

UM2052 用户手册

版本：V1.2



广芯微电子（广州）股份有限公司

<http://www.unicmicro.com/>

条款协议

本文档的所有部分，其著作权归广芯微电子（广州）股份有限公司（以下简称广芯微电子）所有，未经广芯微电子授权许可，任何个人及组织不得复制、转载、仿制本文档的全部或部分组件。本文档没有任何形式的担保、立场表达或其他暗示，若有任何因本文档或其中提及的产品所有资讯所引起的直接或间接损失，广芯微电子及所属员工恕不为其担保任何责任。除此以外，本文档所提到的产品规格及资讯仅供参考，内容亦会随时更新，恕不另行通知。

1. 本文档中所记载的关于电路、软件和其他相关信息仅用于说明半导体产品的操作和应用实例。用户如在设备设计中应用本文档中的电路、软件和相关信息，请自行负责。对于用户或第三方因使用上述电路、软件或信息而遭受的任何损失，广芯微电子不承担任何责任。
2. 在准备本文档所记载的信息的过程中，广芯微电子已尽量做到合理注意，但是，广芯微电子并不保证这些信息都是准确无误的。用户因本文档中所记载的信息的错误或遗漏而遭受的任何损失，广芯微电子不承担任何责任。
3. 对于因使用本文档中的广芯微电子产品或技术信息而造成的侵权行为或因此而侵犯第三方的专利、版权或其他知识产权的行为，广芯微电子不承担任何责任。本文档所记载的内容不应视为对广芯微电子或其他人所有的专利、版权或其他知识产权作出任何明示、默示或其它方式的许可及授权。
4. 使用本文档中记载的广芯微电子产品时，应在广芯微电子指定的范围内，特别是在最大额定值、电源工作电压范围、热辐射特性、安装条件以及其他产品特性的范围内使用。对于在上述指定范围之外使用广芯微电子产品而产生的故障或损失，广芯微电子不承担任何责任。
5. 虽然广芯微电子一直致力于提高广芯微电子产品的质量和可靠性，但是，半导体产品有其自身的具体特性，如一定的故障发生率以及在某些使用条件下会发生故障等。此外，广芯微电子产品均未进行防辐射设计。所以请采取安全保护措施，以避免当广芯微电子产品在发生故障而造成火灾时导致人身事故、伤害或损害的事故。例如进行软硬件安全设计（包括但不限于冗余设计、防火控制以及故障预防等）、适当的老化处理或其他适当的措施等。

版本修订

版本	日期	描述
V1.0	2022.01.12	初始版
V1.1	2022.04.29	中文修订版本
V1.2	2022.07.08	调整中文翻译内容及优化文档排版

目录

1	产品概述	1
1.1	特性	1
1.2	系统框图	3
2	封装及管脚描述	4
2.1	封装管脚分布	4
2.2	引脚功能描述	5
3	射频控制	6
3.1	工作模式	6
3.1.1	状态图	6
3.1.2	Power-Down 模式	7
3.1.3	Sleep 模式	8
3.1.4	Standby 模式	8
3.1.5	RX 模式	8
3.1.6	TX 模式	8
3.1.7	工作模式配置	9
3.1.8	时间	9
3.2	数据率	10
3.3	RF 信道频率	10
3.4	PA 控制	10
3.5	RX/TX 控制	10
4	协议引擎	11
4.1	特性	11
4.2	协议引擎概述	11
4.3	协议引擎数据包格式	12
4.3.1	前导码	12
4.3.2	地址	12
4.3.3	保护位	12
4.3.4	数据标识(PCF)	12
4.3.5	Payload 长度	13
4.3.6	Payload	13
4.3.7	CRC 校验	14
4.3.8	自动打包	14
4.3.9	自动拆包	16
4.4	自动包交互处理	17
4.4.1	自动应答	17
4.4.2	自动重传 (ART)	17
4.5	协议引擎流程图	18
4.5.1	PTX 操作	18
4.5.2	PRX 操作	20
4.6	多从机	22
4.7	协议引擎时序	24
4.8	协议引擎交互框图	25
4.8.1	带 ACK 包和中断事件的单一数据包处理	25
4.8.2	未收到数据包的单一数据包处理	26
4.8.3	未收到 ACK 包的单一数据包处理	26
4.8.4	ACK 数据包的单一数据包处理	27
4.8.5	ACK 数据包的单一数据包处理和数据包丢失	27

