

VG254M

产品描述

VG254M 是一款低成本，高集成度的 2.4GHZ 的无线收发 SOC，片上集成发射机，接收机，频率综合器，GFSK/FSK 调制解调器和微处理器。发射机支持功率可调，接收机采用数字扩展通信机制，在复杂环境和强干扰条件下，可以达到优良的收发性能。外围电路简单，只需少数外围被动器件。VG254M 传输 GFSK/FSK 信号，发射功率最大可以到 8dBm。接收机采用低中频结构，接收灵敏度可以达到 -92dBm@1Mbps / -86dBm@2Mbps。片上集成发射接收 FIFO 寄存器以缓存数据。数字基带通过 4 线 SPI 接口和内置 MCU 通信。为了提高电池使用寿命，芯片在各个环节都降低功耗，在保持寄存器值条件下，最低电流为 5uA。

VG254M 集成 8 位 OTP 微处理器，采用 CMOS 制程并同时提供客户低成本、高性能等显著优势。处理器核心建立在 RISC 精简指令集架构可以很容易地做编程和控制，共有 55 条指令。除了少数指令需要 2 个指令周期外，大多数指令都是 1 个指令周期即能完成，可以让用户轻松地以程序控制完成不同的应用，支持 C、汇编语言，编程简单易上手。

VG254M 采样 SOP-16 封装。

特性

集成射频前端、数字基带和微处理器

可编程发射功率，最大 8dBm

接收灵敏度 -92dBm@1Mbps

GFSK/FSK 调制

1/2Mbps 的空气数据速率

1-32 bytes 动态 FIFO

工作电压 1.9V-3.6V

2Kx14 bits OTP 程序空间

128 bytes SRAM

16 阶高精度 LVD

14 个可独立控制的双向 IO

5 路 10bit PWM

集成 1~20MHz 高精度震荡器

双时钟机制随时切换



应用领域

遥控门禁系统

胎压监测设备

遥控风扇、照明开关

无线传感数据传输

玩具遥控

目录

产品描述	1
特性	1
应用领域	1
1、脚位定义及说明	1
1.1 脚位示意图	1
1.2 脚位说明	2
2、绝对最大额定值	3
3、工作条件	3
4、电特性参数	4
5、功能描述	5
5.1 框架图	5
5.2 晶体	5
6、射频功能操作	6
6.1 操作模式	6
6.2 空中传输速率 (Air Data Rate)	8
7、协议引擎	10
7.1 特性	10
7.2 协议引擎概述	10
7.3 协议引擎封包格式	10
7.4 自动数据包事务处理	15
7.5 协议引擎流程图	16
7.6 多从机 (Multi-Slave)	19
7.7 协议引擎时序 (Protocol engine timing)	21
7.8 协议引擎事件图 (Protocol engine transaction diagram)	22
8、数据接口	1
8.1 特性	1
8.2 功能描述	1
8.3 SPI 操作机制	1
8.4 FIFO	3
8.5 中断	4
9、应用电路图	1
10、封装外形	1

1、脚位定义及说明

1.1 脚位示意图

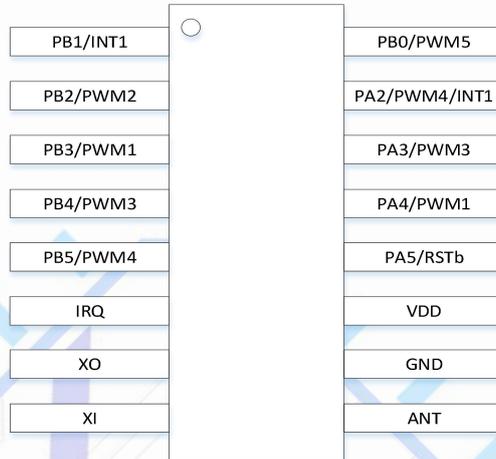


图 1 VG254M 管脚示意图

芯片内部 SPI 配置说明:

MISO-----PA0

MOSI-----PA1

SCK -----PA7

CSN -----PA6



1.2 脚位说明

表 1 VG254M 管脚描述

管脚编号	管脚名称	I/O	管脚功能描述
1	PB1/INT1	I/O	可编程输入输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 外部中断, 且该管脚具有红外遥控驱动功能
2	PB2/PWM2	I/O	可编程输入输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 PWM2 输出管脚。
3	PB3/PWM1	I/O	可编程输入输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 PWM1 输出管脚。
4	PB4/PWM3	I/O	可编程输入输出脚, 可编程上拉电阻, 电平变化唤醒 PWM3 输出管脚。
5	PB5/PWM4	I/O	可编程输入输出脚, 可编程上拉电阻, 电平变化唤醒 PWM4 输出管脚。
6	IRQ	O	中断信号输出
7	XO	A	晶体振荡器输入管脚, 外接晶体
8	XI	A	晶体振荡器输入管脚, 外接晶体
9	ANT	A	射频信号输入管脚, 外接匹配网络
10	GND	P	接地
11	VDD	P	电源输入
12	GND	P	接地
13	PA5/RSTb	I/O	可编程输入脚, 开漏输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 复位管脚。
14	PA3/PWM3	I/O	可编程输入输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 PWM3 输出管脚。
15	PA2/PWM4 /INT1	I/O	可编程输入输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 PWM4 输出管脚, 外部中断
16	PB0/PWM5	I/O	可编程输入输出脚, 可编程上拉电阻, 可编程下拉电阻, 电平变化唤醒 PWM5 输出管脚。

2、绝对最大额定值

表 2 绝对最大额定值

参数	符号	条件	最小	最大	单位
电源电压	V_{DD}		-0.3	3.6	V
接口电压	V_{IN}		-0.3	$V_{DD} + 0.3$	V
结温	T_J		-40	125	°C
储藏温度	T_{STG}		-40	125	°C
焊接温度	T_{SDR}	持续时间不超过 30 秒		255	°C
ESD 等级		人体模型(HBM)	-3	3	kV
栓锁电流		@ 85 °C	-100	100	mA

3、工作条件

表 3 推荐工作条件

参数	符号	条件	最小	最大	单位	参数
运行电源电压	V_{DD}	-40°C到+85°C	1.9		3.6	V
运行温度	T_{OP}		-30		70	°C
电源电压斜率	V_{SL}		1			mV/us

4、电特性参数

表 4 电特性参数

参数	符号	条件	最小	典型	最大	单位
频率范围	F_{RF}		2400		2525	MHz
频道间隔	δF			1		MHz
数据率	DR			1/2		Mbps
灵敏度	S	DR = 1 Mbps,		-92		dBm
	S	DR = 2 Mbps,		-86		dBm
工作电流	I_{RX}			17		mA
	I_{TX}	0dBm		15		mA
关断电流	I_{pd}	全部关闭		2		uA
发射功率	P_{max}	最大功率		8		dBm
	P_{min}	最小功率		-34		dBm
Spi 速率	DR_{spi}				8	Mbps
饱和输入电平	P_{LVL}			-10		dBm
晶振频率	F_{XI}			16		MHz
晶振误差	δF_{XI}		-20		20	ppm
负载电容	C_{Load}			12		PF

5、功能描述

5.1 框架图

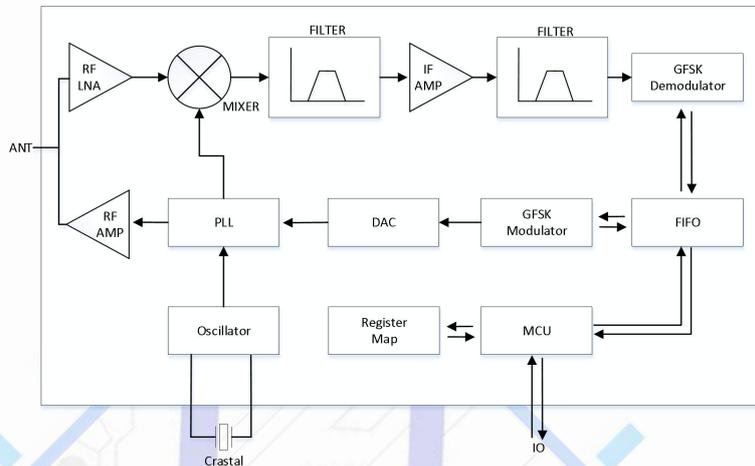


图 2 VG254M 内部电路框架图

VG254M 是一款数模混合设计的一体化接收机。该芯片集成 GFSK 接收机和发射机，并集成高性能 OTP MCU。仅需要外接少量器件即可实现稳定的数据传输功能。

5.2 晶体

为保证参考时钟的稳定性和相位噪声，VG254M 采用两个管脚驱动晶体振荡器。晶体振荡所需的负载电容集成于芯片内，无需外置电容。推荐使用精度在为 ± 20 ppm，等效电阻小于 60Ω ，负载电容为 12 pF ， 16MHz 的晶体。需要注意的是，由于不同封装规格的晶体存在着寄生电容差异，请用户选用晶体时注意评估，避免由于晶体震荡频率偏离目标值过大而引起接收机性能降低。

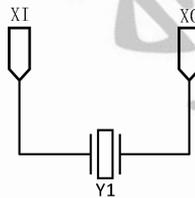


图 3 VG254M 晶振驱动电路图

6、射频功能操作

本章描述 VG254M 射频收发器的操作模式以及用于控制无线电的参数。

6.1 操作模式

可将 VG254M 配置为掉电 (Power down)、睡眠 (Sleep)、待机 (Standby)、接收 (RX) 或发射 (TX) 模式。VG254M 内置一个状态机，用于控制芯片各操作模式之间的转换。该状态机接收来自用户定义的寄存器值和内部信号的输入。

6.1.1 状态图

图 4 展示了操作模式及其运作方式。状态图中突出显示了三种不同类型的状态：

推荐操作模式 (Recommended operating mode): 在正常操作期间使用的推荐状态。

可能操作模式 (Possible operating mode): 一种可能的操作状态，但在正常操作期间通常不使用。

暂态 (Transition state): 在振荡器启动和锁相环 (PLL) 锁定过程中使用的、有时间限制的状态。

当 VDD 达到 1.9V 或更高时，VG254M 进入上电复位状态 (Power on reset state)。在此状态下，芯片将保持复位状态，直到进入掉电模式 (Power Down mode)。

