荷大小，当LoRa的码元时间等于或大于16.38ms时推荐使用低数据率优化功能。

### 6.1.1.5 LoRa® 发射的参数关系

根据所知的关键参数，LoRa®码元率被定义为：

$$R_s = \frac{BW}{2^{SF}}$$

其中BW为带宽，SF为扩频因子。传输的信号是一个恒定包络的信号。等效于每1Hz带宽上每秒发送一个码片。

## 6.1.2. LoRa® 包引擎

LoRa® 有自己的包引擎, 支持 LoRa®如以下部分所描述的物理层。

### 6.1.3. LoRa® 帧结构

LoRa®调制解调器使用两种类型的数据包格式：显性和隐性。显性包包含一个短消息头，其包含了字节数、编码率和包中是否使用了CRC的信息。其包结构的格式如下图所示。



图 6- 2: LoRa® 包格式

LoRa®数据包开始于一个前导序列，该序列用于使接收机与输入信号同步。默认的情况下，该前导通常被配置为一个包含 12 个码元长的序列。这是一个可变量，因此可以扩展前导码长度，例如，为了减少接收密集型应用中的接收机占空比。一旦考虑了固定的前导码元（4 个码元），所传输的前导码长度可以从 10 到 65535 个码元数不等。

接收机会周期性地重启前导检测操作。出于这个原因，前导序列长度应该配置为与发射机前导序列相同的长度。如果不知道前导序列长度或者其是可变的情况下，应该在接收端将前导序列的长度设置为最大。前导序列后面是一个包头，其包含了后续的有效载荷的信息。有效载荷是可变长度字段，包含以编码率编码的实际数据，其编码率或者以显式模式在包头中指定，或者由用户在隐式模式下指定。也可以附加一个可选的 CRC。

根据所选的操作模式，可以使用两种类型的包头。

#### 6.1.3.1 显性包头模式

这是默认的操作模式。这时包头提供了有效载荷的信息，即：

- 以字节计算的有效载荷长度；
- 前向纠错编码率；
- 对有效载荷进行校验的可选的16位CRC；

包头以最大纠错码(4/8)传输。同时它也有自己的校验码，以便允许接收器丢弃无效的包头。

#### 6.1.3.2 隐性包头模式

在某些情况下，有效载荷、编码率和 CRC 是固定的或预先知道的，这时可以通过使用隐性包头模式来减少传输时间。该模式中包头被删除。此时在收发两端中，有效载荷的长度、错误编码率和有效载荷校验 CRC 需要手动的配置成相同的参数。

#### 6.1.4. LoRa® 空中时间

图6-3、图6-4中详细描述了LoRa® 模式的包格式，“图6-3：定长包格式”和“图6-4:可变长度包格式”。空中时间(ToA)可以通过以下的方程式获得：

$$ToA = \frac{2^{SF}}{BW} * N_{symbol}$$

- SF: 扩频因子 (5~12)
- BW: 带宽 (kHz)
- ToA: 空中时间 (ms)
- N<sub>symbol</sub>: 码元数

由于调制参数的不同，码元数的计算是不同的。

对于 SF5 和 SF6:

$$N_{symbol} = N_{symbol\_preamble} + 6.25 + 8 + \text{ceil} \left( \frac{\max(8 * N_{byte\_payload} + N_{bit\_CRC} - 4 * SF + N_{symbol\_header}, 0)}{4 * SF} \right) * (CR + 4)$$

对于其它的 SF:

$$N_{symbol} = N_{symbol\_preamble} + 4.25 + 8 + \text{ceil} \left( \frac{\max(8 * N_{byte\_payload} + N_{bit\_CRC} - 4 * SF + 8 + N_{symbol\_header}, 0)}{4 * SF} \right) * (CR + 4)$$

对低速率优化开启模式下的其它 SF:

$$N_{symbol} = N_{symbol\_preamble} + 4.25 + 8 + \text{ceil} \left( \frac{\max(8 * N_{byte\_payload} + N_{bit\_CRC} - 4 * SF + 8 + N_{symbol\_header}, 0)}{4 * (SF - 2)} \right) * (CR + 4)$$

- CRC 开启时N\_bit\_CRC = 16, 反之为0
- 显性包头模式时N\_symbol\_header = 20, 隐形包头模式时为0
- CR为 1, 2, 3 或 4, 分别对应于编码率 4/5, 4/6, 4/7 或 4/8

#### 6.1.5. LoRa® 信道状态检测 (CAD)

对于扩频调制技术的应用，由于有用信号可能位于接收机的噪声之下，使得确定信道是否已经被占用面临挑战。在这种情况下使用接收信号强度指示 (RSSI) 显然是行不通的。为了达到这一目的，信道状态检测功能 (CAD) 被用来检测其它 LoRa® 信号是否存在。

在芯片SX1261/2中，信道状态检测是通过检测LoRa®的前导或数据码元来实现的，而前一代产品仅能通过检测LoRa®的前导来实现该功能。

在CAD模式中，芯片SX1261/2通过在用户选择的时间（由码元的数量确定）内扫描整个带宽，通过返回的信道状态检测IRQ来判断通道内是否有 LoRa® 信号。

信道状态检测所需要的时间取决于 LoRa® 调制的参数设置。对于给定的配置 (SF/BW), 典型的CAD检测时间可以选择为1、2、4、8或16个码元。一旦选定的码元数的持续时间完成，接收机将在Rx中再保持半个码元时间用于后续的测量和处理。

## 6.2 FSK 调制解调器

### 6.2.1. 调制参数

FSK 调制解调器能够对速率从 0.6 kbps 到 300kbps 不等的 2-FSK调制的数据包进行传输和接收，所有参数都是通过命令 *SetModulationParams(...)* 来设置的。该功能在调制类型确定后才能调用。

比特率的设置涉及到晶体振荡器，在此提供了一个精确的计算射频信号比特率(或等效的码片率)的方法。在命令 *SetModulationParams(...)* 中，比特率(BR)被表示为32倍的晶体振荡器频率除以芯片使用的实际比特率(BitRate)的形式。公式如下：

$$BR = \frac{F_{XOSC} * 32}{BitRate}$$

FSK 调制是在 PLL 的带宽内进行的，通过改变 PLL 反馈回路中的分数分频比加以实现。一个高分辨率的  $\Sigma$ - $\Delta$  调制器允许非常窄的频率偏移。频率偏移 *Fdev* 是函数 *SetModulationParams(...)* 的参数之一，表示为：

$$Fdev = \frac{FdevHz}{FreqStep}$$

其中：

$$FreqStep = \frac{XtalFreq}{2^{25}}$$

另外，在发射模式下，在包模式或连续模式中，几个成型滤波器可以被用于信号处理中。在接收模式下，用户需要根据自身的参数选择最佳接收带宽。为了保证正确的解调，在选择带宽时必须遵守以下限制：

$$(2 * Fdev + BR) < BW$$

参数 *BW* 定义的带宽数据如下表：

表 6- 4: FSK 包类型下的带宽定义

带宽	寄存器值	带宽 [kHz 双边带]
BW4	0x1F	4.8
BW5	0x17	5.8
BW7	0x0F	7.3
BW9	0x1E	9.7
BW11	0x16	11.7
BW14	0x0E	14.6
BW19	0x1D	19.5
BW23	0x15	23.4
BW29	0x0D	29.3

表 6-4: FSK 包类型下的带宽定义

带宽	寄存器值	带宽 [kHz 双边带]
BW39	0x1C	39.0
BW46	0x14	46.9

BW58	0x0C	58.6
BW78	0x1B	78.2
BW93	0x13	93.8
BW117	0x0B	117.3
BW156	0x1A	156.2
BW187	0x12	187.2
BW234	0x0A	234.3
BW312	0x19	312.0
BW373	0x11	373.6
BW467	0x09	467.0

带宽的选择必须满足：

$$\text{Bandwidth[双边带]} \geq BR + 2 * \text{frequency deviation} + \text{frequency error}$$

其中，频率误差是所使用的晶体频率误差的 2 倍。

SX1261/2 通过参数 *PulseShape* 提供了几个脉冲成型选项。如果将未指定的值赋予该参数，则不使用脉冲成型滤波功能。

## 6.2.2. FSK 包引擎

SX1261/2 是基于包的传输而设计的。包控制器负责将接收到的数据位流打包拼装成数据包，并将其存储到数据缓冲区中。它还执行数据位流的解码操作，比如去白化处理以及对接收到的数据位流进行循环冗余检验（CRC）。

在发射端，包控制器通过构造一个包并将其按位发送到调制器以便发送，同时它可以对有效载荷进行白化处理，并在包的结尾添加校验和。包控制器只能工作在半双工模式，即同一时刻能且只能进行发送或接收操作。

包控制器通过命令 *SetPacketParams(...)* 进行配置，详见第87页[13.4.6节 "SetPacketParams"](#)。该功能只能在调制类型已经确定的情况下调用，下一章节将会对SX1261/2的帧结构差异进行详细讲解。

### 6.2.2.1 接收端的前导检测

如果检测到一个不够数量的前导序列（0x55或0xAA），SX1261/2可以放弃接收该数据包。用户可以通过命令 *SetPacketParams(...)* 中的参数 *PreambleDetectorLength* 来设置此功能。该参数的范围从“PreambleDetectorLength off”到“PreambleDetectorLength 32 bits”，当参数是“PreambleDetectorLength off”时，芯片不对检测到的前导长度作判断而是直接锁定接下来的同步字；当参数是“Preamble detector length 32 bits”时，芯片要求收到32位前导数据。这种情况下，如果没有检测到32位前导码，则芯片在RxSingle模式下放弃接收，或者在RX连续模式下进入下一轮检测。

为了获得最佳性能，建议将前导序列检测器长度设置为“Preamble detector length 8 bits”或“Preamble detector length 16 bits”，而具体的参数取决于发射机所发送的前导长度。

**注意:** 在所有情况下，PreambleDetectorLength必须小于后续的同步字长度以实现正确检测数据包。如果前导序列的长度大于后续的同步字长度（典型的情况是没有同步字），此时用户需要用0x55 作为同步字。

### 6.2.3. FSK 包格式

FSK包格式采用了一种常规的包格式，适用于专有的NRZ编码，低能耗的通信链路。在该包格式中，内嵌了有效载荷的CRC校验，动态的有效载荷长度和包确认。可选的白化功能。在FSK 协议中，提供了两种格式的可用包结构：固定包长和可变包长。

#### 6.2.3.1 固定包长

如果包长度是固定的而且通信双方都已知，那么包长度的信息就不用在空中传输。相反，包长度信息可以通过参数`packetLength`来配置，包长度以字节数表示（0 到 255）。

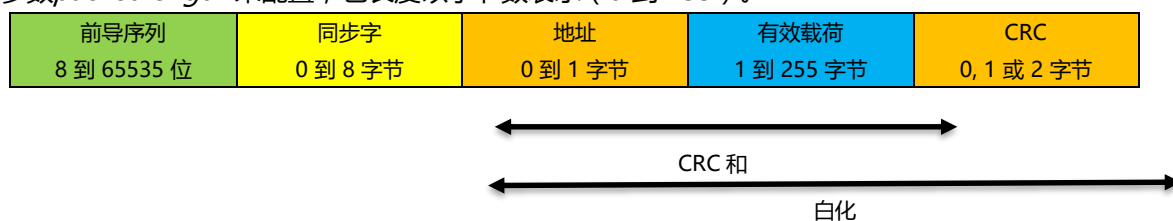


图 6-3: 定包长格式

通过参数 `PreambleLen` 可配置前导序列的长度，该长度可从 8 位 到 65535 位。**通常建议至少使用 16 位的前导序列长度以确保接收端对数据的有效接收。** CRC 的操作，包长度以及前导序列长度都可以通过命令 `SetPacketParams(...)` 加以配置，详见第 61 页 [第 11 节的“命令列表”](#)。

#### 6.2.3.2 可变包长

在包是不确定的或者是可变长度的情况下，包的长度信息必须包含在包本身信息中加以传输。可变长包的格式如下图所示。

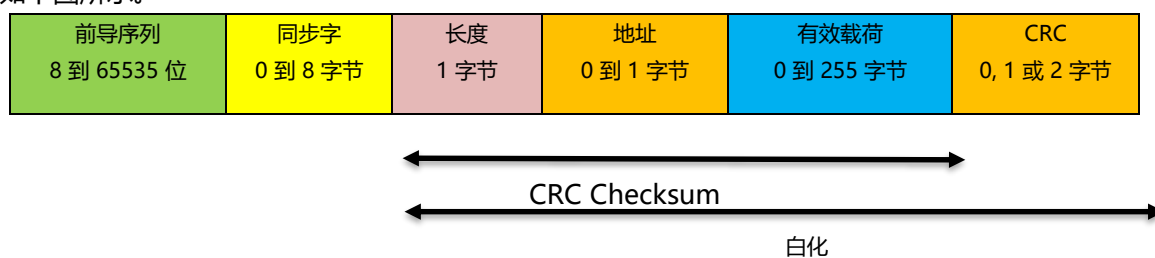


图 6-4: 可变包长格式

### 6.2.3.3 设置包长度或终端地址

包长度信息和终端或广播地址通常不被认为是有效载荷的一部分，它们由硬件自动添加。

当使用命令 `SetPacketParam(...)` 将包类型设置为可变长度时，包长度信息自动添加到包本身信息中。

可以在命令 `SetPacketParam(...)` 中通过字段 `AddrComp` 对终端或广播地址进行开启操作。此字段允许用户启用并选择附加包过滤功能。

### 6.2.3.4 白化

白化处理操作是建立在一个 9 位的线性反馈移位寄存器的基础上，用于生成随机序列，然后用该随机序列对有效载荷（包括有效载荷长度、节点或广播地址和 CRC 校验和）进行异或(XOR)操作以生成白化的有效载荷。在接收机侧，收到的有效载荷再和同样的随机序列经过异或运算得到去白化的数据。这个设置限制了连续的 1 或 0 的数量不超过 9 个。请注意：当用户数据与 0 和 1 的长字符串具有很高的相关性时，只需要数据白化。当数据已经是随机的时，那么就不需要对其进行白化处理。例如，随机数发生器输出的随机数，当对其进行白化处理时，将会产生更长的 1 或 0 字符串，因此对于已经存在的随机序列，就不需要再对其进行随机化的操作。

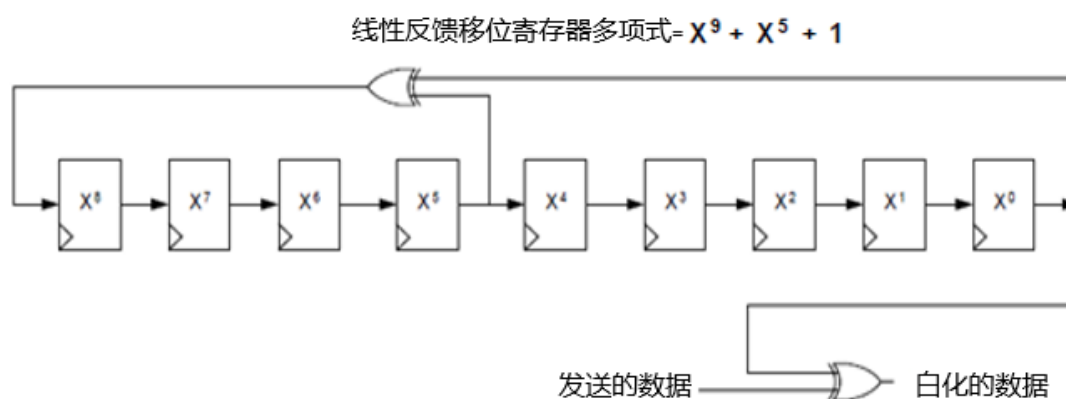


图 6-5: 线性反馈移位寄存器的数据白化操作

白化处理是基于一个 9 位的线性反馈移位寄存器多项式  $x^9 + x^5 + 1$ 。在该结构中，线性反馈移位寄存器输出的最低有效位和最高有效位的数进行异或处理的。

在初始阶段，可以通过地址为 0x06B8 和 0x06B9 的寄存器对线性反馈移位寄存器的每个触发器进行初始化。

表 6-5: 白化操作初始值

白化操作初始值	寄存器地址	缺省值
最高有效位白化初始值	0x06B8	0x01
最低有效位白化初始值	0x06B9	0x00

## 6.2.3.5 CRC

SX1261/2可以灵活的选用多项式和多项式的初始值。此外，用户还可以选择CRC的计算值进行取反以符合一些国际标准。

可以通过命令 *SetPacketParam(...)* 中的字段 *CRCType* 部分对CRC进行使能和配置，此字段允许用户使能和选择CRC的长度和配置。

表 6- 6: CRC 类型配置

CRC 类型	描述
0x01	CRC_OFF (没有 CRC)
0x00	CRC_1_BYTE (CRC 长度 1 字节)
0x02	CRC_2_BYTE (CRC 长度 2 字节)
0x04	CRC_1_BYTE_INV (CRC 长度 1 字节并取反)
0x06	CRC_2_BYTE_INV (CRC 长度 2 字节并取反)