4.8.6	ACK 数据包的双数据包处理及第一个 ACK 包丢失	28
4.8.7	最大重传下的双数据包处理	28
5	数据和控制接口	29
5.1	特征	29
5.2	功能描述	29
5.3	SPI 操作	29
5.3.1	SPI 命令	29
5.3.2	SPI 时序	31
5.4	数据 FIFO	32
5.5	中断	32
6	寄存器映射	34
7	电气参数	35
7.1	绝对最大额定值	35
7.2	运行条件	35
7.3	电流功耗	35
7.4	频率综合器	36
7.5	接收机	36
7.6	发射机	36
7.7	晶体	36
7.8	直流特性	37
8	典型应用原理图	38
9	封装尺寸	39
9.1	SOP8 (4.9*3.9MM)	39
9.2	ESOP8 (4.9*3.9MM)	40

图目录

图 1-1: UM2052 系统框图	3
图 2-1: SOP8 封装管脚分布图	4
图 2-2: ESOP8 封装管脚分布图	4
图 3-1: 射频控制状态图	7
图 4-1: 自动数据包解析	16
图 4-2: 数据管道 0-2 地址分配	23
图 4-3: 多从机地址分配	23
图 4-4: NO_ACK 使能的发射时序图	24
图 4-5: 协议引擎上传数据包的时序图 (2Mbps)	24
图 4-6: 第一个 ACK 未收到的协议引擎时序图(2Mbps)	25
图 4-7: 带 ACK 和中断的 TX/RX 时序图	25
图 4-8: 当第一个包发送失败时, 带 ACK 和中断的 TX/RX 时序图	26
图 4-9: T 未收到 ACK 包时, 带 ACK 和中断的 TX/RX 时序图	26
图 4-10: ACK 数据包的 TX/RX 时序图	27
图 4-11: 未收到数据包, TX/RX 时序图和相关的中断	27
图 4-12: 未收到 ACK 数据包, 携带 ACK 数据的 TX/RX 时序图和相关中断	28
图 4-13: 产生最大重传的 TX/RX 时序图	28
图 8-1: SOP8 封装应用原理图	38
图 8-2: ESOP8 封装应用原理图	38
图 9-1: SOP8 封装图	39
图 9-2: ESOP8 封装图	40

表目录

表 2-1: 引脚功能说明	5
表 3-1: UM2052 工作模式	9
表 3-2: UM2052 模式切换时间	9
表 3-3: 输出功率设置	10
表 4-1: 数据帧格式	12
表 4-2: PCF (Packet control field)	12
表 5-1: SPI 命令	30
表 5-2: 图 5-1~图 5-4 使用的缩写名称表	31
表 7-1: 芯片绝对最大额定值	35
表 7-2: 运行条件参数	35
表 7-3: 电流功耗参数	35
表 7-4: 频率综合器参数	36
表 7-5: 接收机参数	36
表 7-6: 发射机参数	36
表 7-7: 晶体参数	36
表 7-8: 直流特性	37

1 产品概述

UM2052 是一款工作于 2.400 -2.4835GHz ISM 频段、专为低功耗无线场合设计、集成嵌入式基带协议引擎的 2.4GHz 无线收发机芯片。芯片采用 GFSK/FSK 调制解调方式,支持 1Mbps 和 2Mbps 数据速率,内部集成了 LDO 电源,保证 1.9~3.6V 宽电压范围内稳定工作。

嵌入式基带引擎基于包通信原理,支持多种通信模式,可以手动或全自动协议操作,内部集成收发 FIFO,可以保证芯片与 MCU 数据连续传输。协议引擎能处理所有高速操作从而降低系统消耗。

UM2052 具有非常低的系统应用成本,只需要一个 MCU 和少量外围无源器件既可以组成一个无线数据收发系统。

应用场景:

- 无线鼠标、键盘
- 远程控制
- 消费类电子产品
- 无线游戏手柄
- 无线音频设备
- 玩具
- 网络电话和无线耳机
- 个人健康和娱乐设施
- 智能家居

1.1 特性

UM2052 包含以下特性:

- **射频**
 - 支持长包模式 (最长 255 字节)
 - 共用收发通道
 - GFSK/FSK 调制
 - 数据率支持 1Mbps 和 2Mbps
 - 工作在 1Mbps 时非重叠信道间隔为 1MHz
 - 工作在 2Mbps 时非重叠信道间隔为 2MHz
- **发射机**
 - 可编程发射功率: -34~+8dBm