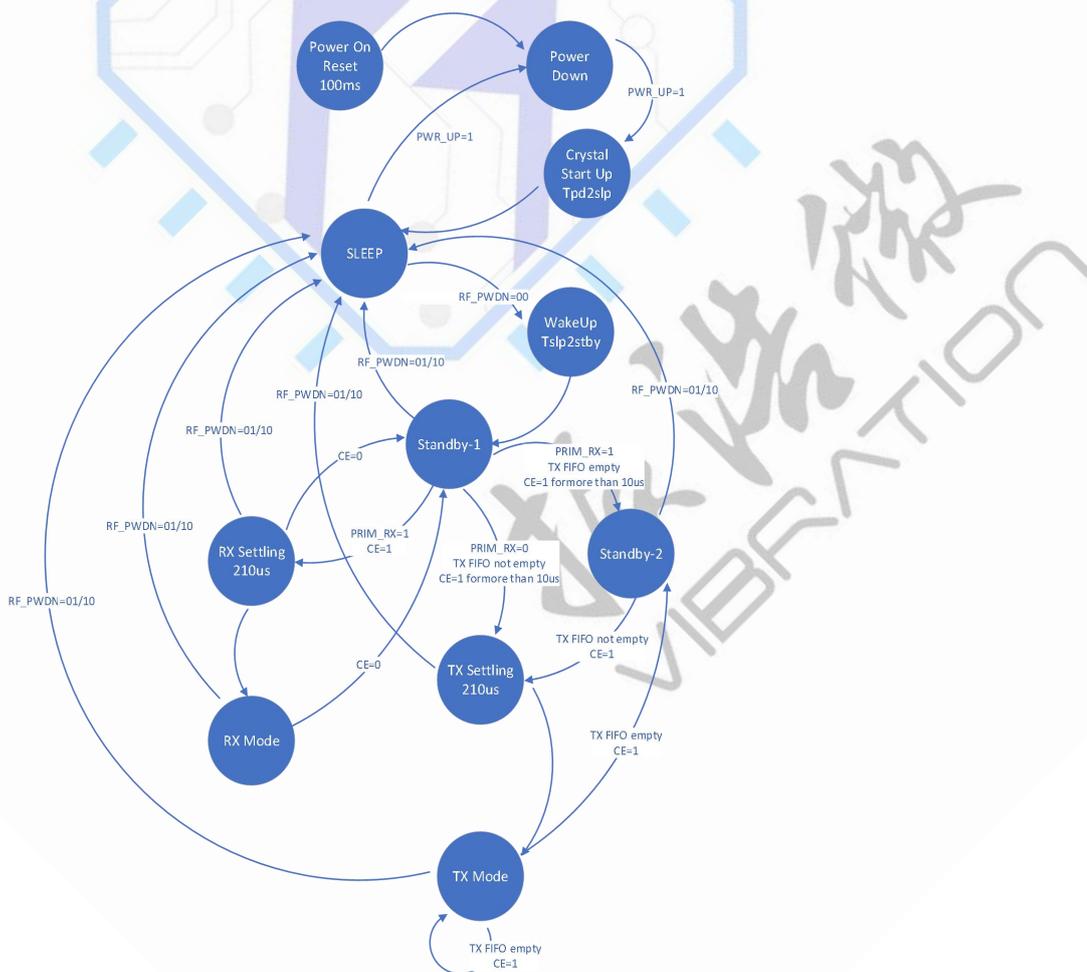


图 4 VG254M 射频工作模式状态图

6.1.2 掉电模式 (Power-Down Mode)

在此模式下，VG254M 被禁用以最小化平均电流消耗。所有寄存器值均被保留，且 SPI 接口保持活跃，允许修改配置及读/写数据寄存器。启动时间参见表 6.2。

进入方式：当芯片处于睡眠模式时，将 CONFIG 寄存器的 PWR_UP 位设为 0。

6.1.3 睡眠模式 (Sleep Mode)

此模式下 VG254M 同样最小化平均电流消耗。所有寄存器值保持有效，SPI 接口和晶振电路保持运行。

6.1.4 待机模式 (Standby Modes)

待机模式分为待机模式 1 和待机模式 2。在两种待机模式下寄存器值都会保留，且 SPI 可以工作。

将 CONFIG 寄存器的 PWR_UP 位置 1 进入睡眠模式后，再清除 RF_PWDN 的两位，即可转入待机模式 1。此模式在保持快速启动能力的同时最小化电流消耗，仅部分晶振电路工作。在待机模式 1 状态下，把 CE 置 1 可以立即退出待机模式 1；在 RX mode 或者 TX mode 下把 CE 清零即可进入待机模式 1。

相较于待机模式 1，待机模式 2 激活时钟缓冲器，电流消耗更高。当 PTX 设备的 TX FIFO 为空且 CE 为 1 时，芯片进入待机模式 2。若有新数据包写入 TX FIFO，PLL 立即启动并在标准锁定延时 (210 μ s) 后传输数据。

注意：从待机模式 1 切换到待机模式 2 需保持 CE=1>20 μ s

6.1.5 接收模式 (RX Mode)

在此模式下，VG254M 工作于接收模式。把寄存器控制位 PWR_UP、PRIM_RX 和 CE 置 1 即可进入 RX Mode。

在 RX 模式下，接收器解调来自 RF 信道的信号，不断将解调后的数据提供给基带协议引擎。基带协议引擎持续搜索有效的数据包。如果找到有效的数据包（通过匹配地址和有效的 CRC），则该数据包的有效载荷将被存入 RX FIFO 中的空闲位置。如果 RX FIFO 已满，则收到的数据包将被丢弃。

芯片保持在 RX 模式，直到 MCU 将其配置为待机模式 1 或掉电模式。但是，如果启用了基带协议引擎中的自动协议功能，芯片可以进入其他模式以执行协议。

6.1.6 发射模式 (TX Mode)

TX 模式是用于传输数据包的活动模式。要进入此模式，芯片必须将 PWR_UP 置 1，PRIM_RX 清零，TX FIFO 中存在有效载荷，并且在 CE=1 超过 20 μ s。

VG254M 保持在 TX 模式下，直到完成一个数据包的传输。如果 CE = 0，芯片将返回待机模式 1。如果 CE = 1，TX FIFO 的状态将决定下一步操作：如果 TX FIFO 不为空，VG254M 会保持在 TX 模式并传输下一个数据包；如果 TX FIFO 为空，芯片则进入待机模式 2。

6.1.7 操作模式配置 (Operational modes configuration)

表 5 描述了如何配置操作模式：

表 5 工作模式设置表

Mode	PWR_UP	RF_PWDN	PRIM_RX	CE Command	FIFO state
RX mode	1	00	1	1	-
TX mode	1	00	0	1	Data TX FIFO. Will empty in TX FIFO
TX mode	1	00	0	Minimum 20us high pulse	Data TX FIFO. Will empty in TX FIFO
Standby-I mode	1	00	-	0	No ongoing packet transmission
Standby-II mode	1	00	0	1	TX FIFO empty
Sleep Mode	1	01/10	-	-	-
Power Down	0	01	-	-	-

6.1.8 时序信息 (Timing Information)

本节中的时序信息涉及模式间切换的时序以及 CE 的时序要求。如表 6 所述，从 TX 模式切换到 RX 模式（或反之亦然）的时序与从待机模式切换到 TX 模式或 RX 模式的时序相同。

表 6 VG254M 芯片操作时序

Name	Action	Max.	Min.	Comments
T _{pd2stb}	Power Down→Standby mode		2ms	Depended on crystal
T _{stby2a}	Standby mode→TX/RX mode		130us	
T _{hce}	Minimum CE high		20us	
T _{pecc2esn}	Delay from CE positive edge to CSN low		4us	

为使 VG254M 从掉电模式进入 TX 或 RX 模式，必须先经过睡眠模式和待机模式。在 VG254M 退出掉电模式后，需延迟 T_{pd2slp} 和 T_{slp2stby}（参见表 6）方可置 1 CE 引脚。

注意：若 VDD 电源关闭，寄存器值将丢失，进入 TX 或 RX 模式前必须重新配置芯片。

6.2 空中传输速率 (Air Data Rate)

空中传输速率指芯片收发数据时的调制信号速率，可选 1Mbps 或 2Mbps。较低速率可提供更高的接收灵敏度，较高速率可降低平均电流消耗，并减少空中冲突概率通过配置 RF_SETUP 寄存器中的 RF_DR 位进行设置。

通信要求：发射端与接收端的速率配置必须一致。

6.3 射频信道频率 (RF Channel Frequency)

射频信道频率决定芯片所用信道的中心频率。信道占用带宽在 1Mbps 时小于 1MHz，在 2Mbps 时小于 2MHz。芯片工作频率范围为 2.400GHz 至 2.525GHz，射频信道频率设置的编程分辨率为 1MHz。在 2Mbps 模式下，信道占用带宽需大于射频频率设置的分辨率。为确保 2Mbps 模式下信道无重叠，信道间隔必须 \geq 2MHz。在 1Mbps 模式下，信道占用带宽等于或小于射频频率分辨率。

射频信道频率由 RF_CH 寄存器按以下公式设置：

$$F_o = 2400 + RF_CH \text{ [MHz]}$$

发射端与接收端必须配置相同的射频信道频率方可通信。

6.4 功率放大器控制 (PA Control)

功率放大器 (PA) 控制用于设置芯片功率放大器的输出功率。在 TX 模式下，PA 控制有四级可编程功率（见表 7）。

该功能通过 RF_SETUP 寄存器中的 PA_PWR 位配置。

表 7 PA 功率设置和发射电流

SPI RF-SETUP (PA_PWR[3:0])	RF output power	DC current consumption
1111	8dBm	37mA
0110	4dBm	21.5mA
0011	0dBm	15mA
0010	-4dBm	12.5mA

6.5 接收、发射控制

RX/TX 控制通过 CONFIG 寄存器中的 PRIM_RX 位进行设置，该位决定 VG254M 芯片处于发射 (TX) 或接收 (RX) 模式。

7、协议引擎

协议引擎是基于封包的数据链路层，具备自动封包组装与定时、自动应答及封包重传功能。该引擎可实现超低功耗与高性能通信，其特性显著提升了双向/单向系统的能效，且无需增加主控制器复杂度。