CRC 的初始值和其对应的多项式必须一起进行选择 and 修改。

表 6- 7: CRC 初始值

	寄存器地址	缺省值
CRC 最高位的初始值 [15:8]	0x06BC	0x1D
CRC 最低位的初始值 [7:0]	0x06BD	0x0F

表 6- 8: CRC 多项式

	寄存器地址	缺省值
CRC 最高位的多项式值 [15:8]	0x06BE	0x10
CRC 最低位的多项式值 [7:0]	0x06BF	0x21

这种灵活的运用允许用户选择任何标准的 CRC 或使用自己的私有 CRC 对给定的数据包进行特定的检测。例如：

**对于 IBM 的 CRC 配置，用户必须选择：**

- CRC多项式选用0x8005
- 初始值选用0xFFFF
- 命令 *SetPacketParam(...)* 部分的字段 *CRCType* 选用 CRC\_2\_BYTE

**对于 CCIT 的 CRC 配置，用户必须选择：**

- CRC多项式选用0x1021
- 初始值选用0x1D0F
- 命令 *SetPacketParam(...)* 部分的字段 *CRCType* 选用 CRC\_2\_BYTE\_INV

## 7. 数据缓存

数据收发器配备有 256 字节的 RAM 数据缓存。除了 Sleep 模式，该数据缓存可以在任何模式下被访问。用户可以定义这个 RAM 数据缓存区域，并且可以访问发送数据或者最近接收到的数据。

### 7.1 操作原理

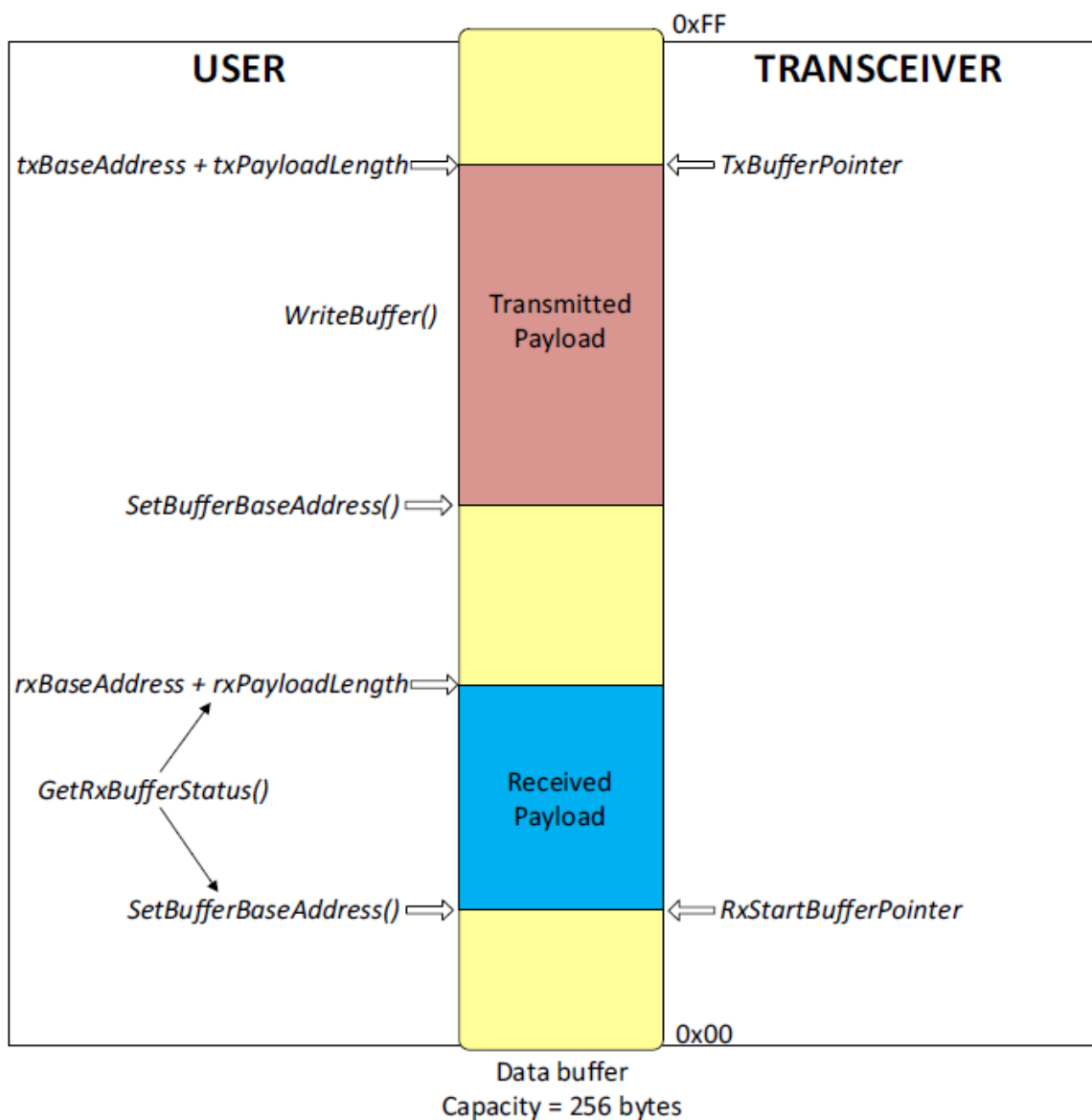


图 7-1: 数据缓存图

该数据缓存可以设置用来存储发送和接收到的数据包。

## 7.2 接收模式下的数据缓存

在接收模式下，RxBaseAddr 指定了存储器里缓存的偏移地址，其用来存放接收数据。在接收模式下被写入的最后一个字节的数据缓存偏移地址被存放在 RxDataPointer，其在开始接收的时候被初始化成 RxBaseAddr 的值。

可以用命令 GetRxbufferStatus()来读取指向上次收到的数据包的第一个字节和数据包长度的指针。

RxSingle 模式下，每次进入到接收模式时 RxDataPointer 都会被自动初始化为 RxBaseAddr。在连续接收模式下，RxDataPointer 指针都会在原来的值上开始自动增加。

## 7.3 发送模式下的数据缓存

在每次转换到发送模式时，指针 TxDataPointer 就会被初始化为 TxBaseAddr，而且每次发送出去一个字节这个指针就会自动增加。一旦发送的字节数等于在函数 SETPACKETPARAMS ( ... ) 中定义的 PayLoeLoad 参数，此操作就会停止。

## 7.4 数据缓存的使用

命令 SetBufferBaseAddresses(...) 用来设置 RxBaseAddr 和 TxBaseAddr。

默认情况下，RxBaseAddr 和 TxBaseAddr 被初始化为 0x00。

由于数据缓存区的连续特性，Tx 和 Rx 的基地址可以在 256 个字节的存储区里随意设置。每个指针都可以被单独配置成缓存区中的任何位置。在接收或者发送模式，为了可以最大化利用数据缓存区，可以通过设置基地址 TxBaseAddr 和 RxBaseAddr 为内存底部 ( 0x00 ) 来使用整个数据缓存区。

当芯片被设置成 Sleep 模式的时候，数据缓存区被清空 ( 意味着不可访问 )。在其他所有模式下数据被保存。

通过命令 WriteBuffer(...) 和 ReadBuffer(...)可以访问数据缓存。在这两个函数中，偏移参数定义了第一个被写入或被读出的数据的地址指针。偏移 0 定义了数据缓存的第一个位置。

所以，在写入或者读取操作之前，必须把这个偏移初始化为对应的缓存的起始地址。一旦从缓存读取或者写入数据，地址指针就会自动增加。

有两种可能性来获得偏移值：

- 第一，用 RxBaseAddr值，因此用户在接收数据之前需要定义它。
- 第二，可以用命令GetRxbufferStatus(...)的返回值RxStartBufferPointer来初始化偏移。

注意：

所有接收到的数据都将被写到数据缓存区，即使 CRC 是无效的，允许用户自我决定处理损坏的数据。在接收的时候，如果数据包的大小超过了为 RX 分配的缓存区内存，那么它将覆盖发射数据缓存的一部分。

## 8. 数字接口和数字控制

SX1261/2 是通过一个串行 SPI 接口以及一组通用输入/输出口 (DIOs) 来控制的。至少需要一个 DIO 用于 IRQ, 并且 BUSY 线也要求强制使用的, 以确保主控制器知道芯片已经准备好接收命令。

SX1261/2 用了一个内部控制器 (CPU) 处理通信和芯片控制 (模式切换, API 等)。BUSY 用来作为忙的信号, 表明只有这个信号为低的时候, 芯片才准备好接收新命令。当 BUSY 为高的时候, 主控制器必须等到它变为低的时候, 才能发送另一个命令。通过 SPI, 应用层发送命令到芯片内部或者直接访问数据内存空间。

### 8.1 复位

芯片一个完整的“恢复出厂设置”可以通过触发 SX1261/2 的 15 引脚 (NRESET) 来实现。接着它会自动执行标准的校准程序, 而且以前的内容将会被丢掉。这个引脚需要至少保持 50us 的低电平 (一般 100us) 以使复位成功。

### 8.2 SPI 接口

SPI 接口通过一个在 Motorola/ Freescale 命名法中与 CPOL=0 和 CPHA=0 相对应的同步全双工协议, 提供对配置寄存器的访问。芯片只实现了 Slave 端功能。

对于写访问, 会发送一个地址字节和一个数据字节的包; 但是对于读访问, 只需要发送一个地址字节, 然后就会收到一个读取的字节。NSS 会在开始发包的时候置低, 在最后一个字节传输完成后拉高。

MOSI 在 SCK 下降沿由主机生成, 并在 SCK 上升沿由从机 (即 SPI 接口) 进行采样。MISO 在 SCK 下降沿由从机生成。

NSS 引脚电平变低时传输启动。当 NSS 高电平时, MISO 处在高阻抗状态。

SPI 运行在一个外部的 SCK 时钟上, 使其能够达到 16MHz 的速率。

#### 8.2.1. 收发器在 active 模式下的 SPI 时序

在这种模式下, 芯片能够用标准方式处理 SPI 命令, 即在第一个 SPI 交互的时候不需要额外的延时。

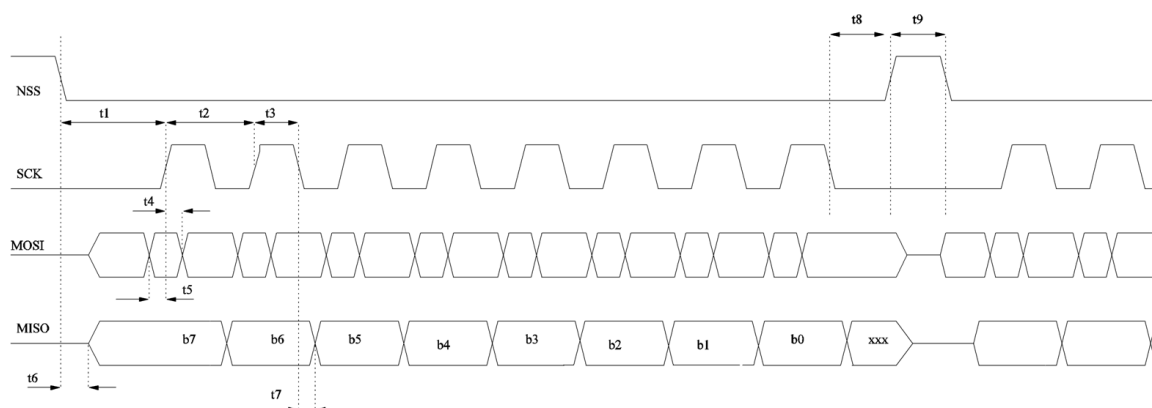


图 8-1: SPI 时序图

下表的所有的时序都是在最大负载电容为 10pF 给出的。

表 8-1: SPI 时序要求

符号	描述	最小值	典型值	最大值	单位
t1	NSS 下降沿 到 SCK 建立时间	32	-	-	ns
t2	SCK 周期	62.5	-	-	ns
t3	SCK 高电平时间	31.25	-	-	ns
t4	MOSI 到 SCK 保持时间	5	-	-	ns
t5	MOSI 到 SCK 建立时间	5	-	-	ns
t6	NSS 下降沿到 MISO 的延迟时间	0	-	15	ns
t7	SCK 下降沿到 MISO 的延迟时间	0	-	15	ns
t8	SCK 到 NSS 上升沿的保持时间	31.25	-	-	ns
t9	NSS 高电平时间	125	-	-	ns
t10	从 SLEEP 切换到 STDB_RC 模式时, NSS 下降沿到 SCK 建立时间	100	-	-	s
t11	从 SLEEP 切换到 STDB_RC 模式时, NSS 下降沿到 MISO 的延迟时间	0	-	150	s

### 8.2.2. 离开 Sleep 模式的 SPI 时序

芯片离开 Sleep 模式的一种方法是等到 NSS 的下降沿。在下降沿, 所有必要的内部变换器都被打开; 芯片初始化后才能接受第一个 SPI 命令。这意味着, NSS 的下降边缘与 SCK 的第一个上升边缘之间的延迟必须考虑到唤醒序列和芯片初始化。在睡眠模式和初始化阶段, BUSY 引脚被设置为高电平, 告知主机芯片不能接收新命令。一旦芯片处于 STANBY\_RC 模式, BUSY 信号置低, 主机能够开始发送命令。对插入电池或者硬件复位也是同样适用的。

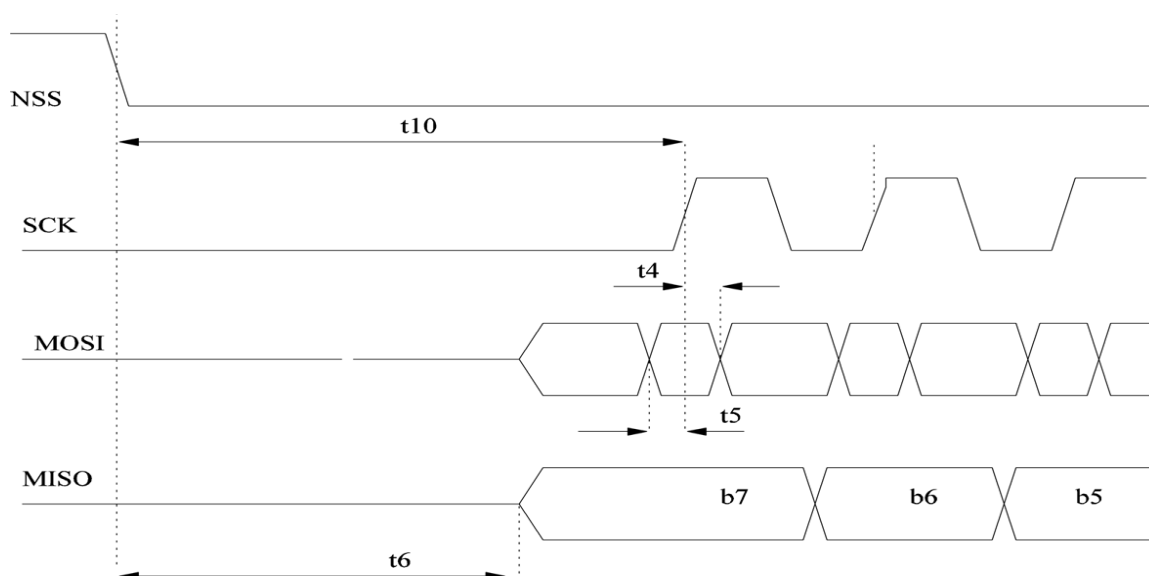


图 8-2: SPI 时序转换

### 8.3 多功能数字输入/输出 (DIO)

该芯片有 4 个控制线接口，该控制线由 BUSY 引脚和 3 个 DIOs 引脚组成。其中 3 个 DIOs 引脚可配置为中断、调试或用来控制器件直连的外围设备 (TCXO 或 RF 开关)。

#### 8.3.1. BUSY 控制线

BUSY 控制线用于指示内部状态机的状态。当 BUSY 线保持低电平时，说明内部状态机处于空闲模式且芯片已经准备好从主控制器接收命令。

一旦芯片达到稳定模式，BUSY 控制线恢复到 0，此时芯片准备好接收新命令。本质上，BUSY 线保持高电平的时间取决于命令的性质。例如，设置芯片从 STDBY\_RC 模式切换到 TX 模式就比简单改变一些无线参数的值花费的时间要多，因为内部状态机会保持 BUSY 线为高电平直到芯片可以有效地发送数据包。

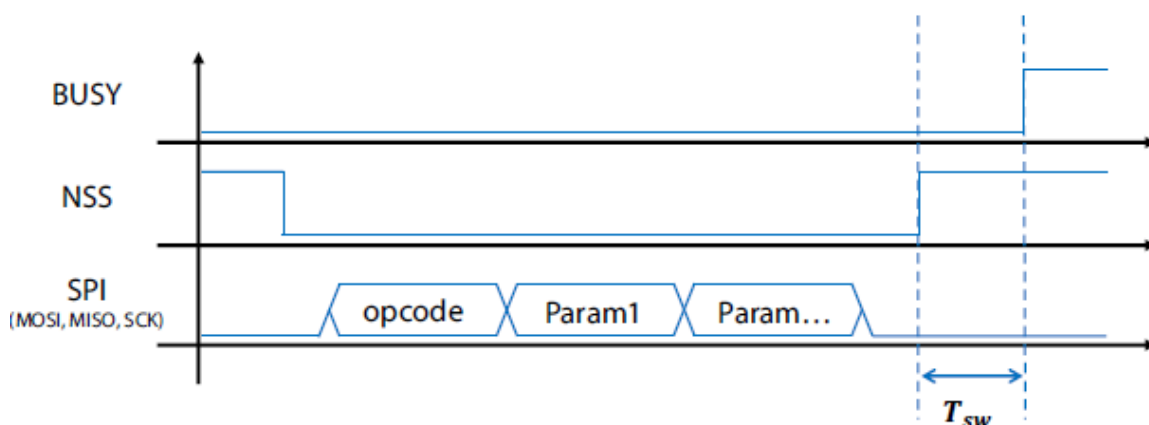


图 8-3: 切换时间的定义

从内部状态机的角度来看，所有的“写”命令将使 BUSY 信号经过一小段时间  $T_{sw}$  后被拉高，如上图所示的  $T_{sw}$ 。 $T_{sw}$  代表内部状态机唤醒和开始处理命令所需要的时间。