7.1 特性

协议引擎主要特性：

- 1 至 32 字节动态负载长度
- 自动封包处理
- 自动封包事务管理
 - 含负载的自动应答
 - 自动重传
- 支持 1:3 星型网络的 3 条数据通道

7.2 协议引擎概述

协议引擎使用自定义协议实现自动封包处理与定时：

发送时：组装封包并为传输数据提供时钟信号

接收时：持续在解调信号中搜索有效地址。发现有效地址后处理剩余封包并通过 CRC 校验。若校验通过，负载移入 RX FIFO 空闲槽位。所有高速比特处理与定时均由协议引擎控制。

协议引擎的自动封包事务管理简化了可靠双向数据链路的实现：

封包事务是两台收发器间的封包交换，一方作主接收器 (PRX)，另一方作主发射器 (PTX)。

事务由 PTX 发送封包启动，当 PTX 收到 PRX 的应答封包 (ACK) 时完成。PRX 可在 ACK 封包中附加用户数据实现双向通信。

自动事务处理流程：

- 1、PTX 向 PRX 发送数据包启动事务，协议引擎自动将 PTX 切换至接收模式等待 ACK 包。
- 2、PRX 收到包后，协议引擎自动组装并发送 ACK 包至 PTX，随后返回接收模式。
- 3、若 PTX 未立即收到 ACK 包，协议引擎在可编程延迟后自动重发原始数据包，并维持接收模式等待 ACK 包。

可配置最大重传次数/重传间隔等参数，所有处理均无需 MCU 介入。

7.3 协议引擎封包格式

本节描述协议引擎数据包的格式：该数据包包含前导码(Preamble)、地址(Address)、封包控制域(Packet Control Field)、负载(Payload)及校验码(CRC)。表 8 展示了该数据包格式 (MSB 居左)。

表 8 数据包格式

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

7.3.1 前导码(Preamble)

前导码是用于使接收器解调器与输入比特流同步的比特序列，长度为 1 字节（01010101 或 10101010）。若地址首比特为 1，则前导码自动设为 10101010；若首比特为 0，则设为 01010101。此举可确保前导码具备足够的跳变以稳定接收器。

7.3.2 地址(Address)

此为接收器地址字段，用于确保数据包被正确接收器检测接收，防止多 OM6220 系统间意外串扰。可通过 AW 寄存器配置地址字段宽度为 5 字节或 4 字节。

7.3.3 保护间隔(Guard)

2 字节保护间隔提供更优的同步特性。

7.3.4 封包控制域(Packet Control Field)

图 7.2 表 9 展示 9 比特封包控制域格式（MSB 居左）：

表 8 数据包格式

Payload length	PID	NO_ACK
6bit	2bit	1bit

该域包含 6 比特负载长度字段、2 比特 PID（数据包标识）字段及 1 比特 NO_ACK 标志。
负载长度：此 6bit 数据声明负载的字节长度（0-32 字节）。

编码方案：

000000 = 0 字节（仅用于空 ACK 包，需通过 R_RX_PAYLOAD 指令读取且无后续数据）

100000 = 32 字节

100001 = 无关值

仅当启用动态负载长度（DPL）功能时生效

PID（数据包标识）：此 2bit 数据用于检测接收包为新包或重发包，防止 PRX 操作向 MCU 重复提交相同负载。PTX 端每通过 SPI 接收新包时递增 PID 值。PRX 操作通过比对 PID 与 CRC 值判定包类型：当连续数据包 PID 相同时比对两包 CRC；若 CRC 亦相同，则判定为重发包并丢弃。

NO_ACK（不应答标志）：仅当启用自动应答功能时有效，置高表示不要求自动应答。PTX 可通过 W_TX_PAYLOAD_NOACK 指令设置此标志，但须先在 FEATURE 寄存器中将 EN_DYN_ACK 位置 1。

生效行为：

PTX 发包后直接进入待机模式 1

PRX 收包时不发送 ACK

7.3.5 负载(Payload)

负载是用户自定义的数据包内容，其宽度可为 0 至 32 字节，在上传至设备后通过无线传输。协议引擎提供静态负载长度与动态负载长度两种处理模式。

默认采用静态负载长度模式。在静态负载长度下，发射器与接收器之间所有数据包的长度相同。接收端的负载长度由 RX_PW_Px 寄存器设定。发射端的负载长度由移入 TX_FIFO 的字节数决定，且必须严格匹配接收端 RX_PW_Px 寄存器的设定值。

动态负载长度（DPL）是静态负载长度的替代方案，允许发射器向接收器发送可变长度的负载。这意味着在需要传输不同长度负载的系统中，无需将所有数据包扩展至最大负载长度。启用 DPL 功能后，OM6220 将自动解析接收数据包的负载长度，而不再依赖 RX_PW_Px 寄存器。MCU 可通过 R_RX_PL_WID 指令读取接收负载的长度。

注意：使用 R_RX_PL_WID 指令时，必须检查报告的长度是否 ≤ 32 字节。若长度超过 32 字节，表明数据包存在错误，必须使用 Flush_RX 命令丢弃该包。

启用 DPL 需执行以下操作：在 FEATURE 寄存器中置位 EN_DPL 比特；RX 模式下需配置 DYNPD 寄存器；当 PTX 向启用 DPL 的 PRX 发送数据时，必须在 DYNPD 寄存器中置位 DPL_P0 比特。

7.3.6 校验码(CRC)

CRC 是数据包中的检错机制（1 或 2 字节），校验范围覆盖地址、封包控制域及负载。多项式及初始值：

1 字节 CRC: $X^8 + X^2 + X + 1$ （初始值 0xFF）

2 字节 CRC: $X^{16} + X^{12} + X^5 + 1$ （初始值 0xFFFF）

CRC 字节数通过 CONFIG 寄存器 CRCO 位设定。若校验失败，协议引擎拒收数据包。

7.3.7 自动数据包组装

在发送前自动将前导码、地址、封包控制域、负载及 CRC 组装为完整数据包。



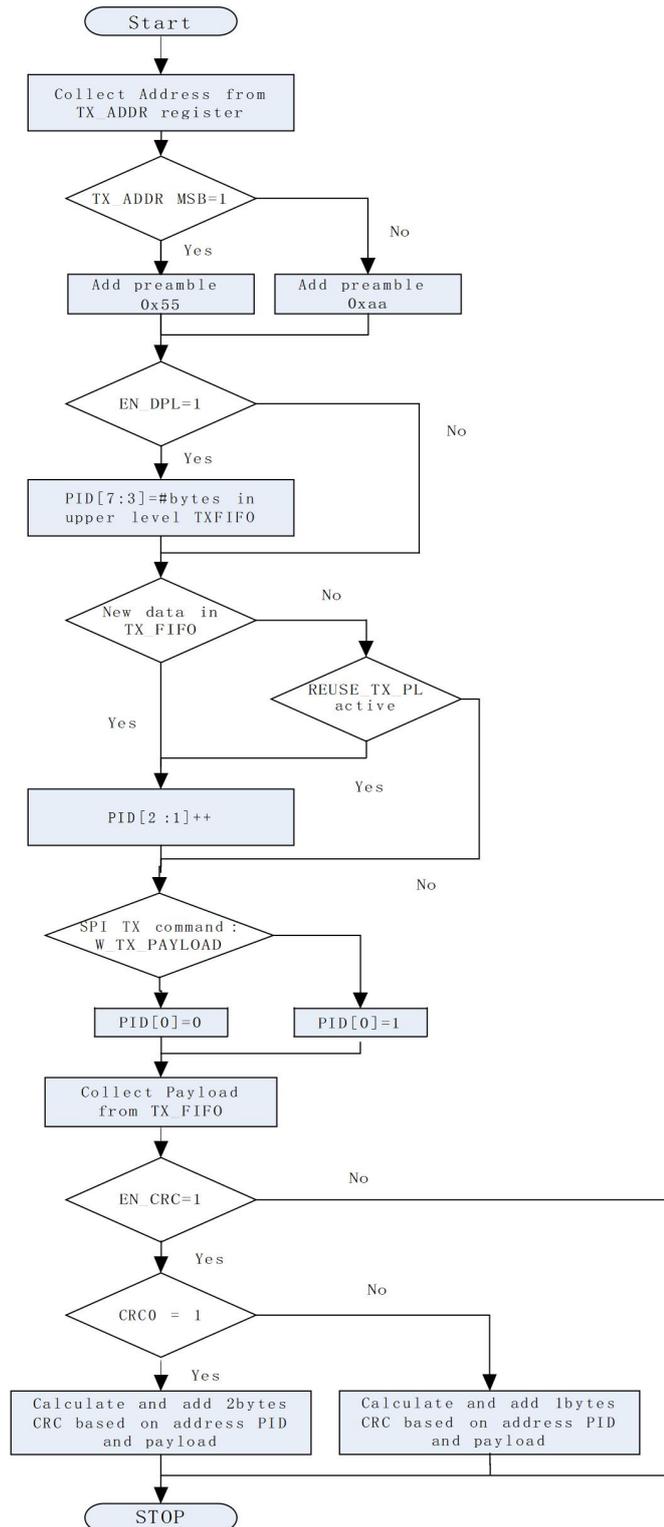


图 5 自动封包流程

7.3.8 自动数据包解包

数据包通过校验后，协议引擎将解组该数据包并将负载载入 RX FIFO，同时触发 RX_DR 中断请求。

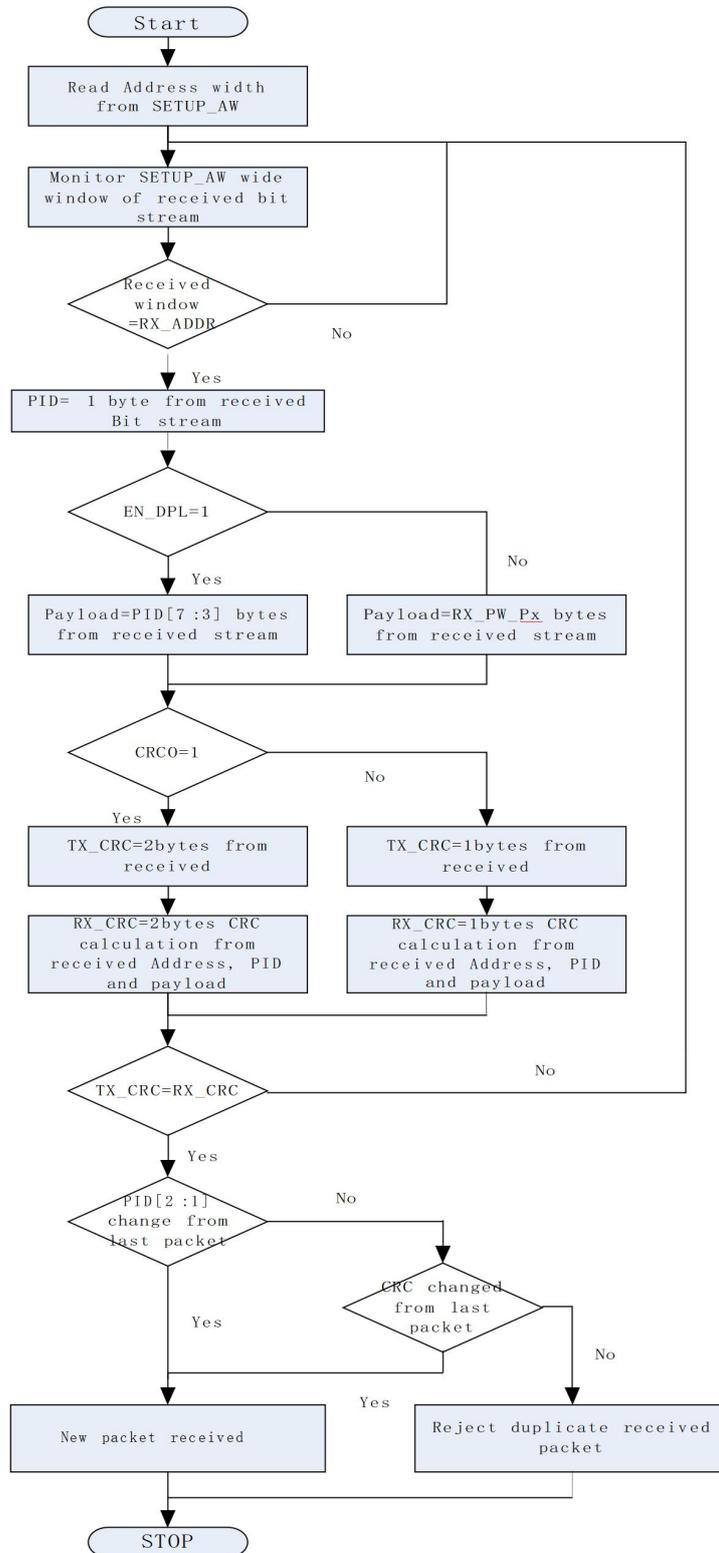


图 6 自动解包流程

7.4 自动数据包事务处理

协议引擎提供两种自动数据包事务处理功能：自动应答（Auto Acknowledgement）与自动重传（Auto Re-transmit）。

7.4.1 自动应答

自动应答功能在接收并校验通过数据包后，自动向 PTX 发送 ACK 包。此功能可减轻系统 MCU 负载并降低平均电流功耗。通过在 EN_AA 寄存器置位启用该功能。

注意：若接收包 NO_ACK 标志置位，则不执行自动应答。

一个 ACK 包可以包含一个从 PRX 到 PTX 的可选负载。为了使用此功能，必须启用动态负载长度 (DPL) 功能。PRX 端的 MCU 必须通过使用 W_ACK_PAYLOAD 指令将负载移入 TX FIFO 来上传负载。该负载保持在 TX FIFO (PRX) 中挂起，直到从 PTX 接收到新的数据包。射频收发器可以同时保持三个 ACK 包负载挂起。

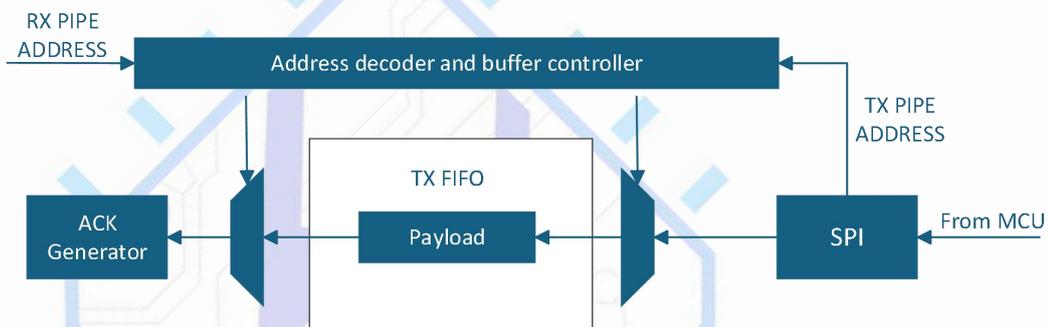


图 7 TX FIFO (PRX) 中挂起的负载

图 7 展示了 TX FIFO (PRX) 在处理挂起的 ACK 包负载时是如何操作的。来自 MCU 的负载通过 W_ACK_PAYLOAD 指令被移入。地址解码器和缓冲区控制器确保负载被存储在 TX FIFO (PRX) 中的一个空闲槽位中。当一个数据包被接收时，地址解码器和缓冲区控制器被通知了 PTX 地址。这确保了正确的负载被提供给 ACK 生成器。

如果 TX FIFO (PRX) 包含多于一个到一个 PTX 的负载，负载的处理采用先入先出原则。如果所有挂起的负载都是发往一个链路丢失的 PTX，则 TX FIFO (PRX) 将被阻塞。在这种情况下，MCU 可以通过使用 FLUSH_TX 命令清空 TX FIFO (PRX)。

为了启用带负载的自动应答，必须在 FEATURE 寄存器 EN_ACK_PAY 比特必须被置 1。

7.4.2 自动重传 (ART)

自动重传：如果没有收到 ACK 包，则重新传输一个数据包。它用于 PTX 侧的自动应答系统中。当一个数据包未被确认时，您可以通过在 SETUP_RETR 寄存器中设置 ARC 比特来设置其允许重传的次数。每次传输一个数据包之后，PTX 进入 RX 模式并等待一段时间的 ACK 包。PTX 处于 RX 模式的时间量基于以下条件：

自动重传延迟 (ARD) 已用尽。

256μs 内没有地址匹配。

地址匹配之后接收完数据包（CRC 正确或不正确）之后的 256μs 内。

当 ACK 包被接收时，射频收发器触发 TX_DS IRQ。

如果 TX FIFO 中没有更多未传输的数据并且 CE 为低，则射频收发器进入待机模式 1。如果 ACK 包未被接收，在由 ARD 定义的延时之后，射频收发器返回到 TX 模式并重传数据。这种情况一直持续，直到接收到确认，或者达到了最大重传次数。

每次丢失一个数据包时，两个丢包计数器会递增：位于 OBSERVE_TX 寄存器中的 ARC_CNT 和 PLOS_CNT。ARC_CNT 计数当前事务的重传次数。您通过发起一个新的事务来重置

ARC_CNT。 PLOS_CNT 计数自上次信道更改以来的总重传次数。 您通过写入 RF_CH 寄存器来重置 PLOS_CNT。可以使用 OBSERVE_TX 寄存器中的信息来对信道质量做整体评估。

ARD 定义了从已传输数据包结束到在 PTX 上开始一次重传之间的时间。 ARD 在 SETUP_RETR 寄存器中设置，以 256 μ s 为步长。 如果 PTX 没有接收到 ACK 包，将进行重传。

在使用带负载的 ACK 包时，对 ARD 的长度有限制。 ARD 时间绝不得短于启动时间和 ACK 包空中传输时间之和。

对于 2 Mbps 数据速率和 5 字节地址；对于 ARD=256 μ s（复位值），15 字节是最大 ACK 包负载长度。

对于 1 Mbps 数据速率和 5 字节地址；对于 ARD=256 μ s（复位值），5 字节是最大 ACK 包负载长度。

ARD=512 μ s 在 1 或 2 Mbps 模式下对于任何 ACK 负载长度都足够长。

作为自动重传的替代方案，可以将射频收发器手动设置为重传一个数据包若干次。 这是通过 REUSE_TX_PL 命令实现的。 当使用此命令时，MCU 必须通过在 CE 引脚上发送一个脉冲来发起该数据包的每次传输。