相反地，“读”命令将在没有内部状态机的帮助下直接被处理，所以 BUSY 信号将会在“读”命令后保持低电平。

从 NSS 的上升沿到 BUSY 的上升沿的这段时间  $T_{sw}$  的最大值，在所有情况下都是 600ns。

在 Sleep 模式下，BUSY 管脚通过一个 20KΩ 的电阻保持高电平，芯片一旦离开 Sleep 模式 BUSY 管脚就会变低电平。

在 FS 模式下，PLL 锁定后 BUSY 管脚将会变为低电平。

在 RX 模式下，一旦切换到 RX 模式并且准备好接收数据的时候 BUSY 管脚就会变为低电平。

在 TX 模式下，当 PA 已经启动且开始发射前导码的时候 BUSY 管脚将会变为低电平。

除此之外，在处理内部 IRQ 的时候 BUSY 管脚将会拉高。这种场景下，必须等到 BUSY 管脚变低才能发送 SPI 命令（“读”或“写”命令）。

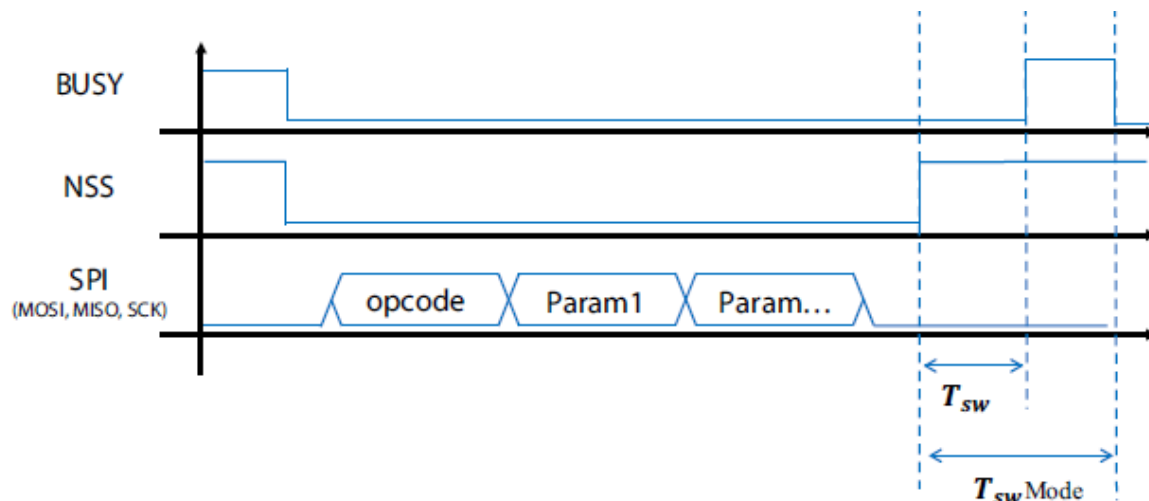


图 8-4：在 ACTIVE 模式下切换时间的定义

下表给出了所有可能的转换所需要的时间  $T_{SWMode}$  的值。转换时间被定义为 SPI 交互结束时 NSS 上升沿和 BUSY 信号下降沿之间的时间。

表 8-2：切换时间

Transition	$T_{SWMode}$ Typical Value [ $\mu$ s]
SLEEP to STBY_RC cold start (no data retention)	3500
SLEEP to STBY_RC warm start (with data retention)	340
STBY_RC to STBY_XOSC	31
STBY_RC to FS	50
STBY_RC to RX	83
STBY_RC to TX	126
STBY_XOSC to FS	40
STBY_XOSC to TX	105
STBY_XOSC to RX	62
FS to RX	41
FS to TX	76
RX to FS	15
RX to TX	92

### 8.3.2. 数字输入/输出

三个 DIOs 中的任何一个都可以选来作为输出中断源。当应用程序接收到一个中断，用命令 `GetIrqStatus(...)` 可以确定中断源。命令 `ClearIrqStatus(...)` 可以用来清除中断。管脚描述如下：

DIO1 是通用的 IRQ 管脚，任何中断都可以映射到 DIO1。有效 IRQ 的完整列表参考 53 页 [8.4 节 “数字接口状态与芯片模式”](#)。

DIO2 具有双重功能。类似 DIO1，DIO2 可以被用作一个通用的 IRQ 管脚，而且任何 IRQ 都可以通过 DIO2 进行输出。同样，通过用命令 `SetDio2AsRfSwitchCtrl(...)`，DIO2 可以被配置用于驱动 RF 开关。这种模式下，DIO2 在 TX 期间为高电平且在其它模式下为低电平。

DIO3 也有双重功能，类似 DIO1 或 DIO2，它可以用作通用 IRQ。同样，DIO3 也可以通过命令 `SetDio3AsTCXOCtrl(...)` 用于自动控制 TCXO。这种情况下，在需要的时候芯片将会自动周期性的对 TCXO 供电。

## 8.4 数字接口状态与芯片模式

表 8-3：每种芯片模式的数字端口配置

Mode	DIO3	DIO2	DIO1	BUSY	MISO	MOSI	SCK	NSS	NRESET
Reset	PD	PD	PD	PU	HIZ	HIZ	HIZ	IN	-
Start-up	HIZ PD	HIZ PD	HIZ PD	HIZ PU	HIZ	HIZ	HIZ	IN	IN PU
Sleep	HIZ PD	HIZ PD	HIZ PD	HIZ PU	HIZ	HIZ	HIZ	IN	IN PU
STBY_RC	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	IN PU
STBY_XOSC	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	IN PU
FS	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	IN PU
RX	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	IN PU
TX	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	IN PU

注意：

- PU=典型条件下用一个 50KΩ 上拉
- PD=典型条件下用一个 50KΩ 下拉（电阻值随着供电电压变化）

## 8.5 IRQ 处理

根据所选择的帧和芯片模式，总共有十个可能的中断源。每种中断都可以被使能或屏蔽。此外，它们都可以被映射到 DIO1、DIO2 或 DIO3。

表 8-4: IRQ 状态寄存器

位	IRQ	描述	协议
0	TxDone	包发送完成	All
1	RxDone	包接收完成	All
2	PreambleDetected	检测到前导	All
3	SyncWordValid	检测到有效同步字	FSK
4	HeaderValid	接收到有效的 LoRa 包头	LoRa®
5	HeaderErr	LoRa® 包头 CRC 错误	LoRa®
6	CrcErr	收到错误的 CRC	All
7	CadDone	CAD 检测完成	LoRa®
8	CadDetected	检测到信道活动状态	LoRa®
9	Timeout	Rx or Tx 超时	All

关于怎么设置 IRQ 和 DIOs 的更多信息，请参考 78 页 [13.3.1 节 “SetDioIrqParams”](#) 的函数 SetDioIrqParams()。

## 9. 工作模式

SX1261/2 具有 6 种操作模式。下表介绍了在每种工作模式下使能的模拟前端和数字模块。

表 9-1: SX1261/2 工作模式

模式	使能模块
SLEEP	可选寄存器, Backup Regulator, RC64K 振荡器, 数据 RAM
STDBY_RC	LDO, RC13M 振荡器
STDBY_XOSC	DC-DC 或 LDO, XOSC
FS	以上所有模块 + 工作在 TX 频率的频率合成器
Tx	频率合成器和发射机, 调制器
Rx	频率合成器和接收机, 调制器

### 9.1 Startup

在上电或者复位后, 芯片进入 STARTUP 状态, 芯片的控制是由在电池电压下运行的睡眠状态机来完成的。BUSY 管脚被设置成高电平, 意味着芯片忙碌中且不能接收新命令。当数字电压和 RC 时钟变为有效时, 芯片能启动且受 CPU 控制。在这个阶段, BUSY 管脚拉低且芯片准备好接收命令。

### 9.2 校准

在 POR 情况下或者调用校准命令时, 会自动调用校准程序。可以将参数添加到校准命令中, 以确定应该重复哪个校准部分。下面几个部分可以被校准:

- RC64K, 用 32MHz 晶振作为参考
- RC13M, 用 32MHz 晶振做作为参考
- PLL, 选择合适的 VCO 频率以及分频比
- RX ADC
- 镜像抑制 (指定单频的接收模式)

一旦校准完成, 芯片进入 STDBY\_RC 模式。

#### 9.2.1. 指定频段的镜像校准

对于由参数 freq1 和 freq2 定义的区间的频率的镜像校准, 可以通过命令 CalibrateImage(...)来完成。一旦执行, 校准对于两个极端参数之间的所有频率都是有效的。通常, 用户可以选择参数 freq1 和 freq2 来覆盖所有指定的 ISM 频段。

表 9- 2: 覆盖 ISM 频段的镜像校准

Frequency Band [MHz]	Freq1	Freq2
430 - 440	0x6B	0x6F
470 - 510	0x75	0x81
779 - 787	0xC1	0xC5
863 - 870	0xD7	0xDB
902 - 928	0xE1 (default)	0xE9 (default)

冷启动模式下，在 POR 情况下或者当设备从 Sleep 模式中恢复时，镜像校准作为初始化校准过程的一部分来执行，且在频段 902-928MHz 具有最佳镜像抑制。但是，在这个阶段，内部状态机并不知道频率源来自于 XTAL 还是 TCXO。当 32MHz 来自于 TCXO 时，校准会失败，因此用户应该在调用函数 SetDIO3AsTcxoCtrl(...)后，再执行一次完整的校准。

默认情况下，镜像校准在 902-928MHz 频段上进行。然而，可以要求芯片在其他的频率上执行一个新的镜像校准。

注意：

联系你的 Semtech 代表获取更多关于给定频段以外的其他优化校准设置。

### 9.3 Sleep 模式

在这个模式下，大部分芯片内部块被关闭或处于低功耗模式，RC64K 时钟和定时器在运行（可选地）。芯片可以从 STDBY\_RC 模式进入这种模式，并且当以下事件中的一件发生时，芯片可以离开 Sleep 模式：

- NSS 管脚在任何情况下拉低
- RTC 定时器产生一个 End-Of-Count（对应于监听模式）

当芯片处于 Sleep 模式，BUSY 管脚保持为高。

### 9.4 STDBY 模式

在 STDBY 模式下，主机应该在进入 RX 或 TX 模式前配置芯片。默认在这种状态下，系统由 13MHz 的 RC 振荡器来提供时钟以减少功耗（在除了 SLEEP 模式外的所有其他模式下 XTAL 是打开的）。然而，如果应用对时间要求很严格，XOSC 可以打开或保持打开状态。

在 STDBY 模式下 XOSC 或 RC13M 的选择由命令 SetStandby(...)中的模式参数来决定。

只有使用 RC13M 的模式被称为 STDBY\_RC，而使用 XOSC 的模式则称为 STDBY\_XOSC。

如果要使用 DC-DC，在 STDBY\_RC 模式下，通过使用 SetRegulatorMode(...)命令进行开启。然而当进入 STDBY\_XOSC 模式时，DC-DC 将自动开启。DC-DC 将使用 RC13M 时钟。此时 LDO 将保持激活状态，目标电压比 DC-DC 低 50mV。

## 9.5 FS 模式

在 FS 模式中，PLL 和相关变换器被打开。只要 PLL 锁住或者超时 BUSY 管脚就会变低。

出于调试的目的，通过使用 SetFs()命令请求芯片保持在这种模式下。

由于 SX1261/2 使用低中频架构，RX 和 TX 的频率是不同的。RX 的频率等于 TX 频率减去中频频率。在 FS 或 TX 模式，射频频率是用户直接来配置的。

## 9.6 RX 模式

在 RX 模式，射频前端，RX ADC 以及选中的调制解调器都被打开。在 RX 模式，电路能够在不同的子模式下运行：

- 连续模式：芯片处于 RX 模式，等待接收到来的数据包，直到主机请求一个不同的模式
- 单一模式：接收完数据包后芯片自动回到 STDBY\_RC 模式
- 具有超时功能的单一模式：接收完数据包或者超时之后，设备自动回到 STDBY\_RC 模式
- 监听模式：设备在 Sleep 模式和 RX 模式之间交替直到一个 IRQ 被触发。

在 RX 模式，只要进入接收并且准备好接收数据，BUSY 管脚就会拉低。

SX1261/2 可以在 RX Boosted 增益设置或 RX 节电增益设置中运行。在 RX 节电增益中，芯片将会消耗较少的功耗，但是灵敏度会降低。在 RX Boosted 增益时，芯片将会消耗较多的功耗，但是灵敏度会提高。

表 9-3: RX 增益配置

Rx Gain	寄存器地址	[1:n]
RxGain	0x08AC	RX 节电增益:0x94(默认) RX boosted 增益: 0x96

注意：

当处于 LoRa 模式，用户也可以使用命令 SetLoRaSymbNumTimeout(...)对 LoRa 前导码存在（或不）做一个快速和即时的评估。如果用户定义参数 SymbNum 不等于 0，调制解调器将会等待接收 SymbNum 个码元来验证 LoRa 数据包是否正确。如果解调器的各种状态在此时都没有锁住，那么芯片将会产生 RxTimeout IRQ 中断。否则，芯片将在整个包期间都会停留在 RX。更多信息，请参见第 93 页的 [13.4.9 节 “SetLoRaSymbNumTimeout”](#)。

## 9.7 TX 模式

在 TX 模式下，功率放大器启动后，将发送数据缓存区的数据。发射模式也可以工作在不同的子模式下：单一发送模式或者具有超时功能的单一发送模式。

无论出于何种原因，发射被终止或者不成功（例如 TxDone IRQ 从来没有被触发），TX 模式中的超时功能可以作为一个安全机制以防止系统等待一个未知的未知的时间。在 TX 模式使用超时功能，消除了使用主机资源来执行相同任务的需求。

在 TX 模式下，只要 PA 启动且开始传输前导码，BUSY 管脚就会变低。

### 9.7.1. PA Ramping

用命令 SetTxParams(...)设置输出功率的同时，可以选择 PA Ramping 时间。PA Ramp 时间可以在 10us 到 3.4ms 之间选择。

## 9.8 激活模式的切换时间

欲了解更多关于活跃模式的切换时间的详细信息，参考 51 页的 [8.3.1 节 “BUSY 控制线”](#)。

## 9.9 芯片电路模式图形说明

所有的操作模式和状态，每个模式选择的转换都显示在下图。

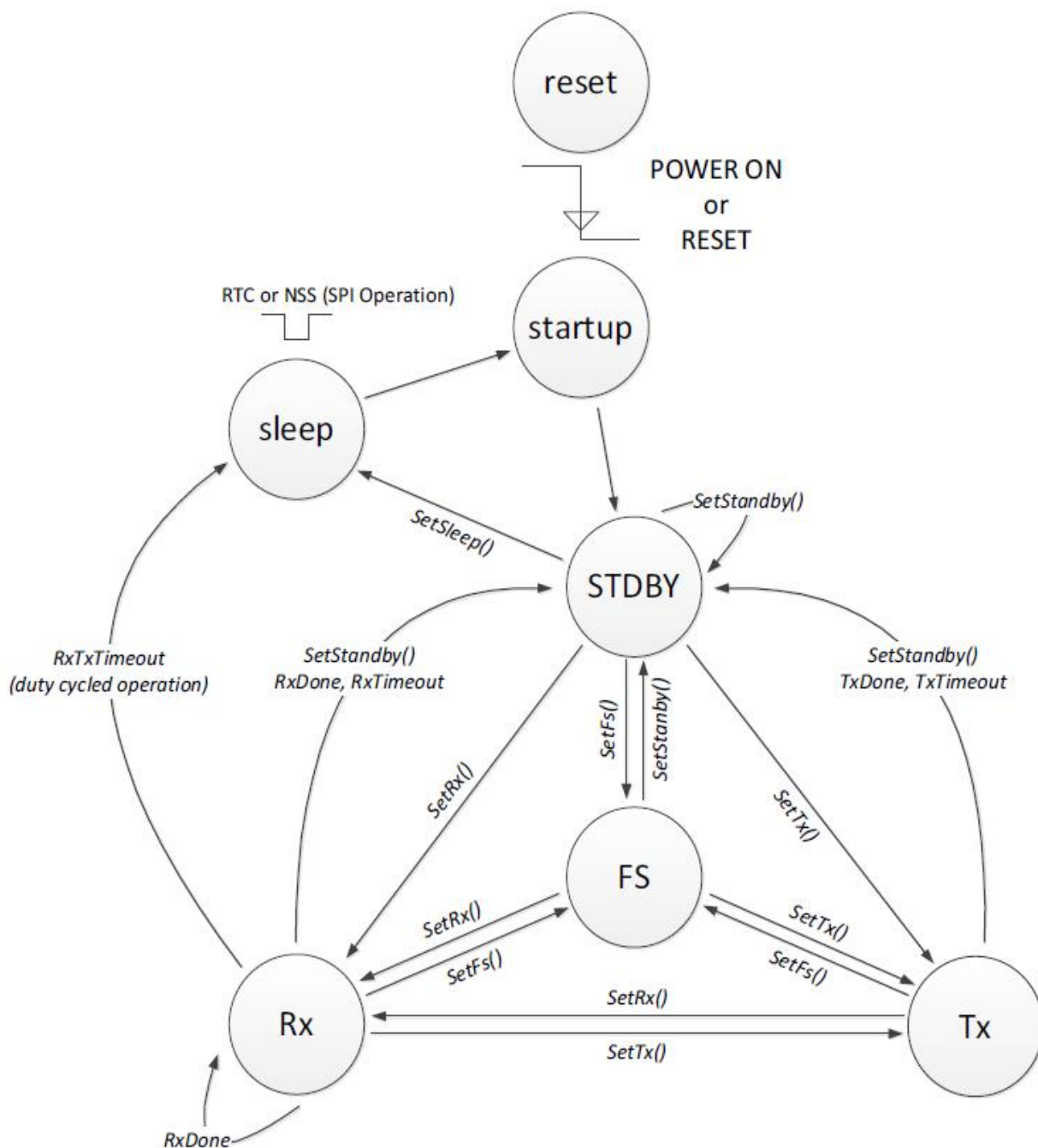


图 9-1: 芯片电路模式

## 10. 主机控制接口

通过 SPI 接口，主机可以向芯片发送命令或者访问数据内存空间来直接提取或写入数据。正常操作下，需要减少除了数据缓存以外的直接数据写入操作。

用户通过 API 与电路交互（指令系统）。

SX1261 使用管脚 BUSY 指示芯片的状态以及在其执行其内部处理操作的时候是否有能力接收另一个命令。在执行任何一个通用命令之前，有必要先检查一下 BUSY 管脚的状态以确保芯片处于可以处理另外的功能的状态。