7.5 协议引擎流程图

本节包含勾勒协议引擎中 PTX 和 PRX 操作的流程图。

7.5.1 PTX 操作

图 8 描述了 VG254M 在待机模式 1 和 TX 模式之间的切换流程。



执行自动重传功能时，重传次数可能达到 ARC 定义的最大值。若发生此情况，RF 收发器触发 MAX_RT IRQ 并返回待机模式 1。

若 RCON 寄存器中的 CE 位为高且 TX FIFO 为空，RF 收发器进入待机模式 2。

7.5.2 PRX 操作

图 9 描述了 VG254M 在待机模式 1 和 RX 模式之间的切换流程。

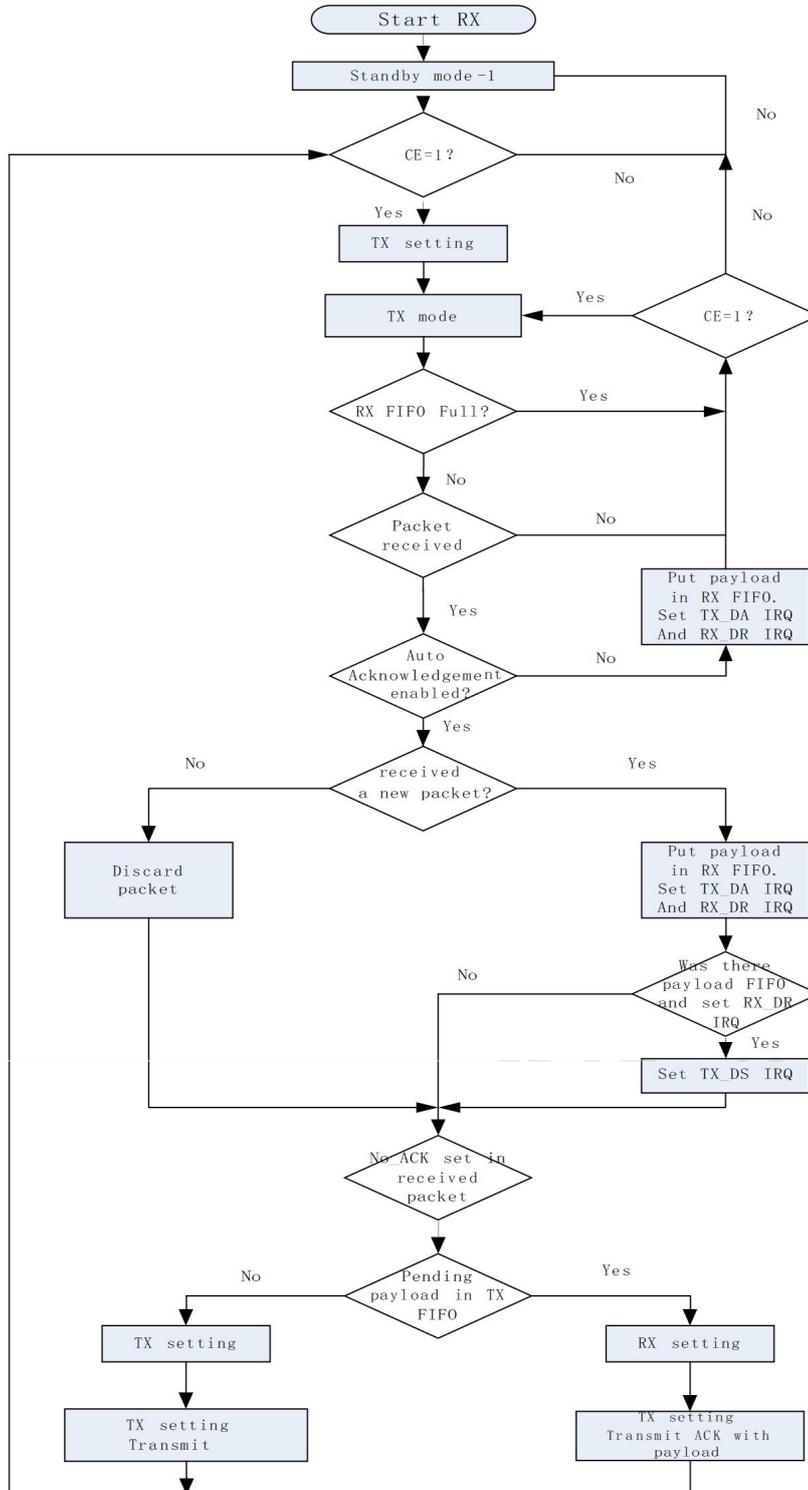


图 9 接收操作流程图

将 RFCON 寄存器中的 CE 位置高以激活 PRX 模式。RF 收发器进入 RX 模式并开始搜索数据包。若接收到数据包且启用自动确认 (Auto Acknowledgement)，RF 收发器将判断该数据包是新的还是已接收数据包的副本。若数据包为新的有效载荷，则将其存入 RX FIFO 并触发 RX_DR 中断请求 (IRQ)。

若当前接收的数据包通过含有效载荷的 ACK 数据包确认了来自发射器 (PTX) 的上一数据包，则 TX_DS IRQ 表明 PTX 接收到了含有效载荷的 ACK 数据包。若接收的数据包中未设置 No_ACK 标志，PRX 将进入 TX 模式；若 TX FIFO 中存在待处理的有效载荷，则将其附加至 ACK 数据包。ACK 数据包发送完成后，RF 收发器返回 RX 模式。

注：若 ACK 数据包丢失，可能接收到先前已收数据包的副本。此时 PRX 将丢弃该数据包，并在返回 RX 模式前发送一个 ACK 数据包。

7.6 多从机 (Multi-Slave)

多从机 (Multi-Slave) 是一种用于接收 (RX) 模式的功能，包含三条具有唯一地址 (unique addresses) 的并行数据通道 (data pipes)。数据通道 (data pipe) 是物理射频信道 (physical RF channel) 中的一种逻辑信道 (logical channel)。每个数据通道在 RF 收发器中拥有独立的物理地址 (数据通道地址) 解码功能。

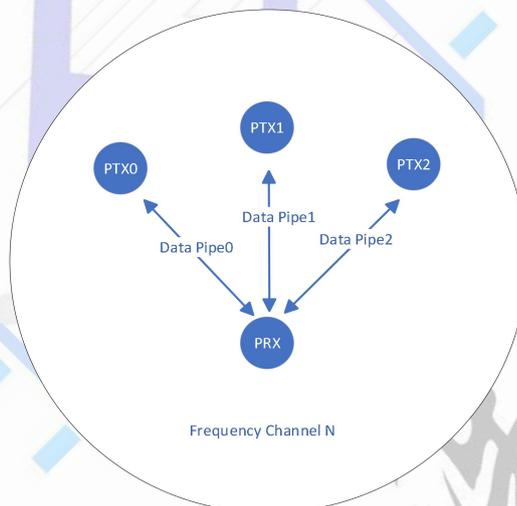


图 10 多从机工作示意图

配置为 PRX (主接收器) 的 RF 收发器可在单一频率信道中接收指向三条不同数据通道的数据包 (见图 10)。每条数据通道拥有唯一地址，且可独立配置工作模式。

最多三台配置为 PTX (主发射器) 的 RF 收发器可与一台 PRX 通信。所有数据通道地址会同时被搜索，但每次仅一条通道能接收数据包。所有数据通道均可执行协议引擎功能。

以下设置为所有数据通道的通用配置：

- CRC 启用/禁用 (启用协议引擎时 CRC 强制启用)
- CRC 编码方案
- RX 地址宽度
- 频率信道
- 空口数据速率
- LNA 增益 (LNAGAIN)

通过 EN_RXADDR 寄存器的位使能数据通道，默认仅通道 0 和 1 启用。各通道地址在 RX_ADDR_PX 寄存器中配置。

注：必须确保所有数据通道地址互不相同

每条通道最多可配置 5 字节地址。通道 0-2 共享高 4 字节地址，最低有效字节（LSByte）必须在所有六条通道中保持唯一。图 11 为通道 0-2 的寻址示例。仅通道 0 支持 5 字节可配置地址，其他通道仅支持 1 字节可配置地址。

	Byte4	Byte3	Byte2	Byte1	Byte0
Data Pipe0 (RX_ADDR_P0)	0XE7	0XD3	0XF0	0X35	0XC0
Data Pipe1 (RX_ADDR_P1)	0XE7	0XD3	0XF0	0X35	0XC1
Data Pipe2 (RX_ADDR_P2)	0XE7	0XD3	0XF0	0X35	0XC2

图 11 通道 0-2 的寻址示例

PRX（主接收器）使用多从机和协议引擎，从一个以上 PTX 接收数据包。为确保来自 PRX 的 ACK 数据包被传输到正确 PTX，PRX 采用其接收该数据包的数据通道地址，并在传输 ACK 数据包时将其用作 TX 地址。图 12 是 PRX 和 PTX 的地址配置示例。在 PRX 上 RX_ADDR_Px（定义为管道地址）必须是唯一的。在 PTX 上 TX_ADDR 必须与 RX_ADDR_P0 相同，且必须与所指定管道的管道地址相同。

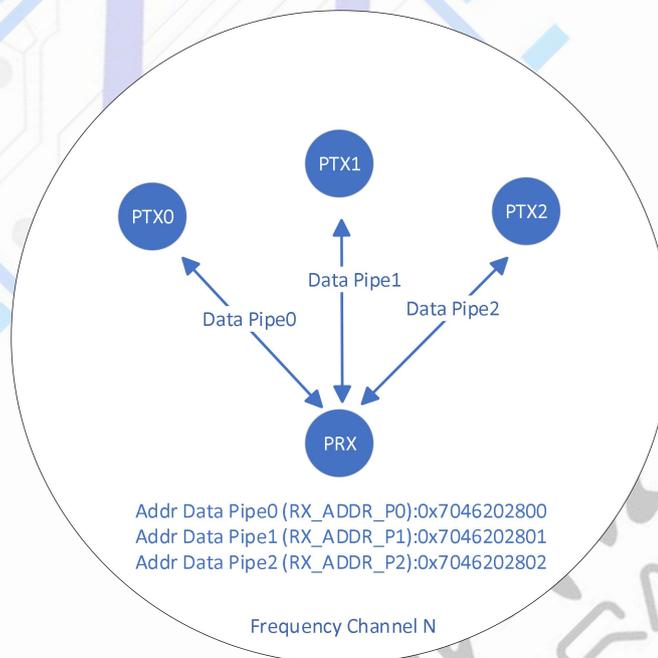
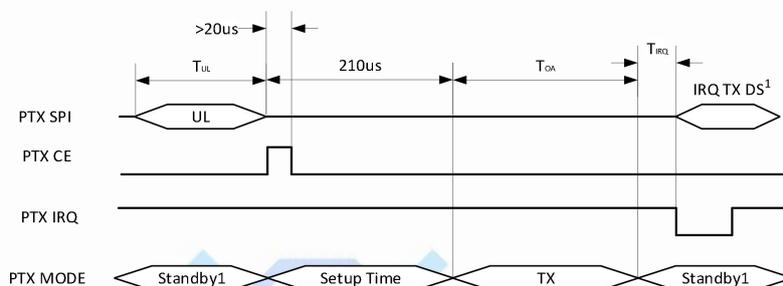


图 12 多从机模式下 PRX 和 PTX 的地址配置示意图

PRX 仅在一个数据管道接收到完整数据包后其他数据管道才能开始接收数据。当多个 PTX 向 PRX 传输时 ARD 可用于错开自动重发使得它们仅彼此阻塞一次。

7.7 协议引擎时序 (Protocol engine timing)

本节描述协议引擎的时序以及所有模式如何被启动和操作。协议引擎时序通过数据与控制接口控制。RF 收发器可被设置为静态模式或内部状态机控制事件的自主模式。每个自主模式/序列以 RFIRQ 中断结束。所有中断在时序图中被指示为 IRQ 事件。