### 10.1 命令结构

在一个不需要任何参数的命令的情况下，主机只会发送操作码（一个字节）。

在需要一个或者若干参数的命令的情况下，操作码字节后直接跟着参数字节，NSS 的上升沿终结命令。

表 10-1: SPI 接口命令序列

Byte	0	[1:n]
主机下发的数据	Opcode	参数
发给主机的数据	RFU	状态

### 10.2 交互终止

主机可以通过拉高 NSS 信号来终止一个 SPI 交互；主机不会发送命令长度为参数。主机不可以在交互期间拉高 NSS。

如果需要发送一个带参数的命令，所有的参数都必须在 NSS 被拉高前发送出去。否则芯片将会为这些丢失的参数取一些未知的值。

## 11. 命令列表

下表给出了命令列表以及它们对应操作码。如果没有特别指定，所有的参数都是 8 位的。

### 11.1 工作模式命令

这些功能对芯片的行为有直接的影响。它们控制内部状态机来传输或接收数据包以及中间的所有模式。

表 11- 1: 选择芯片工作模式的命令

命令	操作码	参数	描述
SetSleep	0x84	sleepConfig	配置芯片在 SLEEP 模式
SetStandby	0x80	standbyConfig	配置芯片在 SLEEP STDBY_RC 或 STDBY_XOSC 模式
SetFs	0xC1	-	配置芯片在 FS 模式
SetTx	0x83	timeout[23:0]	配置芯片在 TX 模式
SetRx	0x82	timeout[23:0]	配置芯片在 RX 模式
StopTimerOnPreamble	0x9F	StopOnPreambleParam	在同步字/包头或前导检测上停止 RX 超时
SetRxDutyCycle	0x94	rxPeriod[23:0], sleepPeriod[23:0]	存储用于侦听模式的 RTC 设置值，如果周期参数不为 0，则将芯片设置为 RX 模式
SetCad	0xC5	-	将芯片设置为 CAD 模式
SetTxContinuousWave	0xD1	-	设置芯片进入连续 CW 发射模式
SetTxInfinitePreamble	0xD2	-	设置芯片进入连续前导发射模式
SetRegulatorMode	0x96	regModeParam	为 CFGXXOSC、FS、RX 或 TX 模式选择 LDO 或 DC-DC+LDO
Calibrate	0x89	calibParam	根据参数对 RC13、RC64、ADC、PLL、镜像进行校准
CalibrateImage	0x98	freq1, freq2	在给定频率下进行镜像校准
SetPaConfig	0x95	paDutyCycle, HpMax, deviceSel, paLUT	配置用于 SX1261 或 SX1262 的 PA 的占空比、最大输出功率
SetRxTxFallbackMode	0x93	fallbackMode	在 TX /RX 完成后，定义芯片进入哪种模式

## 11.2 寄存器和缓存访问命令

表 11- 2: 访问芯片寄存器和 FIFO 缓存的命令

命令	操作码	参数	描述
WriteRegister	0x0D	address[15:0],data[0:n]	写一个或者若干寄存器
ReadRegister	0x1D	address[15:0]	读写一个或若干寄存器
WriteBuffer	0x0E	offset, data[0:n]	将数据写入到 FIFO
ReadBuffer	0x1E	offset	从 FIFO 读数据

## 11.3 DIO 和 IRQ 控制

表 11- 3: 控制芯片 IRQs 和 DIOs 的命令

命令	操作码	参数	描述
SetDioIrqParams	0x08	IrqMask[15:0], Dio1Mask[15:0], Dio2Mask[15:0], Dio3Mask[15:0],	在每个 IRQ 上配置 IRQ 和 DIOs
GetIrqStatus	0x12	-	获取被触发的 IRQs 的值
ClearIrqStatus	0x02	-	清除一个或若干 IRQs
SetDIO2AsRfSwitchCtrl	0x9D	Enable	配置 DIO2 以便控制射频开关
SetDIO3AsTcxoCtrl	0x97	tcxoVoltage, timeout[23:0]	配置 DIO3 以便控制的 TCXO

## 11.4 射频、调制和数据包命令

表 11- 4: 控制射频和数据包设置的命令

命令	操作码	参数	描述
SetRfFrequency	0x86	rfFreq[23:0]	设置芯片的射频频率
SetPacketType	0x8A	protocol	选择对应调制方式的数据包类型
GetPacketType	0x11	-	获取设备的当前数据包配置
SetTxParams	0x8E	power, rampTime	设置 PA 的输出功率和 Ramping 时间
SetModulationParams	0x8B	modParam1, modParam2, modParam3	设置调制参数
SetPacketParams	0x8C	packetParam1, packetParam2, packetParam3, packetParam4, packetParam5, packetParam6, packetParam7, packetParam8, packetParam9	设置包参数
SetCadParams	0x88	cadSymbolNum, cadDetPeak, cadDetMin, cadExitMode, cadTimeout	设置 CAD 参数 ( 仅 LoRa 模式 )
SetBufferBaseAddress	0x8F	TxbaseAddr, RxbaseAddr	设置存放 RX 和 TX 的基地址
SetLoRaSymbNumTimeout	0xA0	SymbNum	设置调制解调器等待有效锁定的码元的个数

## 11.5 状态命令

表 11- 5: 返回芯片状态命令

命令	操作码	参数	描述
GetStatus	0x8C	-	返回芯片当前的状态
GetRssiInst	0x15	-	在 RX 模式下返回 RSSI 瞬时值
GetRxBufferStatus	0x13	-	返回 PayloadLengthRx(7:0), RxBufferPointer(7:0)
GetPacketStatus	0x14	-	返回 FSK 模式下 RssiAvg, RssiSync, PStatus2, PStaus3, PStatus4 的值, 返回 LoRa 模式下 RssiPkt, SnrPkt
GetDeviceErrors	0x17	-	返回芯片的错误信息
ClearDeviceErrors	0x07	0x00	清除所有的错误。错误不能被单独清除
GetStats	0x10	-	返回最近几个接收包的统计信息
ResetStats	0x00	-	复位通过命令 GetStats 读取的值

## 12. 寄存器映射

### 12.1 寄存器列表

表 12-1: 寄存器列表

Register Name	Address	Reset Value	Function
Whitening initial value MSB	0x06B8	0xX1	FSK 模式下 LFSR 白化的初始值。用户不能改变这个寄存器最高 7 位的值
Whitening initial value LSB	0x06B9	0x00	
CRC MSB Initial Value [0]	0x06BC	0x1D	多项式的初始值，用于计算 FSK 模式下的 CRC
CRC MSB Initial Value [1]	0x06BD	0x0F	
CRC MSB polynomial Value [0]	0x06BE	0x10	多项式，用于计算 FSK 模式下的 CRC
CRC LSB polynomial Value [1]	0x06BF	0x21	
SyncWord[0]	0x06C0	-	FSK 模式下同步字的第一个字节
SyncWord[1]	0x06C1	-	FSK 模式下同步字的第二个字节
SyncWord[2]	0x06C2	-	FSK 模式下同步字的第三个字节
SyncWord[3]	0x06C3	-	FSK 模式下同步字的第四个字节
SyncWord[4]	0x06C4	-	FSK 模式下同步字的第五个字节
SyncWord[5]	0x06C5	-	FSK 模式下同步字的第六个字节
SyncWord[6]	0x06C6	-	FSK 模式下同步字的第七个字节
SyncWord[7]	0x06C7	-	FSK 模式下同步字的第八个字节
Node Address	0x06CD	0x00	FSK 模式下的节点地址
Broadcast Address	0x06CE	0x00	FSK 模式下的广播地址
LoRa Sync Word MSB	0x0740	0x14	区分公共或者私有网的 LoRa 信号 公共网络设置为 0x3444 私有网络设置为 0x1424
LoRa Sync Word LSB	0x0741	0x24	
RandomNumberGen[0]	0x0819	-	用于产生 32 位的随机数
RandomNumberGen[1]	0x081A	-	
RandomNumberGen[2]	0x081B	-	
RandomNumberGen[3]	0x081C	-	
Rx Gain	0x08AC	0x94	设置 RX 模式下的增益： RX 节电增益：0x94 RX Boosted 增益：0x96
OCP Configuration	0x08E7	0x18	设置电流过载保护。 这个值依赖于所选择的芯片。默认的值如下： SX1262: 0x38 ( 140mA ) SX1261: 0x18 ( 60mA )
XTA trim	0x0911	0x05	XTA 管脚内部集成电容。这个寄存器只能在 STDBY_XOSC 模式时被改写。
XTB trim	0x0912	0x05	XTB 管脚内部集成电容。这个寄存器只能在 STDBY_XOSC 模式时被改写。



## 13. 命令接口

### 13.1 工作模式配置函数

#### 13.1.1. SetSleep

SetSleep(...)命令用来将芯片设置在 SLEEP 模式，该模式下功耗最小。只能在芯片为 STDBY (STDBY\_RC 或 STDBY\_XOSC) 模式下发送该命令。NSS 上升沿之后，除了 Backup Regulator (如果需要的话) 以及在参数 sleepConfig 里边指定的模块，所有模块关闭。

表 13- 1: SetSleep SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x84	sleepConfig

sleepConfig 参数在表 13-2 中定义。

表 13- 2: Sleep 模式定义

SleepConfig[7:3]	SleepConfig [2]	SleepConfig [1]	SleepConfig [0]
RESERVED	0: 冷启动	0: RFU	0: 禁用 RTC 超时时钟
RESERVED	1: 热启动 (保存芯片配置) <sup>2</sup>	0: RFU	1: RTC(RC64k)超时时钟唤醒

1. 注意，进入 SLEEP 之前启用的调制解调器的配置才会保存，其他调制解调器的配置会丢失，需要重新配置。

当进入 SLEEP 模式，BUSY 引脚输出高电平，且在 SLEEP 期间一直保持高电平。

当处在 SLEEP 模式，主机可以通过触发 NSS 下降沿唤醒该芯片。芯片也可以使用 64kHz RTC 时钟计数器自动唤醒。当使用 RTC 时钟，NSS 上升沿也会唤醒芯片（主机仍能控制芯片）。

默认情况下，当进入 SLEEP 模式，芯片的配置会丢失。但睡眠模式下若能保存芯片配置，可以减少与主机的交互，另外对于 RxDutyCycle 模式这又是必须具备的功能。这可以通过在 SLEEP 模式下使用记忆模式寄存器实现。这要将 SetSleep(...)命令 sleepConfig[2]设为 1。当芯片离开 SLEEP 模式，它会首先用保存在记忆寄存器里边的值恢复寄存器值。

#### 警告：

当发送了 SetSleep(...)命令，芯片会有约 500 微秒的时间不作响应，期间它进行配置保存和模块关闭等操作。用户必须确保在这段时间不向设备发送 SPI 命令，以确保芯片的正常操作。

### 13.1.2. SetStandby

*SetStandby(...)*命令使芯片工作在一个可配置模式，该模式的功耗为处于中间水平。该模式下，芯片处在停止模式下并等待通过 SPI 传输的指令。该模式主要是用来进行一些高级别的芯片设置，比如 *SetPacketType(...)*。

当芯片接入电池或复位操作（NRESET 引脚输出低电平），芯片会默认进入 STDBY\_RC 模式，使用 13MHz RC 时钟。

表 13- 3: STDBY 模式配置

字节	0	1
主机下发的数据	Opcode = 0x80	StdbbyConfig

*StdbbyConfig* 字节如下定义：

表 13- 4: STDBY 模式配置

StdbbyConfig	值	描述
STDBY_RC	0	设置为 STDBY_RC 模式，运行 RC13M
STDBY_XOSC	1	设置为 STDBY_XOSC，运行 XTAL 32MHz

### 13.1.3. SetFs

*SetFs()*命令将芯片设置在频率合成器模式，PLL 锁定在载波频率。这个模式用作测试 PLL，可以理解为是一种中间过渡模式，当从 STDBY\_RC 模式切换到 TX 或 RX 模式时会自动进入这个模式。

表 13- 5: SETFs SPI 交互

字节	0
主机下发的数据	Opcode = 0xC1

在 FS 模式下，PLL 会被设置在 *SetRfFrequency(...)*函数设定的频率，TX 或 RX 操作也会使用该函数设置。

### 13.1.4. SetTx

*SetTx()*命令将芯片设置在发射模式。

表 13- 6: SETTX SPI 交互

字节	0	1-3
主机下发的数据	Opcode = 0x83	timeout(23:0)

- 从 STDBY\_RC 模式开始，振荡器、PLL、PA 依次打开，然后 PA 电压变换器根据 *SetTxParams(...)* 命令设定的 Ramping 时间开始控制电压上升。
- 当 Ramping 结束，包控制器开始进行发包。
- 当包发送完成会产生 TX\_DONE 中断，PA 电压变换器开始降压，然后 PA 关闭，芯片回到 STDBY\_RC 模式。
- 如果在指定的超时时间内，TX\_DONE 中断没有产生，TIMEOUT 中断会触发。

超时时间可以用如下公式计算：

$$\text{Timeout duration} = \text{Timeout} * 15.625 \mu\text{s}$$

*Timeout* 是 24 位的参数，表示使用多少个  $15.625 \mu\text{s}$  为超时时间，详细见下表介绍。

表 13- 7: SetTx 超时时间长度

Timeout(23:0)	超时时间长度
0x000000	禁用超时时钟，单个数据包发送模式，芯片会处在 TX 模式直到发完包并切换到 STBY_RC 模式。
其他值	启用超时时钟，芯片保持在 TX 模式，当达到超时时间或一个包发送完毕，它会自动切换到 STBY_RC 模式。最长超时时间是 262 秒。

超时时间应当根据一个包的长度、调制方式和包参数计算，超时机制提供一个安全保障，防止来自主机命令产生冲突。

无论出于何种原因，发射被终止或者不成功（例如 TxDone IRQ 从来没有被触发），TX 模式中的超时功能可以作为一个安全机制以防止系统等待一个未知的的时间。在 TX 模式使用超时功能，消除了使用主机资源来执行相同任务的需求。

### 13.1.5. SetRx

SetRx()命令将芯片设置在接收模式。

表 13- 8: SETRX SPI 交互

字节	0	1-3
主机下发的数据	Opcode = 0x82	timeout(23:0)

该命令设置芯片在 RX 模式，等待接收一个或多个包。用户可以设置接收超时时间。

表 13- 9: SETRX 超时时间长度

Timeout15:0)	超时时间长度
0x000000	没有设置超时时钟，单包接收模式。芯片会处在 RX 模式直到接收到包，收包完成后切换到 STBY_RC 模式。
0xFFFFFFFF	连续接收模式，芯片会保持在 RX 模式直到主机发命令切换模式。芯片可以接收多个包，每次收到一个包，会向主机发送接收完成中断信号，并自动地继续接收新的包。
其他值	启用超时时钟，芯片保持在 RX 模式，当超时时间结束或者收到一个包，芯片会自动切换到 STBY_RC 模式。一旦检测到有包，超时时钟会自动禁用以确保有足够时间完成收包。最大超时时间是 262 秒。

当启用接收超时功能时 (  $0x000000 < \text{timeout} < 0xFFFFFFFF$  )，芯片将会在超时时间结束时中止接收，除非检测到前导码和同步字 ( GFSK ) 或包头 ( LoRa® )。这确保一个正在接收的包不会因为设置的超时时间而被丢弃。默认情况下，超时时钟只在检测同步字或包头的时候才停止，也可以通过 *StopTimerOnPreamble(...)* 命令设置在检测到前导时停止超时时钟。

### 13.1.6. StopTimerOnPreamble

*StopTimerOnPreamble(...)* 命令允许用户选择接收超时时钟在检测到前导还是同步字/包头时停止。

表 13- 10: STOPTIMERONPREAMBLE SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x9F	StopOnPreambleParam