注释 1:NO_ACK 使能

$T_{irq}=3\mu s$ @ 1Mbps, @2Mbps

图 13 使能 NO_ACK 条件下发射一包数据

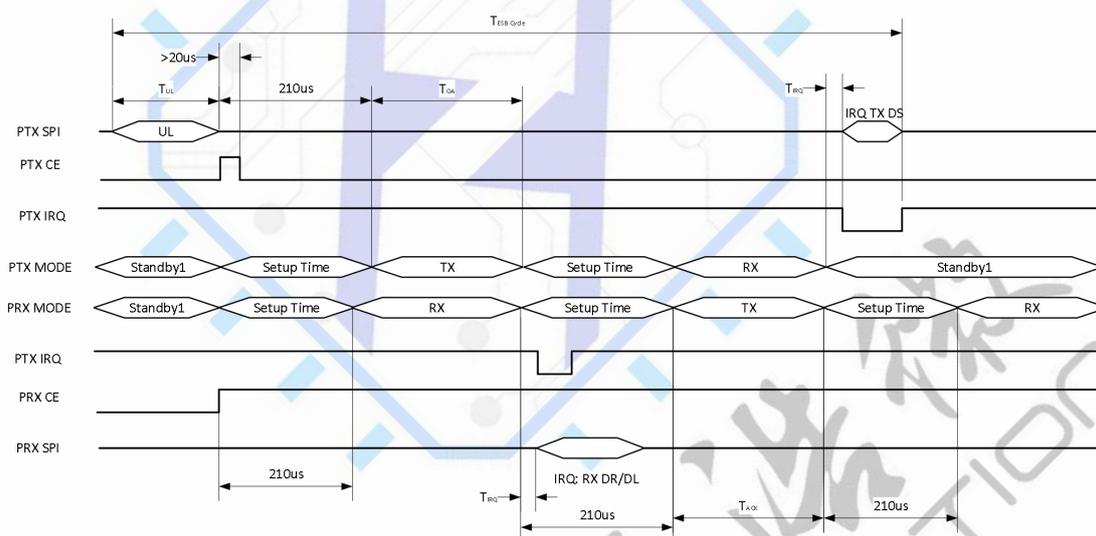


图 14 协议引擎时序用于单次数据包上传(2Mbps)

在图 14 中展示了数据包的传输与确认过程。PRX 操作激活 RX 模式 ($CE=1$)，PTX 操作则在 TX 模式中被激活 ($CE=1$ 维持至少 $20\mu s$)。在 $210\mu s$ 后开始传输，并在 TOA (传输时长) 结束后完成传输。

传输结束时 PTX 操作自动切换为 RX 模式以等待来自 PRX 操作的 ACK 数据包。当 PRX 操作接收到数据包时，其将为主机 MCU 设置中断并切换为 TX 模式以发送 ACK。PTX 操作收到 ACK 数据包后，向 MCU 设置中断并清除 TX FIFO 中的数据包。

图 15 展示了当首个 ACK 数据包丢失时的 PTX 传输时序。

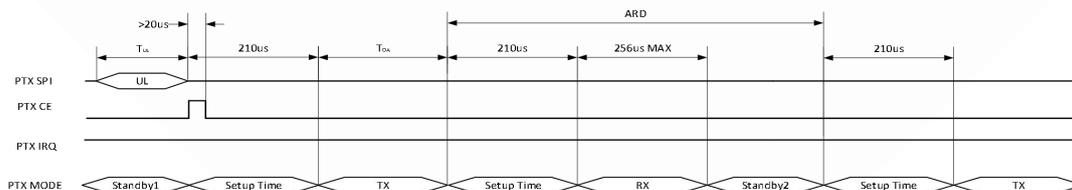


图 15 TX 完成后没收到 ACK(2Mbps)

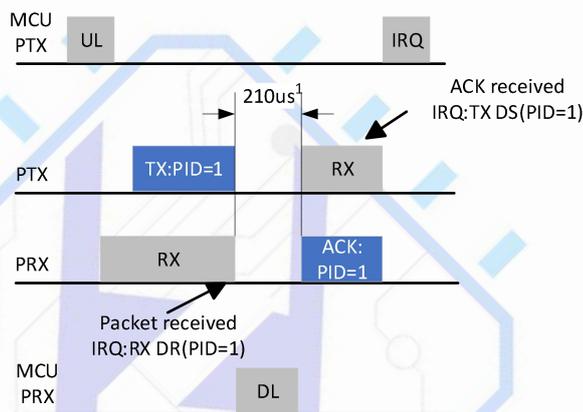
7.8 协议引擎事件图 (Protocol engine transaction diagram)

本节阐述协议引擎自动事务处理的若干场景。图表中的标注项代表中断请求(IRQ)及其他事件。对于 MCU 活动，事件可能置于不同时间点。

注：本节图示表明数据包最早可能的下行传输(DL)至 MCU 及载荷最迟可能的上行传输(UL)至发射器。

7.8.1 含 ACK 及中断的单次事件

图 16 展示基本自动应答机制。当数据包经 PTX 发送且 PRX 接收后，ACK 数据包从 PRX 传输至 PTX。PRX 接收数据包后置位 RX_DR 中断请求(IRQ)，而 TX_DS 中断请求(IRQ)在数据包获应答且 ACK 数据包被 PTX 接收时置位。

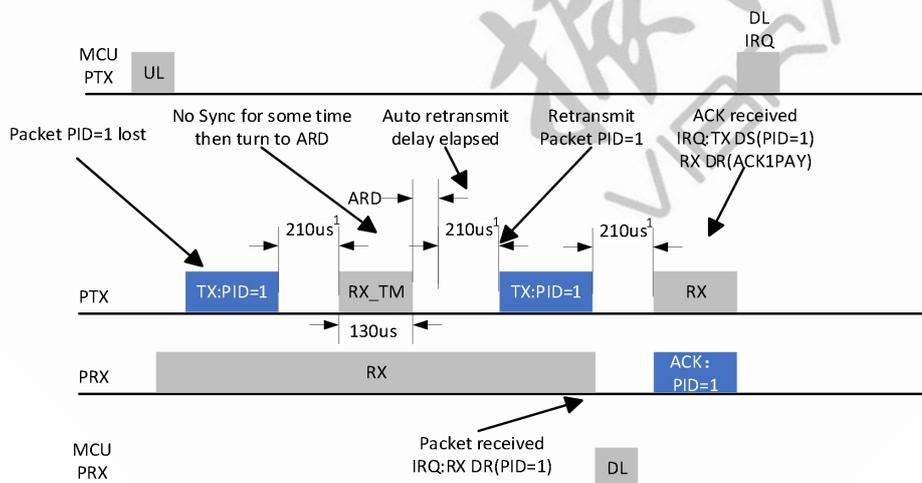


注释 1:收发机转换延时

图 16 单次基本自动应答机制

7.8.2 含丢包情形的单次事件

图 17 展示因首包丢失需重传的场景。数据包传输后，PTX 进入 RX 模式以接收 ACK 包。首包传输后，PTX 在指定时间（含建立时间、RX 超时(RX_TM)及自动重传延迟(ARD)）内等待 ACK 包，若未在该时间槽内收到，PTX 则重新传输数据包（如图 17 所示）。当数据包传输完成时，VG254M 将在 210μs 建立时间后切换至 RX 模式，经历 130μs 接收超时（RX_TM 即 PTX 的接收超时，可设为更短值），随后进入自动重传延迟(ARD)阶段（可设置为 0μs、256μs、512μs 至 3840μs）。



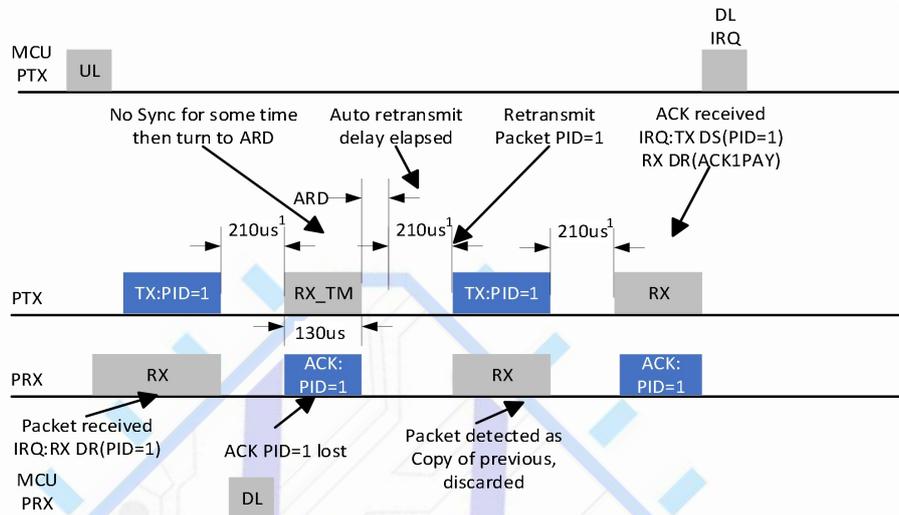
注释 1:收发机转换延时

图 17 自动应答机制状态下丢包一次后重传

当检测到地址码时，PTX 将持续保持在 RX 模式直至数据包接收完成。当重传的数据包被 PRX 接收时（参见图 17），将置位 RX_DR 中断请求(IRQ)并向 PTX 回传 ACK 包。当 ACK 被 PTX 接收后，TX_DS 中断请求(IRQ)被置位。

7.8.3 ACK 包丢失情形的单次事件

图 18 展示 ACK 包丢失后需重传的场景，并标示相应中断请求(IRQ)。

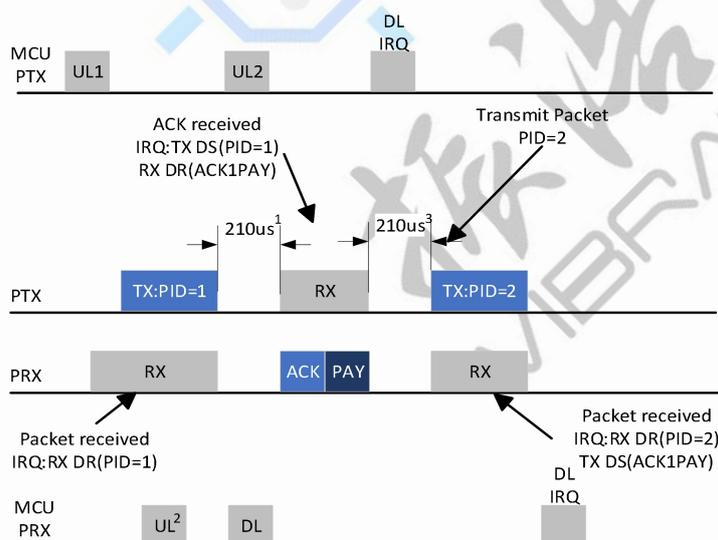


注释 1:收发机转换延时

图 18 ACK 丢失后重传

7.8.4 含 ACK 负载包的单次事件

图 19 展示带负载的基础自动确认场景。当数据包由 PTX 传输并被 PRX 接收后，携带负载的 ACK 包将从 PRX 传输至 PTX。PRX 在接收数据包后将置位 RX_DR 中断请求(IRQ)，而 PTX 则在接收到 ACK 包后置位 TX_DS 中断请求(IRQ)。在 PRX 端，针对 ACK 包负载的 TX_DS 中断请求(IRQ)需在接收到来自 PTX 的新数据包后置位。图 19 中 IRQ 标注位置表明微控制器(MCU)响应中断的触发时点。



注释 1:收发机转换延时

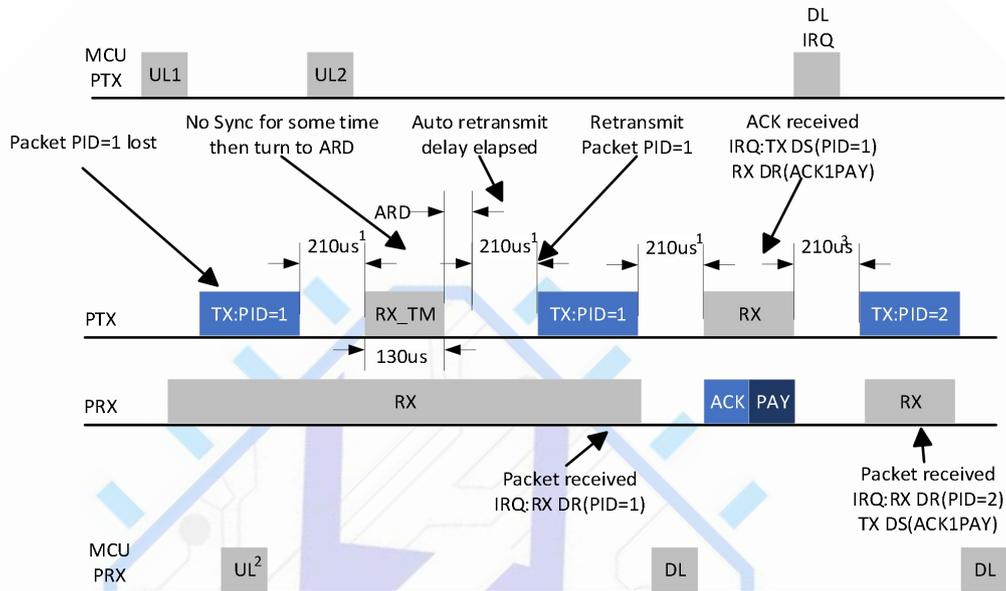
注释 2:上传 Ack 包负载

注释 3:MCU 定义的 PTX 延时 (≥210us)

图 19 含 ACK 负载的 TX/RX 循环及相应中断请求(IRQ)

7.8.5 含 ACK 负载包及数据包丢失情形的单次事务

图 20 展示初始数据包丢失场景：需在 PRX 端 RX_DR 中断请求(IRQ)置位前执行重传。PTX 端在接收 ACK 包后将同时置位 TX_DS 与 RX_DR 中断请求(IRQ)。当 PRX 端接收到第二数据包 (PID=2) 时，将置位 RX_DR (PID=2)及 TX_DS (ACK 包负载) 中断请求(IRQ)。



注释 1:收发机转换延时

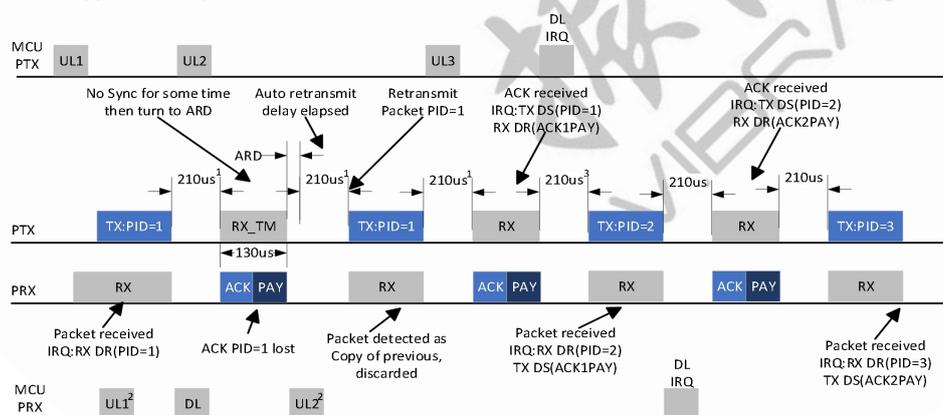
注释 2:上传 Ack 包负载

注释 3:MCU 定义的 PTX 延时 ($\geq 210\mu s$)

图 20 数据包传输失败时的 TX/RX 循环及相应中断请求(IRQ)

7.8.6 含 ACK 负载包传输及首 ACK 包丢失的双事务处理

图 21 展示 ACK 包丢失场景：需在 TX_DS 中断请求(IRQ)置位前执行重传，但 RX_DR 中断请求(IRQ)将即刻置位。数据包重传 (PID=1) 将导致该包被废弃。PTX 端在成功接收第二次传输的 ACK 包后，将同时置位 TX_DS 与 RX_DR 中断请求(IRQ)。当 PRX 端接收第二数据包 (PID=2) 时，将置位 RX_DR (PID=2)及 TX_DS (ACK 包负载区段标识符) 中断请求(IRQ)。图示标注引线详解不同事件流与中断请求。



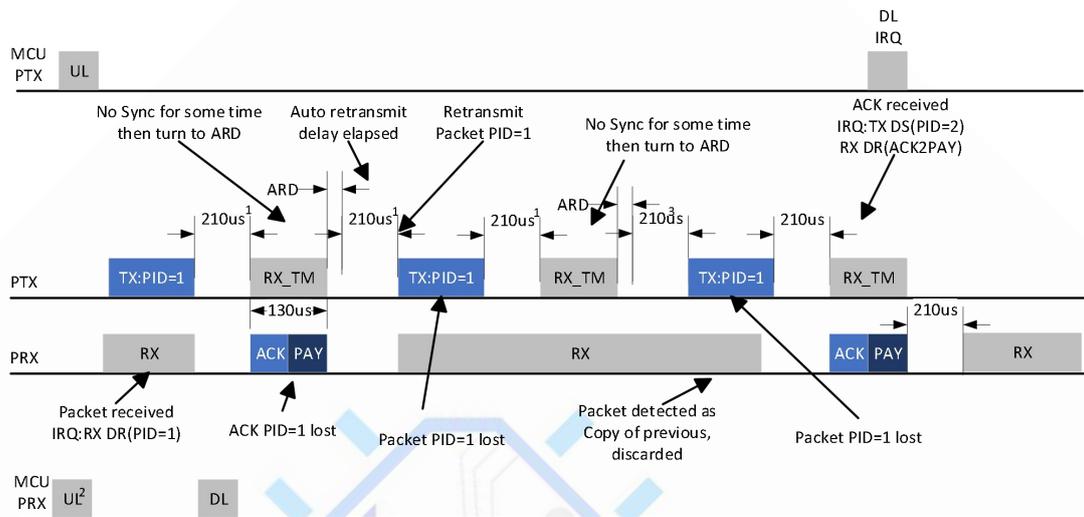
注释 1:收发机转换延时

注释 2:上传 Ack 包负载

注释 3:MCU 定义的 PTX 延时 ($\geq 210\mu s$)

图 21 TX/RX 通信循环含 ACK 及 ACK 故障时的相应中断请求(IRQ)

7.8.7 达到最大重传次数上限的双事件处理



注释 1:收发机转换延时

注释 2:上传 Ack 包负载

注释 3:MCU 定义的 PTX 延时 ($\geq 210\mu s$)

图 22 带 ACK 通信失败时中断请求的 TX/RX 通信循环 (ARC=2)

图 22 呈现含 ACK 负载包的 TX/RX 通信循环及通信失败时的中断请求(IRQ)机制, 自动重传计数器(ARC)预设值为 2。当自动重传计数器(ARC_CNT)超出预设上限(ARC)时, 将置位最大重传中断请求(MAX_RT IRQ)。如图所示, 数据包传输最终触发 MAX_RT 中断请求, 此时发送缓冲器(TX FIFO)内的负载包不予移除, 由 MCU 裁定后续通信协议流程。对 RF 控制寄存器(RFCON)中 CE 位执行电平切换, 将重启相同数据包的发送序列; 亦可使用 FLUSH_TX 指令清空发送缓冲器负载数据。

8、数据接口

数据控制接口提供对射频收发器全部功能的访问权限，其物理层由以下数字信号构成：

****四线制 SPI 配置****

- IRQ：低电平有效中断请求信号（受三路可屏蔽中断源控制）
- CSN：四线 SPI 片选信号
- SCK：四线 SPI 时钟信号
- MOSI：四线 SPI 主机输出从机输入信号
- MISO：四线 SPI 主机输入从机输出信号

通过单字节 SPI 指令操作，可于全工作模式下驱动 VG254M 数据缓冲器或寄存器映射表。

8.1 特性

- 专用单字节 SPI 指令：快速访问高频使用功能
- 0-8Mbps 自适应三线/四线 SPI 接口
- 8 位固定长度指令集
- 可灵活配置寄存器映射表
- 双向单级全深度发送/接收缓冲器（TX/RX FIFO）