StopOnPreambleParam 字节在下表中定义。

表 13- 11: STOPONPREAMBPARAM 定义

StopOnPreambleParam	值	描述
禁用	0x00	检测到同步字或包头时才停止超时时钟
启用	0x01	检测到前导码时停止超时时钟

默认情况下，超时时钟只在检测同步字 ( GFSK ) 或包头 ( LoRa® ) 的时候才停止。当使用 *StopTimerOnPreamble(...)* 命令且 StopOnPreambleParam 设置为 0x01 时，超时时钟在检测到前导时会停止，且芯片会保持在 RX 模式直到包接收完成。需要注意的是，启用检测到前导时停止超时时钟，可能导致芯片在误检测到前导后长时间停留在接收状态。

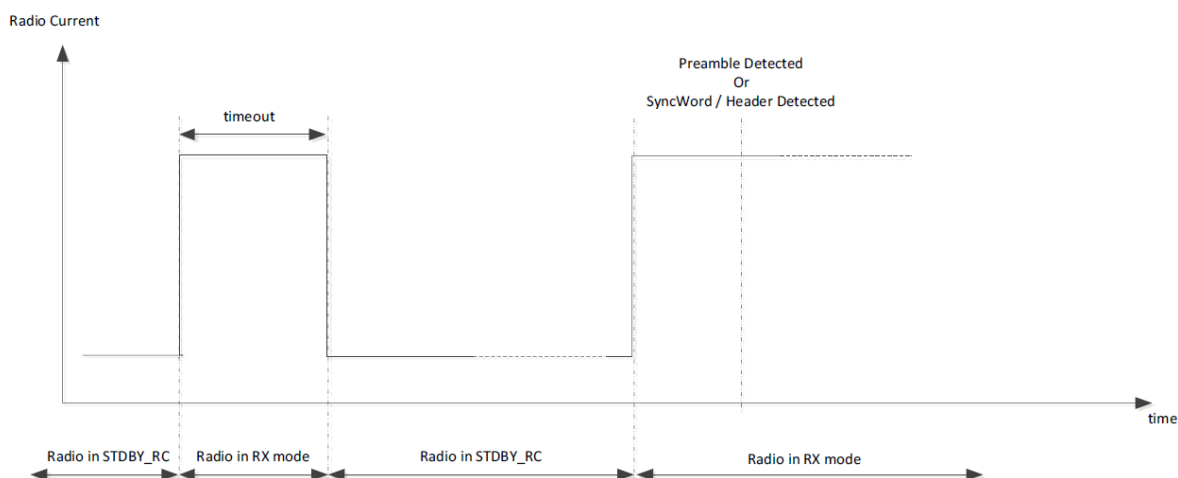


图 13- 1: 检测到前导或包头时停止超时时钟

### 13.1.7. SetRxDutyCycle

这个命令将芯片设置为呼吸模式 (sniff mode)，芯片会有规律地查看是否有新的包，这是一种监听模式。

表 13- 12: SETRXDUTYCYCLE SPI 交互

字节	0	1-3	4-6
主机下发的数据	Opcode= 0x94	rxPeriod(23:0)	sleepPeriod(23:0)

当在 STDBY\_RC 模式下发送这个命令，芯片配置会被保存，并进入如下的循环操作：

- 芯片进入 RX 状态，监听是否有包，监听的时长是根据 rxPeriod 而定
- 芯片监听是否有 LoRa®或 FSK 的前导
- 如果检测到前导，超时时钟停止，并启动一个时长为  $2 * rxPeriod + sleepPeriod$  的超时时钟
- 如果在 RX 窗口 (由 rxPeriod 设定) 没有收到包，芯片会进入 SLEEP 模式，时长为 sleepPeriod，配置会被保存
- 当 SLEEP 时间结束，芯片会自动再进行恢复配置及进入 RX 模式，如此循环。任何时候，主机可以中断该过程。

当如下任一情况出现，循环会中止：

- 在 RX 窗口检测到包，芯片会给主机输出 RX\_DONE 中断并切到 STBY\_RC 模式。
- 在 RX 窗口期间，主机发送 *SetStandby(...)* 命令 (SLEEP 模式下，芯片不能接收命令，需要先被 NSS 下降沿唤醒)

SLEEP 模式的时间长度如下定义：

$$\text{Sleep Duration} = \text{sleepPeriod} * 15.625 \mu\text{s}$$

RX 模式的时间长度如下定义：

$$\text{Rx Duration} = \text{rxPeriod} * 15.625 \mu\text{s}$$

下图显示 RxDutyCycle 模式下进行的操作。可以看出芯片会有大约 1ms 用来保存配置，切换到 SLEEP 模式，重新初始化芯片、锁定 PLL、进入 RX 模式。这个 1ms 延时不是精确的，根据启动 XTAL，PLL 锁定等需要的时间会有差异。

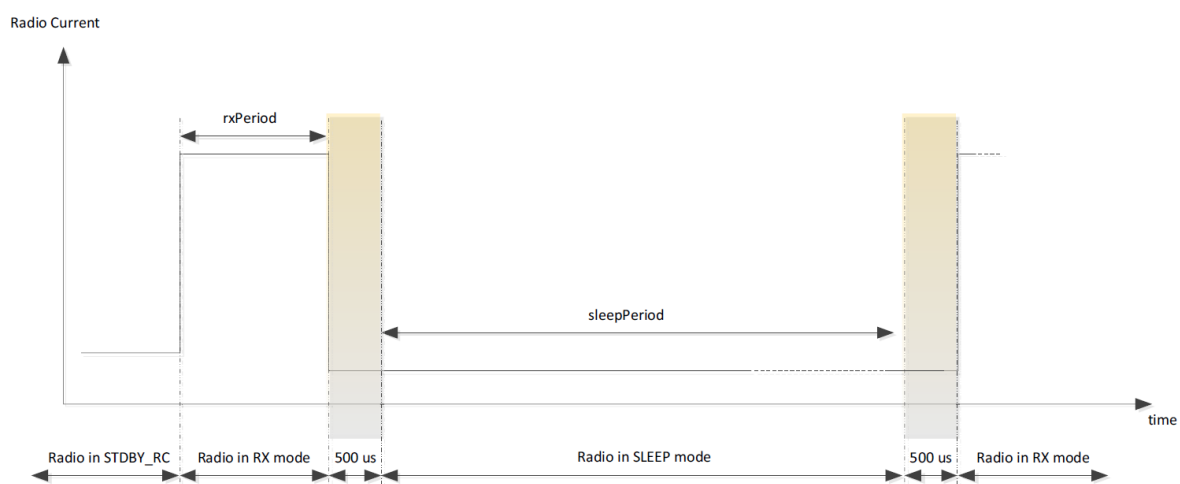


图 13- 2: RX DUTY CYCLE 模式下功耗情况

当检测到前导，芯片会寻找同步字（GFSK）或包头（LoRa®），计时器会重新启动，设置时间为  $2 * \text{rxPeriod} + \text{sleepPeriod}$ 。这是为了保证芯片不会无休止的等待接收数据包（前导误检测）。

包在空中时间与芯片消耗在 RX 模式和 SLEEP 模式的时间有很重要的关系，如果在发送端使用了较长的前导码，要注意需要满足如下公式：

$$T_{\text{preamble}} + T_{\text{header}} \leq 2 * \text{rxPeriod} + \text{sleepPeriod}$$

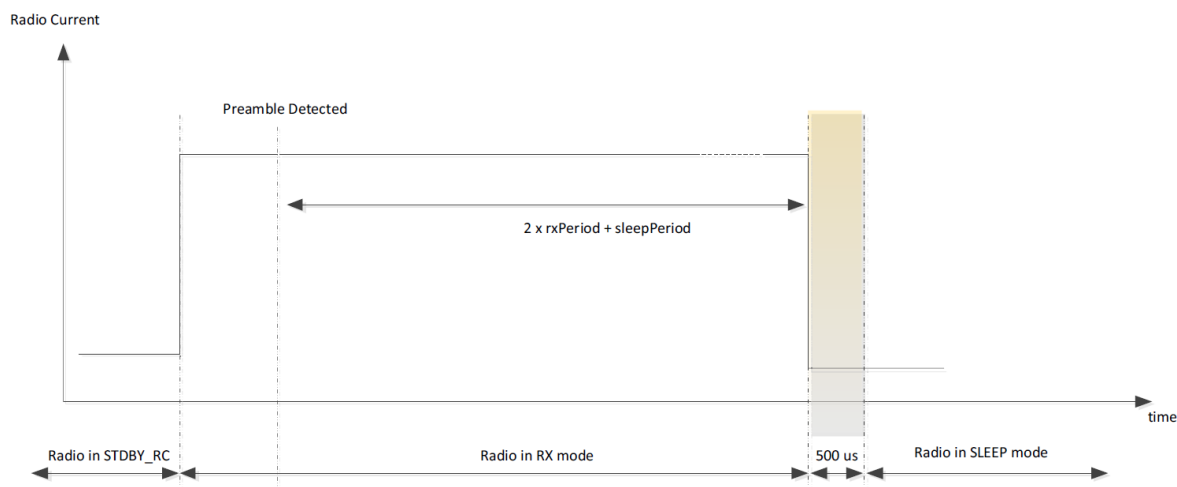


图 13-3: 接收时的 RX DUTY CYCLE

### 13.1.8. SetCAD

*SetCAD()*命令只能用于 LoRa®包类型，信道状态检测（Channel Activity Detection）是一种针对 LoRa®的模式，它的作用是让芯片检测 LoRa®前导。当检测完成时，设备会回到 STDBY\_RC 模式。检测的时间长度是由 *SetCadParams(...)*来设置。检测结束时会有 *CADdone* 中断产生（如果被启用的话）。如果检测到前导，它还会生成 *CadDetected* 中断。

表 13-13: SETCAD SPI 交互

字节	0
主机下发的数据	Opcode = 0xC5

### 13.1.9. SetTxContinuousWave

*SetTxContinuousWave()*是一个用于所有包类型的测试命令，用来在指定频率和输出功率下生成载波。芯片会保持在 TX 连续波状态，直到主机发送模式切换命令。

表 13-14: SETTXCONTINUOUSWAVE SPI 交互

字节	0
主机下发的数据	Opcode = 0xD1

虽然这个命令在实际应用中没有用处，但是它可以帮助开发者查看发送模式下的芯片性能。

### 13.1.10. SetTxInfinitePreamble

SetTxInfinitePreamble() 是一个测试命令，在 FSK 调制方式时，用来生成无限长的 0、1 交替的序列；在 LoRa®调制方式时，芯片只能持续调制 LoRa®前导码元。芯片会持续发送无限长的前导，直到主机发送模式切换命令。

虽然这个命令在实际应用中沒有实际用处，它可以帮助开发者查看发送模式下的芯片性能。

表 13- 15: SENDTXINFINITEPREAMBLE SPI 交互

字节	0
主机下发的数据	Opcode = 0xD2

使用这个功能时，不能设定芯片发送的数据内容。在 LoRa®模式下，只能生成 LoRa 前导码元；在 FSK 模式下，只能生成 FSK 前导 (0x55)。然而，最终用户将能够很容易地监测其调制参数对频谱影响。

### 13.1.11. SetRegulatorMode

默认情况下，只使用 LDO。这在一些不能接受使用 DC-DC 带来的额外成本开销的低成本应用中是有帮助的。只使用 LDO 意味着接收和发送的电流会翻倍。这个命令允许指定 DC-DC 还是 LDO 用来做电源转换。电源转换模式由参数 regModeParam 定义。

#### 注意：

这个功能显然是和芯片的硬件设计相关，用户使用这个命令时要确保知道硬件层面是如何实现的。

表 13- 16: SETREGULATORMODE SPI 交互

字节	0	1
主机下发的数据	Opcode= 0x96	regModeParam 0: 芯片在所有模式下只使用LDO 1: STBY_XOSC、FS、RX 和 TX 模式下使用 DC_DC+LDO

### 13.1.12. Calibrate

当上电时，芯片进行 RC64k，RC13M，PLL 和 ADC 校准。任何时候，可以在 STDBY\_RC 模式下对一个或多个模块进行校准。校准功能对 *calibParam* 参数定义的模块进行校准。

表 13- 17: CALIBRATE SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x89	calibParam

如果对所有模块进行校准，总的校准时间是 3.5ms。校准应当在 STDBY\_RC 模式下进行，在校准时 BUSY 引脚会处于高电平。BUSY 引脚上的下降沿意味着校准结束。

表 13- 18: 校准参数设置

CalibParam	校准设置
Bit 0	0: RC64k不校准 1: RC64k 校准
Bit 1	0: RC13M不校准 1: RC13M校准
Bit 2	0: PLL不校准 1: PLL校准
Bit 3	0: ADC pulse不校准 1: ADC pulse校准
Bit 4	0: ADC bulk N不校准 1: ADC bulk N校准
Bit 5	0: ADC bulk P不校准 1: ADC bulk P校准
Bit 6	0: Image不校准 1: Image校准
Bit 7	0: RFU

### 13.1.13. CalibrateImage

*CalibrateImage(...)* 允许用户校准芯片在工作频段的镜像抑制。

表 13- 19: CALIBRATEIMAGE SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x98	freq1

关于特定频段的更详细信息，请参见 55 页 [9.2.1 节 "指定频段的镜像校准"](#)。

## 13.1.14. SetPaConfig

*SetPaConfig* 命令用来区分 SX1261 和 SX1262。用户使用这个命令来选择芯片使用的 PA 及其配置。

表 13- 20: SETPACONFIG SPI 交互

字节	0	1	2	3	4
主机下发的数据	Opcode = 0x95	paDutyCycle	hpMax	deviceSel 0: SX1262 1: SX1261	paLut reserved and always 0x01

*paDutyCycle* 用来控制 PA (包括 SX1261 和 SX1262) 的占空比 (导通角)。最大输出功率、功耗和谐波会随着 *paDutyCycle* 的配置变化有明显差异。这个数据手册所提供的建议值可以实现 PA 最佳效率。改变 *paDutyCycle* 将影响谐波功率的分布, 因此必须根据匹配的网络进行选择。

*hpMax* 用来选择 SX1262 的 PA 尺寸, 它对 SX1261 没有影响。降低 *hpMax* 的值可以减小最大输出功率。有效范围是介于 0x00 和 0x07 之间, 0x07 是 SX1262 支持的最大值, 可以达到 +22dBm 输出功率。使用高于 0x07 的 *hpMax* 值会导致芯片老化加快, 在一些极端温度下可能损坏芯片。

*deviceSel* 用来选择 SX1261 还是 SX1262。

*paLut* 是保留字节, 其值始终为 0x01。

## 13.1.14.1 PA最优设置

PA 最优设置用来最大化 PA 效率, 当要求的输出功率低于 +22 dBm (SX1262) 或 +14/15 dBm (SX1261)。例如, 在日本最大输出功率是 +10dBm, 在中国某些频段是 +17dBm。这些最优配置要求:

- 专用匹配网络/PA 负载阻抗
- 对 PA 配置做特定调整, 详见下表:

表 13- 21: PA 最优配置

Mode	Output Power	paDutyCycle	hpMax	deviceSel	paLut	Value in SetTxParams <sup>1</sup>
SX1261	+15 dBm	0x06	0x00	0x01	0x01	+14 dBm
	+14 dBm	0x04	0x00	0x01	0x01	+14 dBm
	+10 dBm	0x01	0x00	0x01	0x01	+13 dBm
SX1262	+22 dBm	0x04	0x07	0x00	0x01	+22 dBm
	+20 dBm	0x03	0x05	0x00	0x01	+22 dBm
	+17 dBm	0x02	0x03	0x00	0x01	+22 dBm
	+14 dBm	0x02	0x02	0x00	0x01	+14 dBm

1. 详见 83 页 [13.4.4 节 “SetTxParams”](#)

**注意：**

这些改变会使得在使用标称功率时，要么不是最优的，要么是不可实现的。

**警告！**

下面几项限制需要特别注意，以防止对 PA 产生过压，电压超过额定值可能对器件造成不可恢复的损伤。

- 当 SX1261 工作频率 超过 400MHz 时，*paDutyCycle* 不可高于 0x07；
- 当 SX1261 工作频率 低于 400MHz 时，*paDutyCycle* 不可高于 0x04；
- 对于 SX1262，*paDutyCycle* 不可高于 0x04

### 13.1.15. SetRxTxFallbackMode

*SetRxTxFallbackMode* 命令设定当芯片完成发送或接收后切换到什么模式。

表 13- 22: SETRXTXFALLBACKMODE SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x93	fallbackMode

fallbackMode 字节定义如下：

表 13- 23: FALLBACKMODE 定义

fallbackMode	值	描述
FS	0x40	芯片在发送和接收之后切换到FS模式
STDBY_XOSC	0x30	芯片在发送和接收之后切换到STDBY_XOSC模式
STDBY_RC	0x20	芯片在发送和接收之后切换到STDBY_RC模式

默认情况下，芯片会回到 STDBY\_RC，除非用这个命令修改了配置。将该模式改为 STDBY\_XOSC 或 FS 只影响了芯片的切换时间。

## 13.2 寄存器和缓存区访问

### 13.2.1. WriteRegister 函数

*WriteRegister(...)* 允许在存储空间的指定起始地址写入一串的字节，每写入一个字节数据后地址会自动增加，这样数据会保存到连续的存储区域。SPI 传输的数据格式如下表。

表 13- 24: WRITEREGISTER SPI 交互

字节	0	1	2	3	4	...	n
主机下发的数据	Opcode = 0x0D	address[15:8]	address[7:0]	data@address	data@address+1	...	data@address+(n-3)
发给主机的数据	RFU	Status	Status	Status	Status	...	status

### 13.2.2. ReadRegister 函数

*ReadRegister(...)* 允许从指定地址读取一连串的数据。读取每一字节后地址会自动增加。SPI 传输的数据形式如下表所示。注意，主机必须在发送两字节的地址后发送一个 NOP，然后可以开始进行下一个 NOP 触发的数据接收。

表 13- 25: READREGISTER SPI 交互

字节	0	1	2	3	4	5	...	n
主机下发的数据	Opcode = 0x1D	address[15:8]	address[7:0]	NOP	NOP	NOP	...	NOP
发给主机的数据	RFU	Status	Status	Status	data@address	data@address+1	...	data@address+(n-4)

### 13.2.3. WriteBuffer 函数

这个函数是用来保存要发送的数据。地址会自动增加，由于数据缓冲区是以环状形式组织，当地址超过 255 会回到 0。该函数会以参数形式传入一个偏移值（offset），它作为数据缓冲区写入的起始地址。下表中描述了 SPI 数据传输格式。

表 13- 26: WRITEBUFFER SPI 交互

字节	0	1	2	3	...	n
主机下发的数据	Opcode = 0x0E	offset	data@offset	data@offset+1	...	data@offset+(n-2)
发给主机的数据	RFU	Status	Status	Status	...	Status

### 13.2.4. ReadBuffer 函数

该函数用来读取 (n-3) 字节的数据，读取的起始地址是 offset。注意，发送完 offset 后必须多发一个 NOP。

表 13- 27: READBUFFER SPI 交互

字节	0	1	2	3	4	...	n
主机下发的数据	Opcode = 0x1E	offset	NOP	NOP	NOP	...	NOP
发给主机的数据	RFU	Status	Status	data@offset	data@offset+1	...	data@offset+(n-3)

## 13.3 DIO 和 IRQ 控制函数

### 13.3.1. SetDioIrqParams

这个命令是用来设置 IRQ 标志位。

表 13- 28: SETDIOIRQPARAMS SPI 交互

字节	0	1-2	3-4	5-6	7-8
主机下发的数据	SetDioIrqParams (0x08)	Irq Mask(15:0)	DIO1Mask(15:0)	DIO2Mask(15:0)	DIO3Mask(15:0)

### 13.3.2. IrqMask

IrqMask 用来屏蔽或取消屏蔽芯片触发的 IRQ。默认情况下，所有 IRQ 是处于被屏蔽状态（全 0），用户可以通过将相应位置 1，将 IRQ 一个个或多个一起启用。

#### 13.3.2.1 DioxMask

如果 DioxMask 和 IrqMask 相应位都置 1，产生的中断信号会触发 DIO 置高。例如，IrqMask 和 DIO1Mask 的第 0 位都置 1，TxDone 中断信号产生的上升沿在 IRQ 寄存器会有记录，同时也会在 DIO1 端口输出。

一个 IRQ 可以映射到所有 DIOs，一个 DIO 可以映射为所有 IRQs（用 OR 操作）。但是有些 IRQ 只在特定芯片模式及包类型下才有效。

根据所选择的帧和芯片模式，总共有十个中断源。每一个都可以启用或屏蔽。另外，每一个中断都可以映射到 DIO1/DIO2/DIO3。但要注意的是，如果 DIO2 和 DIO3 用来控制射频开关或者 TCXO，即使把 IRQ 映射到这两个引脚，IRQ 也不会产生。

**表 13- 29: IRQ 寄存器**

字节	0	1-2	3-4
位	IRQ	描述	协议
0	TxDone	发包完成	All
1	RxDone	收包完成	All
2	PreambleDetected	检测到前导码	All
3	SyncWordValid	检测到有效的同步字	FSK
4	HeaderValid	收到有效的LoRa包头	LoRa®
5	HeaderErr	LoRa包头CRC错误	LoRa®
6	CrcErr	收到错误的CRC	All
7	CadDone	完成信道活动检测	LoRa®
8	CadDetected	检测到信道活动	LoRa®

一个 10 位的寄存器 IRQ\_reg 专门用来记录 IRQ 源，各个位对应某一个上表中描述的 IRQ 源。有一系列的命令用来配置 IRQ 屏蔽标志、DIO 映射、IRQ 清空，在下面的章节会介绍。

### 13.3.3. GetIrqStatus

这个命令用来获取 IRQ 寄存器的值。

**表 13- 30: GETIRQSTATUS SPI 交互**

字节	0	1	2-3
主机下发的数据	Opcode = 0x12	NOP	NOP
发给主机的数据	RFU	Status	IrqStatus(15:0)

### 13.3.4. ClearIrqStatus

这个命令用来清除 IRQ 寄存器的 IRQ 标志位。

表 13- 31: CLEARIRQSTATUS SPI 交互

字节	0	1-2
主机下发的数据	Opcode = 0x02	ClearIrqParam(15:0)

这个函数根据 *ClearIrqParam* 中为 1 的位，按位对应将 IRQ 寄存器的 IRQ 标志清零。例如，*ClearIrqParam* 的第 0 位为 1，则 IRQ 寄存器的 IRQ 标志位第 0 位被清零。

如果一个 DIO 只映射到某一个 IRQ 源，如果 IRQ 寄存器中对应位清零，该 DIO 也会被清零。如果 DIO 映射到多个 IRQ 源，只有所有关联的 IRQ 寄存器位置 0，该 DIO 才会清零。

### 13.3.5. SetDIO2AsRfSwitchCtrl

这个命令用来配置 DIO2，使它用来控制外部的射频开关。

表 13- 32: SETDIO2ASRFSWITCHCTRL SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x9D	enable

当 DIO2 用来控制外部的射频开关，它的状态会和内部状态机直接相关。DIO2 会在 PA 电压上升之前几毫秒内输出高电平，在 PA 电压下降后回到低电平。

该命令的 enable 字节如下定义：

表 13- 33: ENABLE 配置项定义

Enable	描述
0	DIO2 可以用来映射 IRQ
1	DIO2 用来控制射频开关。在 SLEEP、STDBY_RX、STDBY_XOSC、FS 和 RX 模式下 DIO2 = 0；在 TX 模式下 DIO2 = 1

### 13.3.6. SetDIO3AsTCXOCtrl

这个命令用来设置外部 TCXO 参考电压，该参考电压由芯片的 DIO3 来控制。

表 13- 34: SETDIO3ASTCXOCTRL SPI 交互

字节	0	1	2-4
主机下发的数据	Opcode = 0x97	tcxoVoltage	timeout(23:0)

当使用这个命令，芯片通过 DIO3 控制 TCXO。当需要时（在模式 STDBY\_XOSC、FS、TX 和 RX），内部状态机会将 DIO3 设置在定义好的输出电压（通过参数 *tcxoVoltage* 控制）。内部时钟控制器会等待 32MHz 出现，才释放内部状态机。

32MHz 出现和稳定所需要的时间可以通过参数 *timeout* 来检测。如果在 *timeout* 时间结束时还没有检测到 TCXO 输出的 32MHz 时钟，错误控制器会标志 XOSC\_START\_ERR 错误。

*tcxoVoltage* 字节定义如下：

表 13- 35: TCXOVOLTAGE 配置定义

tcxoVoltage	描述
0x00	DIO3 输出 1.6 V 给 TCXO
0x01	DIO3 输出 1.7 V 给 TCXO
0x02	DIO3 输出 1.8 V 给 TCXO
0x03	DIO3 输出 2.2 V 给 TCXO
0x04	DIO3 输出 2.4 V 给 TCXO
0x05	DIO3 输出 2.7 V 给 TCXO
0x06	DIO3 输出 3.0 V 给 TCXO
0x07	DIO3 输出 3.3 V 给 TCXO

*tcxoVoltage* 的电源变换器被设置为比电源电压低 200mV。即使 *tcxoVoltage* 被配置为高于电源电压，电源电压也会受到下式限制： $VDDop > VTCXO + 200\text{ mV}$ 。

*timeout* 时间长度定义为：

$$Timeout\ duration = Timeout * 15.625\ \mu s$$

多数 TCXO 不会立即就绪到指定的频率，它会经历一段初始化时间，这期间频率会与所需的频率有一些偏移。不同厂商生产的 TCXO 的初始化时间会不一样。为了确保 TXCO 的建立时间对调制解调器或数据包没有任何影响，通过配置超时值以给予 TCXO 足够的时间使其频率稳定。在超时周期结束时，内部模块将不检测时钟，并且芯片将继续到下一步骤。

#### 注意：

当芯片从 STDBY\_RC 模式进入 TX 或 RX 模式，需要考虑该 *timeout* 时间。模式切换所用的时间会因该值增加而增加。为了避免增加模式切换时间，可以先将芯片设置在 STDBY\_XOSC，该模式会使能 TCXO 并等待 *timeout*。然后再设置为 TX 或 RX 模式，其内部处理就不会有额外的延时。

## 13.4 射频调制和包相关函数

### 13.4.1. SetRfFrequency

*SetRfFrequency(...)*命令用来设置射频频率。

表 13- 36: SETRF\_FREQUENCY SPI 交互

字节	0	1-4
主机下发的数据	Opcode = 0x86	RfFreq(31:0)

Freq的最低有效位等于锁相环/PLL的步进，频率计算公式如下：

$$RF_{frequency} = \frac{RF_{Freq} * F_{XTAL}}{2^{25}}$$

*SetRfFrequency(...)*定义了芯片在FS、TX和RX模式下的频率。在RX模式，频率被降至中频（默认中频是250 kHz）。

### 13.4.2. SetPacketType

*SetPacketType(...)*命令设置 SX1261 芯片在 LoRa®或 FSK 模式。在芯片的配置命令序列中 *SetPacketType(...)*命令必须是第一个执行。这个命令的参数是 PacketType。

表 13- 37: SET\_PACKETTYPE SPI 交互

字节	0	1
主机下发的数据	Opcode = 0x8A	PacketType

表 13- 38: PACKETTYPE 定义

PacketType	值	调制模式
PACKET_TYPE_GFSK	0x00	GFSK
PACKET_TYPE_LORA	0x01	LORA

用 *SetPacketType(...)*命令完成从一个模式到另一模式的切换。上一个模式的配置参数不会保留。从一帧到另一帧的切换必须在 STDBY\_RC 模式下进行。

### 13.4.3. GetPacketType

*GetPacketType()*命令返回当前芯片的包类型。

表 13- 39: GETPACKETTYPE SPI 交互

字节	0	1	2
主机下发的数据	Opcode = 0x11	NOP	NOP
发给主机的数据	RFU	Status	packetType

### 13.4.4. SetTxParams

这个命令使用参数 *power* 设置输出功率，用参数 *RampTime* 来设置 TX 的 Ramping 时间。这个命令适用于所有协议。

表 13- 40: SETTXPARAMS SPI 交互

字节	0	1	2
主机下发的数据	Opcode = 0x8E	power	RampTime

输出功率参数由 *power* ( 单位为 dBm ) 定义，其范围为：

- - 17 (0xEF) 到 +14 (0x0E) dBm ( 1dB步长 )，如果选择低功率PA
- - 9 (0xF7) 到 +22 (0x16) dBm ( 1dB步长 )，如果选择高功率PA

高功率PA和低功率PA的选择用 *SetPaConfig* 命令完成，其参数是 *deviceSel*。默认情况下的配置是低功率PA，设置为+14 dBm。

功率 Ramp 时间由参数 *RampTime* 定义，参见下表：

表 13- 41: RAMP TIME 定义

RampTime	值	RampTime (μs)
SET_RAMP_10U	0x00	10
SET_RAMP_20U	0x01	20
SET_RAMP_40U	0x02	40
SET_RAMP_80U	0x03	80
SET_RAMP_200U	0x04	200
SET_RAMP_800U	0x05	800
SET_RAMP_1700U	0x06	1700
SET_RAMP_3400U	0x07	3400

### 13.4.5. SetModulationParams

*SetModulationParams(...)*命令用来设置芯片的调制参数。根据之前设置的包的类型，芯片对这些参数会有不同的解读。

表 13- 42: SETMODULATIONPARAMS SPI 交互

字节	0	1	2	3	4	5	6	7	8
主机下发的调制参数	Opcode 0x8B	Mod Param1	Mod Param2	Mod Param3	Mod Param4	Mod Param5	Mod Param6	Mod Param7	Mod Param8

这些参数的意义取决于所选择的协议类型。

在FSK模式，比特速率 (BR)和频偏 (Fdev) 用在发送和接收设置中，而带宽 (BW) 只用在接收设置。脉冲成型 (pulseShape) 表示在发射机端使用了高斯滤波器来过滤调制数据流。

在LoRa®模式，扩频因子 (SF) 用于LoRa®调制，它在参数 *Param[1]*中定义。带宽(BW)指LoRa信号扩展的带宽，它在参数 *Param[2]*中定义。

LoRa®包的有效载荷是经过前向纠错机制处理，它有若干级别的编码率，编码率 (CR) 在参数中 *Param[3]*定义。

参数 *LdOpt*用于配置低数据率优化 (LDRO)，这个参数用在当单个码元传输时间大于等于 16.38ms 时 (典型的有 SF11/BW125、SF12/BW125、SF12/BW250)。详见 39 页 [6.1.1.4 节 "低数据率优化"](#)。

#### 13.4.5.1 GFSK调制参数

下表提供了 GFSK 更详细的参数：

表 13- 43: GFSK MODPARAM1, MODPARAM2 & MODPARAM3 – BR

BR(23:0)	描述
0x000001 to 0xFFFFFFFF	$br = 32 * F_{xtal} / \text{bit rate}$

比特速率由参数 *br*输入，它与主振荡器 (32MHz) 的频率相关。比特速率的范围由600b/s到300kb/s，其默认值是4.8kb/s。

表 13- 44: GFSK MODPARAM4 – PULSESHAPE

PulseShape	描述
0x00	No Filter applied
0x08	Gaussian BT 0.3
0x09	Gaussian BT 0.5
0x0A	Gaussian BT 0.7
0x0B	Gaussian BT 1

**表 13- 45: GFSK MODPARAM5 - BANDWIDTH**

Bandwidth	描述
0x1F	RX_BW_4800 (4.8 kHz DSB)
0x17	RX_BW_5800 (5.8 kHz DSB)
0x0F	RX_BW_7300 (7.3 kHz DSB)
0x1E	RX_BW_9700 (9.7 kHz DSB)
0x16	RX_BW_11700 (11.7 kHz DSB)
0x0E	RX_BW_14600 (14.6 kHz DSB)
0x1D	RX_BW_19500 (19.5 kHz DSB)
0x15	RX_BW_23400 (23.4 kHz DSB)
0x0D	RX_BW_29300 (29.3 kHz DSB)
0x1C	RX_BW_39000 (39 kHz DSB)
0x14	RX_BW_46900 (46.9 kHz DSB)
0x0C	RX_BW_58600 (58.6 kHz DSB)
0x1B	RX_BW_78200 (78.2 kHz DSB)
0x13	RX_BW_93800 (93.8 kHz DSB)
0x0B	RX_BW_117300 (117.3 kHz DSB)
0x1A	RX_BW_156200 (156.2 kHz DSB)
0x12	RX_BW_187200 (187.2 kHz DSB)
0x0A	RX_BW_234300 (232.3 kHz DSB)
0x19	RX_BW_312000 (312 kHz DSB)
0x11	RX_BW_373600 (373.6 kHz DSB)
0x09	RX_BW_467000 (467 kHz DSB)

**表 13- 46: GFSK MODPARAM6, MODPARAM7 & MODPARAM8 - FDEV**

Fdev(23:0)	描述
0x000000 to 0xFFFFF	Fdev = (Frequency Deviation * 2 <sup>25</sup> ) / Fxtal

$$Frequencydeviation = \frac{F_{dev} * F_{XTAL}}{2^{25}}$$

### 13.4.5.2 LoRa 调制参数

下表提供了关于 LoRa 调制的 更详细的参数：

**表 13- 47: LORA® MODPARAM1- SF**

SF	描述
0x05	SF5
0x06	SF6
0x07	SF7
0x08	SF8
0x09	SF9
0x0A	SF10
0x0B	SF11
0x0C	SF12

**表 13- 48: LORA@ MODPARAM2 – BW**

BW	描述
0x00	LORA_BW_7 (7.81 kHz real)
0x08	LORA_BW_10 (10.42 kHz real)
0x01	LORA_BW_15 (15.63 kHz real)
0x09	LORA_BW_20 (20.83 kHz real)
0x02	LORA_BW_31 (31.25 kHz real)
0x0A	LORA_BW_41 (41.67 kHz real)
0x03	LORA_BW_62 (62.50 kHz real)
0x04	LORA_BW_125 (125 kHz real)
0x05	LORA_BW_250 (250 kHz real)
0x06	LORA_BW_500 (500 kHz real)

**表 13- 49: LORA@ MODPARAM3 - CR**

CR	描述
0x01	LORA_CR_4_5
0x02	LORA_CR_4_6
0x03	LORA_CR_4_7
0x04	LORA_CR_4_8

表 13- 50: LoRa® MODPARAM4 – LOWDATA RATE OPTIMIZE

LowDataRateOptimize	描述
0x00	低数据率优化关
0x01	低数据率优化开

### 13.4.6. SetPacketParams

该命令用于设置数据包处理块的参数。

表 13- 51: SETPACKETPARAMS SPI TRANSACTION

Byte	0	1	2	3	4	5	6	7	8	9
主机下发的数据	Opcode= 0x8C	packet Param1	packet Param2	packet Param3	packet Param4	packet Param5	packet Param6	packet Param7	packet Param8	packet Param9