8.2 功能描述

SPI 接口为标准串行外设接口，最大数据传输速率 10Mbps。

8.3 SPI 操作机制

本章节定义 SPI 指令及时序逻辑：

8.3.1 SPI 命令

SPI 命令见表 9。所有指令均以 CSN 线高→低电平切换启动。当指令字通过 MOSI 引脚移入时，状态寄存器值同步通过 MISO 引脚移出。

数据传输格式遵循以下协议栈：

<指令字：单字节（高位(MSBit)→低位(LSBit)先行）>

<数据字节序列：先传输 LSByte,后传输 MSByte，字节内高位(MSBit)优先>

表 9 RF 收发机 SPI 命令

Command	Command word (binary)	#Data bytes	Operation
R_RESISTER	000AAAA	1 to 5 LSByte first	读取指令与状态寄存器，AAAAA 是 5 位寄存器映射地址
W_RESISTER	001AAAA	1 to 5 LSByte first	写入指令与状态寄存器，AAAAA=5 位寄存器映射地址
R_TX_PAYLOAD	0110 0001	1 to 32 LSByte first	读取接收负载 (RX 模式)：读取 1-32 字节(从字节 0 起始)，读取后自动清空 FIFO
W_TX_PAYLOAD	1010 0000	1 to 32 LSByte first	写入发送负载 (TX 模式)：写入 1-32 字节(从字节 0 起始)
FLUSH_TX	1110 0001	0	清空发送缓冲器 (TX 模式)
FLUSH_RX	1110 0010	0	清空接收缓冲器 (RX 模式)。禁止在 ACK 响应传输期间执行 (将中断确认包传输)
REUSE_TX_PL	1110 0011	0	重用最后传输负载 (PTX 操作)： • 该状态将持续至执行 W_TX_PAYLOAD 或 FLUSH_TX。禁止在数据包传输过程中激活/关闭
R_RX_PL_WID	0110 0000	1	读取接收缓冲器顶端负载宽度，若读取值 >32 字节，需立即清空接收缓冲器
W_ACK_PAYLOAD	1010 1PPP	1 to 32 LSByte first	RX 模式有效，在 PPP 数据通道 (000-101) 写入 1-32 字节 (字节 0 起始) ACK 负载。最多允许 3 个待处理 ACK 负载。同一通道遵循先入先出原则。
W_TX_PAYLOAD_NO_ACK	1011 0000	1 to 32 LSByte first	TX 模式有效，对本数据包禁用自动确认
RF_RSTN	10001011	0	复位物理层与射频状态命令
RTC_LOCK	10010000	0	实时时钟锁存当前值命令
CE_HIGH	1101 0101	0	CE 置 1 命令
CE_LOW	1101 0110	0	CE 清零命令
CE_PULSE	11011001	0	CE 发脉冲命令

W_REGISTER 和 R_REGISTER 指令可操作单字节或多字节寄存器。访问多字节寄存器时，首先生效最低有效字节的最高位读写操作。可在多字节寄存器全部字节写入完成前终止写操作，未写入的最高位字节保持不变。例如，仅需向 RX_ADDR_P0 寄存器写入单字节即可修改其最低有效字节。芯片选择信号 (CSN) 从高电平转为低电平后，状态寄存器的内容始终通过 MISO 引脚被读取。

注意：STATUS 寄存器中的 2 位管道信息在中断请求 (IRQ) 高电平转低电平时更新。

若在中断请求 (IRQ) 高电平转低电平期间读取状态寄存器，管道信息不可靠。

8.3.2 SPI 时序

SPI 操作与时序如图 23 至图 24 所示。VG254M 必须在待机、休眠或掉电模式下才能向配置寄存器执行写入操作。在图 23 至图 24 中使用以下缩写：

表 10 简写说明

缩写	描述
Cn	命令
Dn	数据。注意：LSByte 先发，每字节 MSB 先发
Sn	状态
X	无关数据

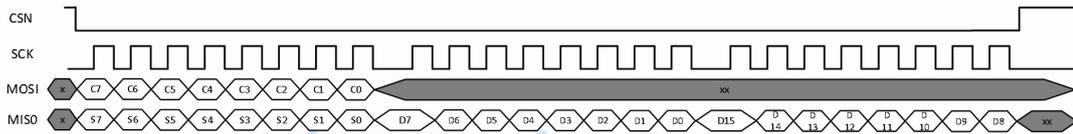


图 23 SPI 读操作

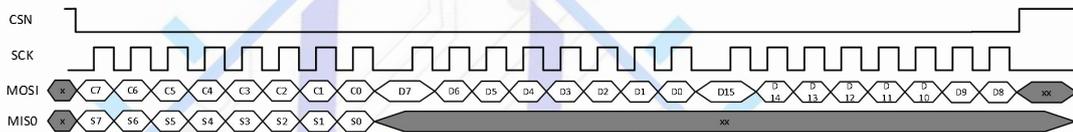


图 24 SPI 写操作

8.4 FIFO

数据 FIFO 用于存储待发送的有效载荷 (TX FIFO) 或已接收可输出的有效载荷 (RX FIFO)。FIFO 在 PTX 模式和 PRX 模式下均可访问。RF 收发器包含以下 FIFO:

- TX 单层 32 字节 FIFO
- RX 单层 32 字节 FIFO

两个 FIFO 均配备控制器并通过专用 SPI 指令经 SPI 接口访问。PRX 模式中的 TX FIFO 可存储供不同 PTX 设备的 ACK 数据包的有效载荷。若 TX FIFO 存在发往某通道的有效载荷，则按先进先出原则处理。当所有待处理有效载荷均指向链路丢失的 PTX 通道时，PRX 的 TX FIFO 将被阻塞。此时 MCU 可通过 FLUSH_TX 指令清空 TX FIFO。

PRX 的 RX FIFO 最多容纳来自一个 PTX 设备的有效载荷，PTX 的 TX FIFO 最多存储一个有效载荷。可通过三条指令写入 TX FIFO: PTX 模式下的 W_TX_PAYLOAD、W_TX_PAYLOAD_NO_ACK，以及 PRX 模式下的 W_ACK_PAYLOAD。三条指令均访问 TX_PLD 寄存器。

RX FIFO 在 PTX/PRX 模式下可通过 R_RX_PAYLOAD 指令读取。该指令访问 RX_PLD 寄存器。若 MAX_RT 中断触发，PTX 的 TX FIFO 中的有效载荷不会被移除。

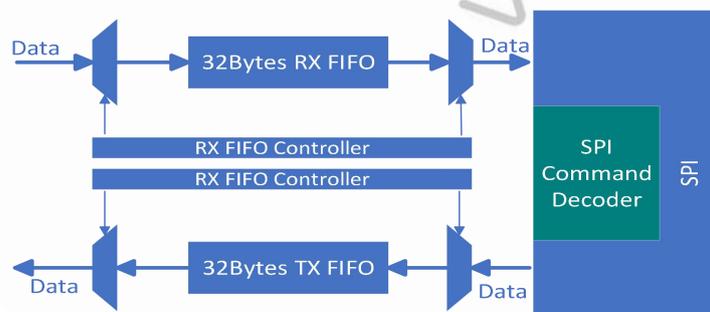


图 25 FIFO 结构图

可通过 FIFO_STATUS 寄存器读取 TX FIFO 与 RX FIFO 是满还是空的状态。

8.5 中断

VG254M 拥有低电平有效的中断（IRQ）引脚。当状态机将 STATUS 寄存器中的 TX_DS 中断、RX_DR 中断或 MAX_RT 中断置为高电平时，IRQ 引脚被激活。当 MCU 向 STATUS 寄存器的中断源比特位写入'1'时，IRQ 引脚复位。CONFIG 寄存器中的中断屏蔽用于选择允许触发 IRQ 引脚的中断源。将任一 MASK 比特位置为高电平会使能对应中断源。默认状态下所有中断源均被使能。

注：STATUS 寄存器中的 2 比特通道信息在 IRQ 高电平向低电平跳变期间更新。若在 IRQ 高电平向低电平跳变期间读取 STATUS 寄存器，通道信息将不可靠。



9、应用电路图

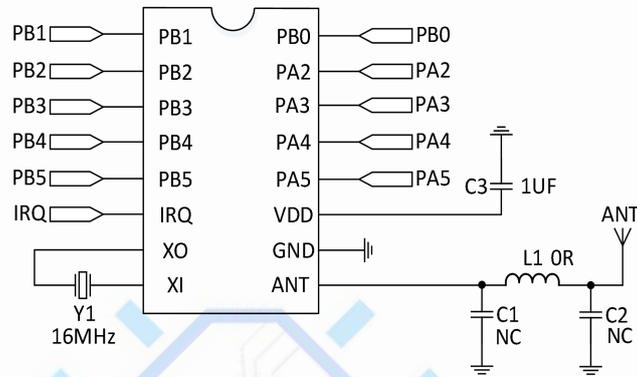


图 26 参考应用电路图

振浩微
VIBRATION

10、封装外形

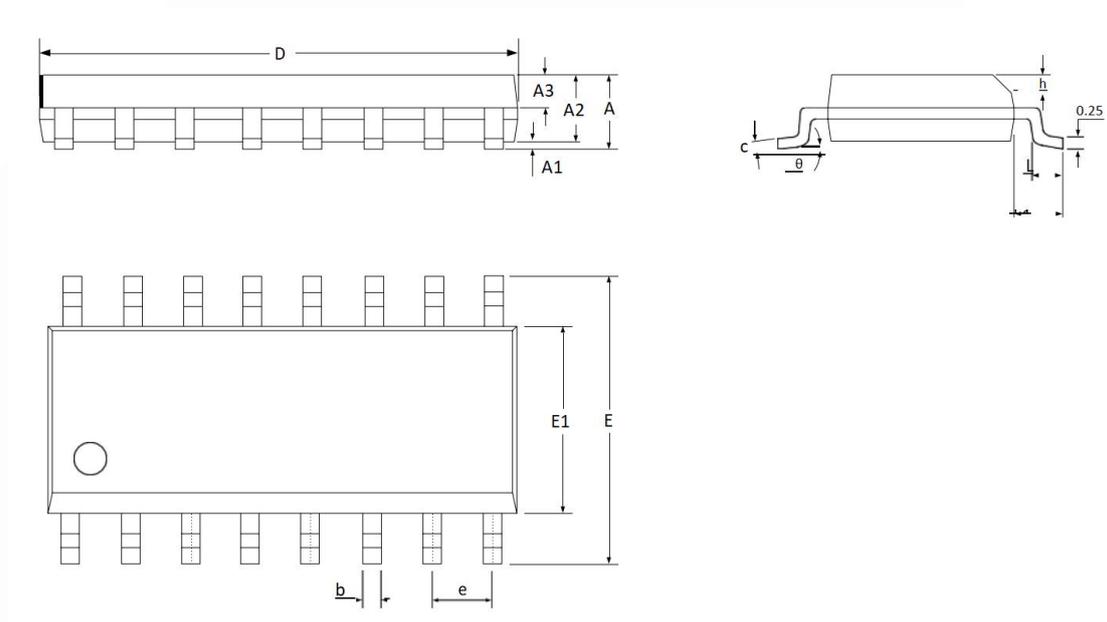


图 27 SOP16 封装示意图

表 11 封装尺寸参数

符号	尺寸 (毫米 mm)		
	最小值	典型值	最大值
A	-	-	1.75
A1	0.10	-	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	-	0.48
c	0.21	-	0.26
D	9.80	9.90	10.00
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27 BSC		
h	0.25	-	0.50
L	0.50	-	0.80
L1	1.05 BSC		
θ	0	-	8°