#### 13.4.6.1 GFSK 包参数

下面的表格提供更多 GFSK 包参数的细节。

表 13- 52: GFSK PACKETPARAM1 &amp; PACKETPARAM2 - PREAMBLELENGTH

PreambleLength (15:0)	描述
0x0001 to 0xFFFF	发送前导长度，单位 bit

前导长度是一个 16 位的数值，表示发送前导的字节个数。每个前导字节是由 0 和 1 交错编码组成的字节 0x55。

表 13- 53: GFSK PACKETPARAM3 - PREAMBLEDETECTORLENGTH

PreambleDetector	描述
0x00	前导检查器长度关闭
0x04	前导检查器长度 8 位
0x05	前导检查器长度 16 位
0x06	前导检查器长度 24 位
0x07	前导检查器长度 32 位

前导检测器作为包控制器的门控，当配置值为非 0x00（前导检测器关闭）时，如果接收机的前导检测器成功接收到设定的前导位数，则包控制器会被激活。

表 13- 54: GFSK PACKETPARAM4 - SYNCWORDLENGTH

SyncWordLength	描述
0x00 ~ 0x40	同步字长度, 单位 bit, (范围 0~8 字节)

同步字可通过简单访问寄存器的方式直接写入到芯片，下面表格是同步字寄存器地址。

表 13- 55: 同步字地址

Sync Word	Register Address
Byte 0	0x06C0
Byte 1	0x06C1
Byte 2	0x06C2
Byte 3	0x06C3
Byte 4	0x06C4
Byte 5	0x06C5
Byte 6	0x06C6
Byte 7	0x06C7

表 13- 56: GFSK PACKETPARAM5 – ADDRCOMP

AddrComp	描述
0x00	关闭地址过滤
0x01	激活节点的地址过滤
0x02	激活节点和广播的地址过滤

通过访问寄存器将节点地址和广播地址直接写入到芯片中，下面表格提供寄存器地址。

表 13- 57: 节点地址

	寄存器地址	默认值
NodeAddrReg	0x06CD	0x00

**表 13- 58: 广播地址**

	寄存器地址	默认值
NodeAddrReg	0x06CD	0x00

**表 13- 59: GFSK PACKETPARAM6 - PACKETTYPE**

PacketType	描述
0x00	固定包长, 收发两边都知道包长度, 长度信息不包含在包内容中.
0x01	可变包长, 有效载荷的第一个字节是数据包长度信息

**表 13- 60: GFSK PACKETPARAM7 - PAYLOADLENGTH**

AddrComp	描述
0x00 to 0xFF	发送载荷的长度(字节)或者最大接收载荷的长度

**表 13- 61: GFSK PACKETPARAM8 - CRCTYPE**

CRCTYPE	描述
0x01	CRC_OFF (无 CRC)
0x00	CRC_1_BYTE (以 1 个字节计算 CRC)
0x02	CRC_2_BYTE(以 2 个字节计算 CRC)
0x04	CRC_1_BYTE_INV(以 1 个字节计算 CRC 并反转)
0x06	CRC_2_BYTE_INV(以 2 个字节计算 CRC 并反转)

在 SX1261 和 SX1262 中，CRC 可以被完全配置和使用不同的多项式，同时初始值也可以直接通过寄存器配置。

**表 13- 62: CRC 初始值**

	寄存器地址	默认值
CRC MSB Initial Value [15:8]	0x06BC	0x1D
CRC LSB Initial Value [7:0]	0x06BD	0x0F

**表 13- 63: CRC 多项式值**

	寄存器地址	默认值
CRC MSB polynomial value [15:8]	0x06BE	0x10
CRC LSB polynomial value [7:0]	0x06BF	0x21

**表 13- 64: GFSK PACKETPARAM9 白化**

AddrComp	描述
0x00	无编码
0x01	使能白化编码

**表 13- 65: 白化初始值设置**

白化初始值	寄存器地址
白化初始值 MSB	0x06B8
白化初始值 LSB	0x06B9

### 13.4.6.2 LoRa数据包参数

下表提供了 LoRa®数据包参数的更多细节：

**表 13- 66: LORA@PACKETPARAM1 & PACKETPARAM2 - PREAMBLELENGTH**

PreambleLength (15:0)	描述
0x0001 ~ 0xFFFF	前导长度: 发送前导码元的数量

**表 13- 67: LoRa® PACKETPARAM3 – HeaderType**

HeaderType	描述
0x00	可变包长(显式包头)
0x01	固定包长(隐式包头)

当 HeaderType 为 0x00 时，包长度、编码率和 CRC 会加入到 LoRa 的包头中并发送到接收机。

**表 13- 68: LoRa® PACKETPARAM4 – PayloadLength**

Payloadlength	描述
0x00 ~ 0xFF	发送机发送载荷的长度, 或者接收机能接受的最大载荷长度(字节)

**表 13- 69: LoRa® PACKETPARAM5 - CRCTYPE**

CRCTYPE	描述
0x00	CRC 关闭
0x01	CRC 打开

**表 13- 70: LoRa® PACKETPARAM6 - INVERTIQ**

AddrComp	描述
0x00	标准 IQ 设置
0x01	反转 IQ 设置

### 13.4.7. SetCadParams

SetCadConfig(...)这个命令定义 CAD 操作包含的码元数量。

**表 13- 71: SETCADPARAMS SPI TRANSACTION**

Byte	0	1	2	3	4	6-7
主机下发的数据	Opcode = 0x88	cadSymbolNum	cadDetPeak	cadDetMin	cadExitMode	cadTimeout(23:0)

使用码元的数量定义在下表中。

表 13- 72: CAD 码元数定义

cadSymbolNum	Value	Number of Symbols used for CAD
CAD_ON_1_SYMB	0x00	1
CAD_ON_2_SYMB	0x01	2
CAD_ON_4_SYMB	0x02	4
CAD_ON_8_SYMB	0x03	8
CAD_ON_16_SYMB	0x04	16

参数 *cadDetPeak* 和 *cadDetMin* 定义了 LoRa 调制解调器在实际应用中的灵敏度。这两个设置取决于 LoRa 扩频因子和带宽，也取决于用于验证或不检测的码元数。

表 13- 73: 使用 4 个码元检测 CAD 时，CADDETPEAK 和 CADDETMIN 的推荐设置

SF	cadDetPeak	cadDetMin
5	18	10
6	19	10
7	20	10
8	21	10
9	22	10
10	23	10
11	24	10
12	25	10

选择合适的值不容易，这个值必须小心的选择以保证良好的灵敏度检测，同时也限制错误检测的数量。

*cadExitMode* 参数定义了 CAD 操作后要执行的操作。这个是可选的。

表 13- 74: CAD 退出模式定义

cadExitMode	值	操作
CAD_ONLY	0x00	LoRa 模式下，芯片执行 CAD 操作。一旦操作完成，无论通道是否有检测到活动信号，芯片都会回到 STBY_RC 模式。
CAD_RX	0x01	芯片执行CAD操作并检查到信道有活动信号，则保留在RX模式直到检测到一包数据或者时间超时，时间定义: cadTimeout * 15.625 us

cadTimeout参数仅用于当CAD配置为cadExitMode=CAD\_RX时，这里的cadTimeout表示CAD成功后继续保持在Rx状态的时间。

$$Rx\ Timeout = cadTimeout * 15.625$$

### 13.4.8. SetBufferBaseAddress

这个命令用于设置 Tx 和 Rx 模式下包处理的数据缓存区的基地址。这些参数的用法和定义描述在不同的包类型章节中。

表 13- 75: SETBUFFERBASEADDRESS SPI TRANSACTION

Byte	0	1	2
主机下发的数据	Opcode = 0x8F	TX base address	RX base address

### 13.4.9. SetLoRaSymbNumTimeout

这个命令设置调制解调器用来验证成功接收时使用的码元数。

表 13- 76: SETLORASYMBNUMTIMEOUT SPI TRANSACTION

Byte	0	1
主机下发的数据	Opcode = 0xA0	SymbNum

在 LoRa 模式下，当进入 RX 模式，调制解调器一旦检测到 LoRa 码元就会锁定，这可能导致错误的检测。这种现象相当罕见但仍然有可能。为了避免这个问题，命令 SetLoRaSymbNumTimeout 定义了用于验证正确接收数据包需要的码元数。

当 SymbNum 参数设成 0 时，只要检测到 LoRa 码元，调制解调器就开始接收。

当 SymbNum 设成非 0 值时，调制解调器将会等待全部的 SymbNum 个 LoRa 码元来验证是否检测到正确的 LoRa 包。如果这时候解调器的各种状态都没有锁定，芯片将会产生 RxTimeout IRQ 信号。

### 13.5 通信状态信息

这些命令返回芯片状态信息，接收包信息，比如包长度，接收功率和包正确接收时的一些指示标志信息。这些返回的参数是不同于 LoRa 协议的。

#### 13.5.1. GetStatus

主机可通过命令 GetStatus 直接获取芯片状态，这个命令在任何时候都可以下发且会返回芯片状态。命令 GetStatus 不是严格需要的，因为芯片也会在命令字节上返回状态信息。返回的状态字节描述在表 13-77 中。

表 13-77: 状态字节定义

7	6:4	3:1	0
保留	芯片模式	命令状态	保留
-	0x0: 未用	0x0: 保留	-
	RFU	RFU	
	0x2: STBY_RC	0x2: Data is available to host <sup>1</sup>	
	0x3: STBY_XOSC	0x3: Command timeout <sup>2</sup>	
	0x4: FS	0x4: Command processing error <sup>3</sup>	
	0x5: RX	0x5: Failure to execute command <sup>4</sup>	
	0x6: TX	0x6: Command TX done <sup>5</sup>	

1. 成功接收数据包并且可以读取。
2. 一次跟主机的交互花费太长的时间来完成并触发了内部看门狗。这个看门狗机制可以被主机禁止，它的意义是确保所有的结果都可被主机标记。
3. 处理器无法处理命令可能是因为一个无效的操作码或者提供了错误的参数数量。
4. 命令已经成功处理，然而芯片不能执行命令；比如它不能进入特殊的芯片模式或者发送请求的数据
5. 当前包的传输已经终止。

命令GetStatus的SPI交互如下表所示。

表 13-78: GETSTATUS SPI TRANSACTION

Byte	0	1
主机下发的数据	Opcode = 0xC0	NOP
发给主机的数据	RFU	Status

### 13.5.2. GetRxBufferStatus

这个命令返回最新接收数据包的长度信息 ( PayloadLengthRx ) 和第一个接收字节的地址 ( RxStartBufferPointer )。它可以应用到所有模式。这地址是相对于数据缓存区的第一个字节的一个偏移地址。

表 13- 79: GETRXBUFFERSTATUS SPI TRANSACTION

Byte	0	1	2	3
主机下发的数据	Opcode = 0x13	NOP	NOP	NOP
发给主机的数据	RFU	Status	PayloadLengthRx	RxStartBufferPointer

### 13.5.3. GetPacketStatus

表 13- 80: GETPACKETSTATUS SPI TRANSACTION

Byte	0	1	2	3	4
主机下发的数据	Opcode = 0x14	NOP	NOP	NOP	NOP
发给主机的数据 ( FSK )	RFU	Status	RxStatus	RssiSync	RssiAvg
发给主机的数据 ( LoRa )	RFU	Status	RssiPkt	SnrPkt	SignalRssiPkt

下面的列表根据不同的包类型，给出不同的 RSSI 和 SNR 的表述。

表 13- 81: STATUS BIT

RSSI	描述
RxStatus FSK	bit 7: 前导错误 bit 6: 同步字错误 bit 5: 地址错误 bit 4: CRC错误 bit 3: 长度错误 bit 2: 终止错误 bit 1: 接收到数据包 bit 0: 发送完数据包
RssiSync FSK	RSSI值在检测同步地址时被锁定。 [负数, dBm, fixdt(0,8,1)] 实际的信号强度是: -RssiSync/2 (dBm)
RssiAvg FSK	接收载荷过程中的RSSI平均值, 在pkt_done IRQ时锁定 [负数, dBm, fixdt(0,8,1)] 实际的信号强度是: -RssiAvg/2 (dBm)

RssiPkt LoRa®	最后接收到的包RSSI的平均值 实际的信号强度是: $-RssiPkt/2$ (dBm)
SnrPkt LoRa®	最后一个接收包的SNR估值。二进制补码格式乘以4。 实际的SNR为 $SNR(dB) = SnrPkt/4$
SignalRssiPkt LoRa®	最后一个接收包在解扩之后RSSI值, 二进制补码格式 [负数, dBm, fixdt(0,8,1)] 实际的Rssi为 $Rssi(dB) = -SignalRssiPkt/2$

### 13.5.4. GetRssiInst

这个命令返回接收包期间的 RSSI 瞬时值，这个命令对所有的协议都有效。

表 13- 82: GETRSSIINST SPI TRANSACTION

Byte	0	1	2
主机下发的数据	Opcode = 0x15	NOP	NOP
发给主机的数据	RFU	Status	RssiInst 信号强度 dBm = $RssiInst/2$ (dBm)

### 13.5.5. GetStats

这个命令返回一些最后接收到的数据包的数量信息，这个命令对所有协议有效。

表 13- 83: GETSTATS SPI TRANSACTION

Byte	0	1	2-3	4-5	6-7
主机下发的数据	Opcode = 0x10	NOP	NOP	NOP	NOP
发给主机的数据(GFSK)	RFU	Status	NbPktReceived(15:0)	NbPktCrcError(15:0)	NbPktLengthError(15:0)
发给主机的数据(LoRa)	RFU	Status	NbPktReceived(15:0)	NbPktCrcError(15:0)	NbPktHeaderErr(15:0)

### 13.5.6. ResetStats

此命令重置由命令 GetStats 读取的值。为了执行这个命令，操作码是 0x0 并跟随 6 个 0 ( 总共 7 个 0 )

## 13.6 其它

### 13.6.1. GetDeviceErrors

这个命令返回可能的错误标志，该错误标志可能发生在不同的芯片操作中，描述如下。

表 13- 84: GETDEVICEERRORS SPI TRANSACTION

Byte	0	1	2-3
主机下发的数据	Opcode= 0x17	NOP	NOP
发给主机的数据	RFU	Status	OpError(15:0)

下表给出每个OpError位的含义。

表 13- 85: OPERROR BITS

OpError	0	1
bit 0	RC64K_CALIB_ERR	RC64k 校准失败
bit 1	RC13M_CALIB_ERR	RC13M 校准失败
bit 2	PLL_CALIB_ERR	PLL 校准失败
bit 3	ADC_CALIB_ERR	ADC 校准失败
bit 4	IMG_CALIB_ERR	IMG 校准失败
bit 5	XOSC_START_ERR	XOSC 启动失败
bit 6	PLL_LOCK_ERR	PLL 锁定失败
bit 7	RFU	RFU
bit 8	PA_RAMP_ERR	PA ramping 失败
bit 15:9	RFU	RFU

### 13.6.2. ClearDeviceErrors

这个命令清除芯片中的所有错误标志记录，这些错误标志不能够单独清理。

表 13- 86: CLEARDEVICEERRORS SPI TRANSACTION

Byte	0	1
主机下发的数据	Opcode= 0x07	0x00
发给主机的数据	RFU	Status

## 14. 应用

### 14.1 主机 API 基本读写功能

主机与SX1261/2的通信是围绕通用函数来组织的，这些函数允许用户控制芯片的行为。每个函数都是基于一个可操作命令（在整个文档称为“Opcode”），该操作码后面跟随着一组参数。SX1261/2使用BUSY引脚指示芯片的状态。在随后的章节中，假设主机有一个SPI口并通过spi.write(data)来访问，数据格式是8位字节。通过SPI片选信号为NSS，低有效。

### 14.2 基本 TX 操作的电路配置

这个章节描述了从上电到发送一帧数据的操作流程。

上电后（插入电池或者硬件复位），芯片自动运行校准流程并进入到 STDBY\_RC 模式，它完成的标志是 BUSY 引脚变低电平。这个状态后的接下来的步骤如下：

1. 如果芯片不在 STDBY\_RC 模式，则使用命令 SetStandby(...)进入 Standby 模式。
2. 使用命令 SetPacketType(...)定义包类型（LORA 或 FSK）
3. 使用命令 SetRfFrequency(...)定义射频频率。
4. 使用命令 SetTxParams(...)定义输出功率和 PA Ramping 时间。
5. 使用命令 SetBufferBaseAddress(...)定义有效载荷存储的位置。
6. 使用命令 WriteBuffer(...)发送有效载荷到数据缓存区。
7. 使用命令 SetModulationParams(...)根据选择的协议定义调制参数。
8. 使用命令 SetPacketParams(...)定义帧格式。
9. 配置 DIO 和 IRQ：使用命令 SetDioIrqParams(...)选择 TxDone IRQ 并映射这个 IRQ 到 DIO(DIO1,DIO2,DIO3)。
10. 定义同步字值：使用命令 WriteReg(...)通过直接寄存器访问来写入寄存器的值。
11. 使用命令 SetTx(...)设置电路在发送模式并开始发送，使用 TimeOut 参数设置超时值。
12. 等待 IRQ TxDone 或 TimeOut：一旦数据包发送完成，芯片自动进入 STDBY\_RC 模式。
13. 清除 IRQ TxDone 标志。

### 14.3 基本 RX 操作的电路配置

这个章节描述了从上电到接收一帧数据的操作流程，这流程对所有协议都有效。

上电后（插入电池或者硬件复位），芯片自动运行校准流程并进入到 STDBY\_RC 模式，它完成的标志是 BUSY 引脚变低电平。这个状态后接下来的步骤如下：

1. 如果芯片不在 STDBY\_RC 模式，则使用命令 SetStandby(...)进入 Standby 模式。
2. 使用命令 SetPacketType(...)定义包类型 ( LORA 或 FSK )
3. 使用命令 SetRfFrequency(...)定义射频频率。
4. 使用命令 SetBufferBaseAddress(...)定义接收数据的存储位置。
5. 使用命令 SetModulationParams(...)根据选择的协议定义调制参数。
6. 使用命令 SetPacketParams(...)定义帧格式。
7. 配置 DIO 和 IRQ：使用命令 SetDioIrqParams(...)选择 RxDone IRQ 并映射这个 IRQ 到 DIO(DIO1,DIO2,DIO3)。
8. 定义同步字值：使用命令 WriteReg(...)通过直接寄存器访问来写入寄存器的值。
9. 使用命令 SetRx(...)设置电路在接收模式，设置参数使能 timeout 或使能 continuous 模式。
10. 等待 IRQ RxDone 或 TimeOut：如果选择了 continuous 模式芯片会留在 Rx 模式并寻找新的数据包，否则返回到 STDBY\_RC 模式。
11. 如果收到 IRQ RxDone，使用命令 GetIrqStatus()检查状态位以保证 CRC 是正确的。

提示：收到 IRQ RxDone 信号意味着接收到一个数据包但是 CRC 有可能不正确，在确认数据包前用户必须检查 CRC。

12. 使用命令 CleanIrqStatus()清除 IRQ 标志 RxDone 或 TimeOut。如果收到有效包 ( CRC OK )，通过命令 GetRxBufferStatus(...)获取包长度和接收载荷的第一个地址。
13. 如果接收到有效包(CRC OK), 开始读取数据包。

### 14.4 按正确的顺序发送命令

除了设置芯片在合适工作模式的命令外，大多数的命令能够以任何的顺序发送。确实，在执行任何其他配置命令前，它强制要求使用命令SetPacketType(...)来设置芯片协议类型。然后用户应该根据选择的协议类型，使用命令SetModulationParams(...)定义调制参数。最后，用户应该使用命令SetPacketParams(...)选择数据帧格式。

#### 注意：

如果不遵循这些顺序要求，芯片的行为是不可预估的。

## 14.5 应用原理图

### 14.5.1. SX1261 带射频开关的应用设计

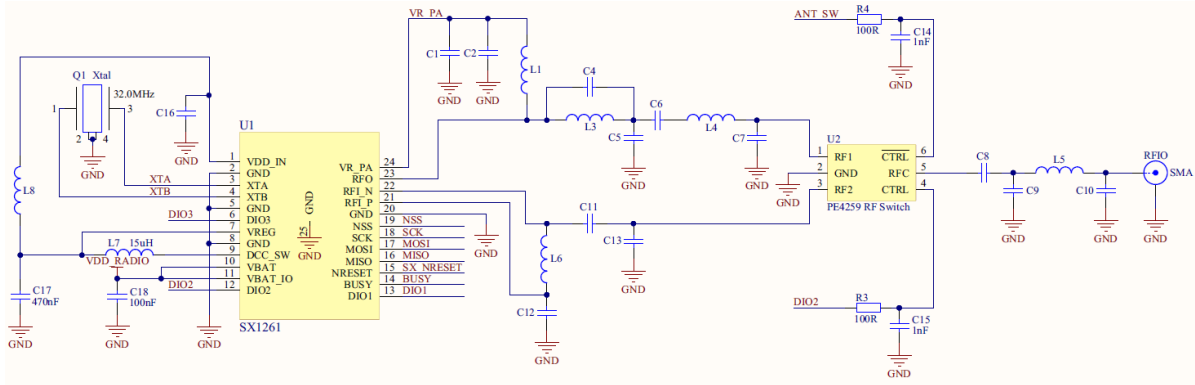


图 14- 1: SX1261 带射频开关的应用设计

### 14.5.2. SX1262 带射频开关的应用设计

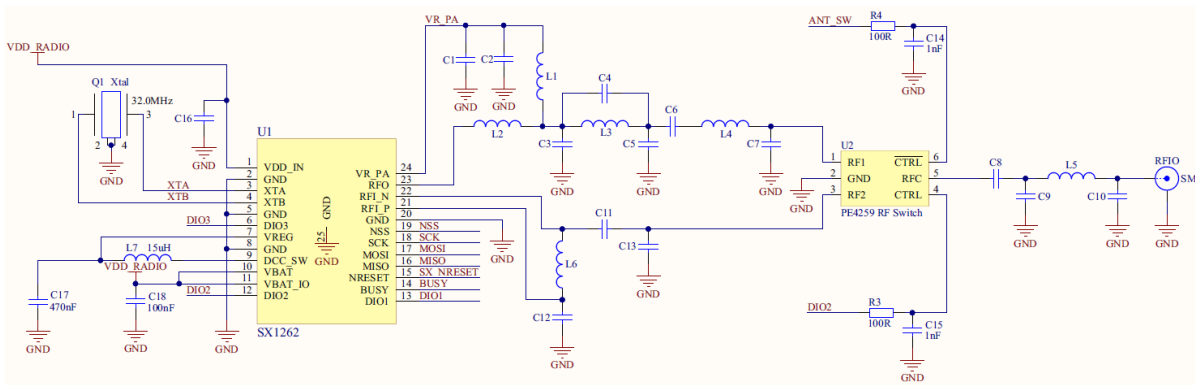


图 14- 2: SX1262 带 RF 开关的应用设计

注意：

这里展示的应用电路图仅作为参考信息。

请在官网上 [www.semtech.com](http://www.semtech.com) 参考最新的参考电路设计。

注意：

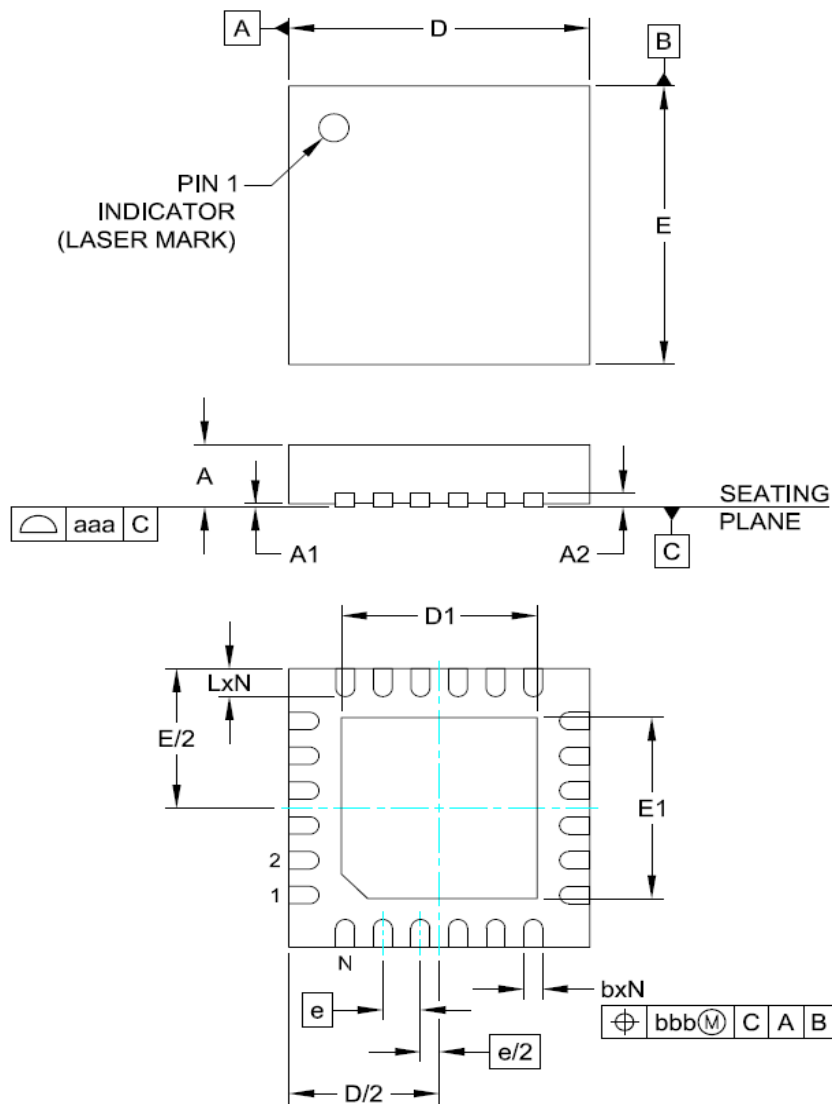
应用于 PCB 设计中的散热技术方案请参考 AN1200.37 “Recommendations for Best Performance” 可在 [www.semtech.com](http://www.semtech.com) 上下载。

在无法实现散热技术或者不支持使用 LowDataRateOptimize 功能的小型化设计中，使用 TCXO 将提供更稳定的参考时钟。

## 15. 封装信息

### 15.1 封装外形图

4x4mm QFN封装，引脚间距是0.5mm



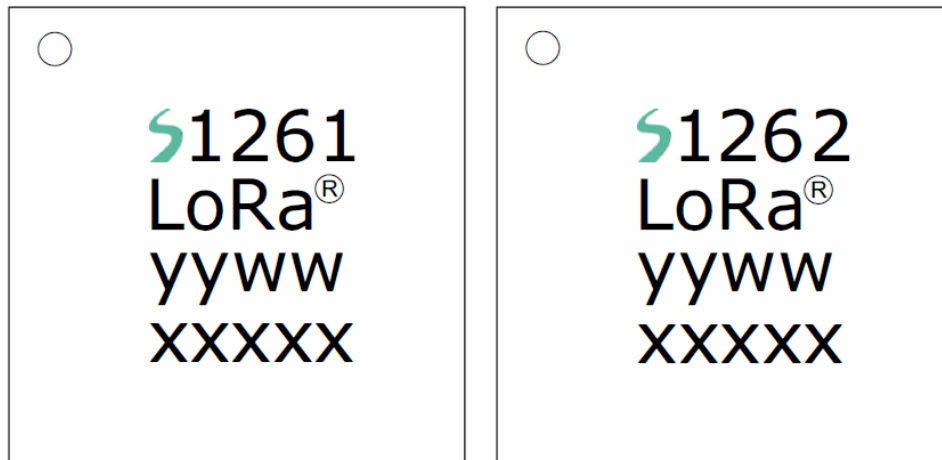
DIM	MILLIMETERS		
	MIN	NOM	MAX
A	0.80	0.90	1.00
A1	0.00	0.02	0.05
A2	(0.20)		
b	0.18	0.23	0.30
D	3.90	4.00	4.10
D1	2.55	2.60	2.65
E	3.90	4.00	4.10
E1	2.55	2.60	2.65
e	0.50 BSC		
L	0.35	0.40	0.45
N	24		
aaa	0.08		
bbb	0.10		

图 15-1: QFN 4x4 封装外型图

注意：

1. 尺寸以 mm 为单位（角度以度为单位）；
2. 芯片底部的热焊盘和引脚保持在同一平面；

### 15.2 包装标识



Marking for the 4 x 4 mm MLPQ 24 Lead package:

Snnnn = Part Number (Example: S1261)

yyww = Date Code (1752)

xxxxx = Semtech Lot Number (Example: E9010)

图 15-2 : SX1261/2 标识

### 15.3 焊盘布局

推荐的焊盘封装如下：

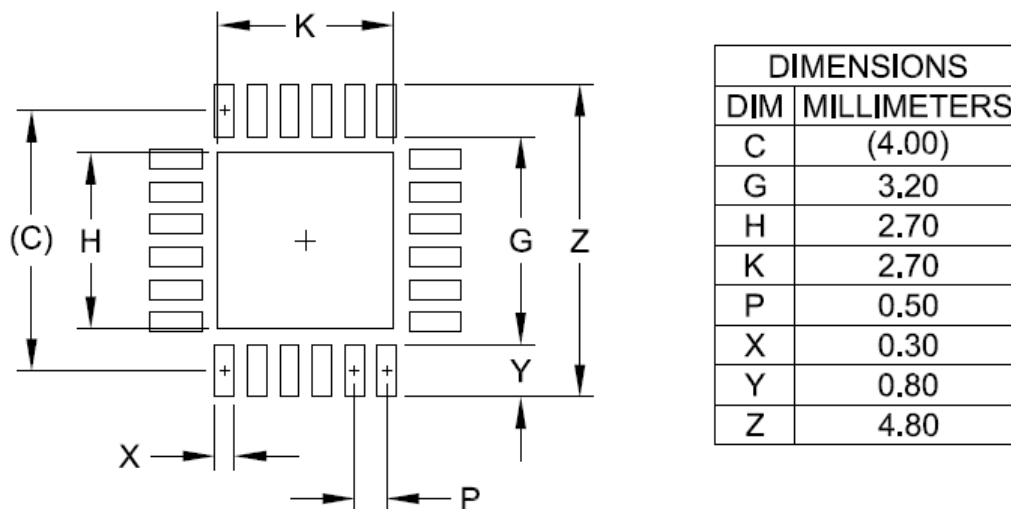


图 15-3：QFN 4x4MM LAND PATTERN

注意：

1. 尺寸以mm为单位（角度以度为单位）。
2. 该焊盘图仅作参考。为确保满足公司的生产要求，请咨询相关生产团队。
3. 芯片底部热焊盘下的散热过孔应连接到系统地平面上，否则可能会损害器件的热性能和/或其他功能性能。
4. 正方形封装——X方向和Y方向均适用以上尺寸。

#### 15.4 回流焊曲线

在Semtech网站上有回流焊接流程说明书，链接如下：

[http://www.semtech.com/quality/ir\\_reflow\\_profiles.html](http://www.semtech.com/quality/ir_reflow_profiles.html)

芯片使用QFN24 4x4 mm封装，也称为MLP封装。

**词汇表**
**首字母缩略词及其意义**

首字母缩略词	意义
ACR	Adjacent Channel Rejection 邻信道抑制
ADC	Analog-to-Digital Converter 模数转换器
API	Application Programming Interface 应用程序接口
$\beta$	Modulation Index 调制指数
BER	Bit Error Rate 误码率
BR	Bit Rate 比特率
BT	Bandwidth-Time bit period product 带宽时间乘积
BW	BandWidth 带宽
CAD	Channel Activity Detection 信道有效性检查
CPOL	Clock Polarity 时钟极性
CPHA	Clock Phase 时钟相位
CR	Coding Rate 编码率
CRC	Cyclical Redundancy Check 循环冗余检查
CW	Continuous Wave 连续波
DIO	Digital Input / Output 数字输入/输出
DSB	Double Side Band 双边带
ECO	Engineering Change Order 工程变更单
FDA	Frequency Deviation 频率偏差
FEC	Forward Error Correction 前向纠错
FIFO	First In First Out 先进先出
FSK	Frequency Shift Keying 频移键控
GFSK	Gaussian Frequency Shift Keying 高斯频移键控
GMSK	Gaussian Minimum Shift Keying 高斯最小频移键控
GDPW	Gross Die Per Wafer 晶片总数
IF	Intermediate Frequencies 中频频率
IRQ	Interrupt Request 中断请求
ISM	Industrial, Scientific and Medical (radio spectrum) 工业、科学和医学（无线电频谱）
LDO	Low-Dropout 低压差变换器
LDRO	Low Data Rate Optimization 低数据率优化

LFSR	Linear-Feedback Shift Register 线性反馈移位寄存器
LNA	Low-Noise Amplifier 低噪声放大器
LO	Local Oscillator 本机振荡器
LoRa®	Long Range Communication <i>the LoRa® Mark is a registered trademark of the Semtech Corporation</i>
LSB	Least Significant Bit 最低有效位
MISO	Master Input Slave Output 主输入从属输出
MOSI	Master Output Slave Input 主输出从输入
MSB	Most Significant Bit 最高有效位
MSK	Minimum-Shift Keying 最小移位键控
NOP	No Operation (0x00) 无操作
NRZ	Non-Return-to-Zero 非归零码
NSS	Slave Select active low 从选择低有效
OCP	Over Current Protection 过流保护
PA	Power Amplifier 功率放大器
PER	Packet Error Rate 误包率
PHY	Physical Layer 物理层
PID	Product Identification 产品标识
PLL	Phase-Locked Loop 锁相环
POR	Power On Reset 上电复位
RC13M	13 MHz Resistance-Capacitance Oscillator 13 MHz 阻容振荡器
RC64k	64 kHz Resistance-Capacitance Oscillator 64 KHz 阻容振荡器
RFO	Radio Frequency Output 射频输出
RFU	Reserved for Future Use 保留将来使用
RTC	Real-Time Clock 实时时钟
SCK	Serial Clock 串行时钟
SF	Spreading Factor 扩频因子
SN	Sequence Number 序列号
SNR	Signal to Noise Ratio 信噪比
SPI	Serial Peripheral Interface 串行外设接口
SSB	Single Side Bandwidth 单边带
STDBY	Standby 待机
TCXO	Temperature-Compensated Crystal Oscillator 温度补偿晶体振荡器

XOSC	Crystal Oscillator 晶体振荡器
------	--------------------------