- 发射电流：17mA @ 0dBm
- **接收机**
 - 快速 AGC 功能
 - 集成通道滤波器
 - 接收电流：18mA @ 2Mbps
 - 接收灵敏度：-87dBm @ 2Mbps
-90dBm @ 1Mbps
- **频率综合器**
 - 高度集成频率综合器
 - 无需外部环路滤波器 (LF)
 - 晶振：16MHz±60ppm
- **协议引擎**
 - 动态数据包长度 1 到 32 字节
 - 自动数据包处理
 - 自动数据包传送
 - 3 个数据通道，可用于 1:3 的星型网络
- **电源管理**
 - 内部集成 LDO
 - 支持 1.9 到 3.6V 电源电压
 - 优化电源管理支持空闲模式快速上电 (时间)
 - Power Down 电流：2μA
 - Sleep 电流：50μA
- **接口**
 - 支持 3 线或 4 线 SPI 接口 (SOP8 封装固定 3 线 SPI)
 - SPI 数据率最大支持 8Mbps
 - 独立的 32 字节 RX FIFO 和 TX FIFO
 - 宽电压 GPIO，耐压 5V

1.2 系统框图

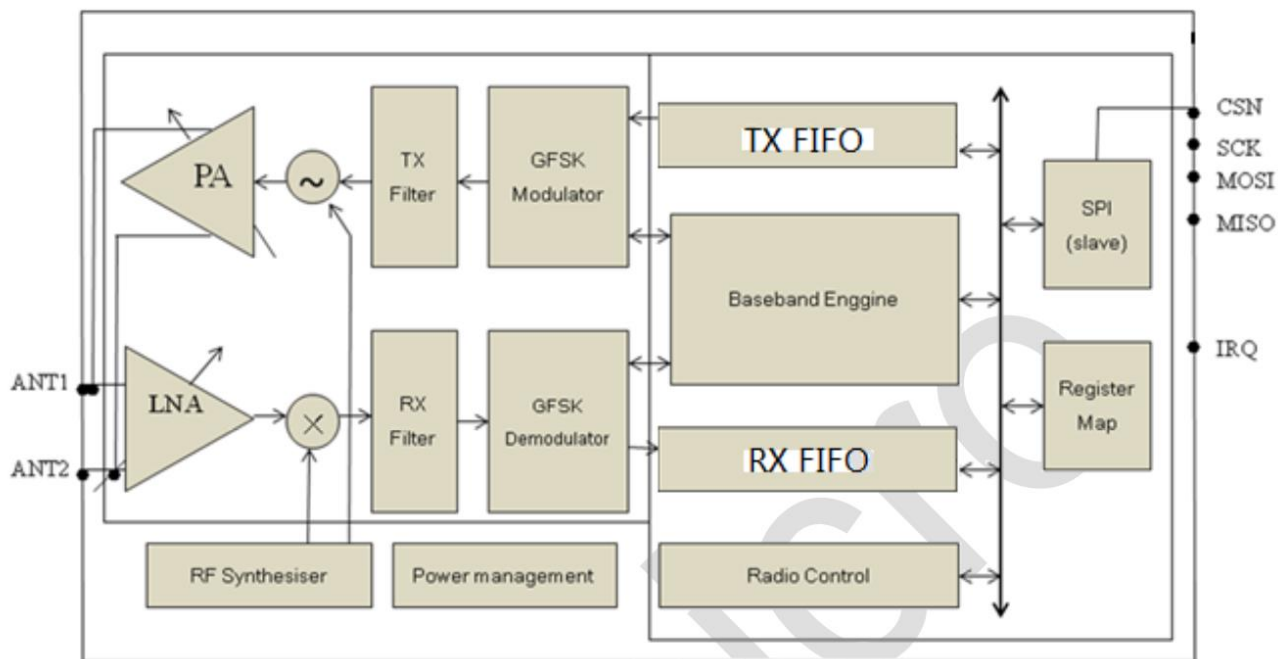


图 1-1: UM2052 系统框图

2 封装及管脚描述

目前 UM2052 支持两种封装，SOP8 和 ESOP8。SOP8 封装有 8 个管脚，其中数字引脚 3 个，仅支持 3 线 SPI。MCU 一般采用 GPIO 模拟 SPI-3，如想使用硬件 SPI 则需 MCU 支持 SPI-3，或使用 ESOP8 封装。

ESOP8 封装有 9 个管脚，IC 底部有一个 PAD，其中数字引脚有 4 个，支持标准 4 线 SPI。标准 4 线 SPI 能支持更高 SPI 速率。

2.1 封装管脚分布

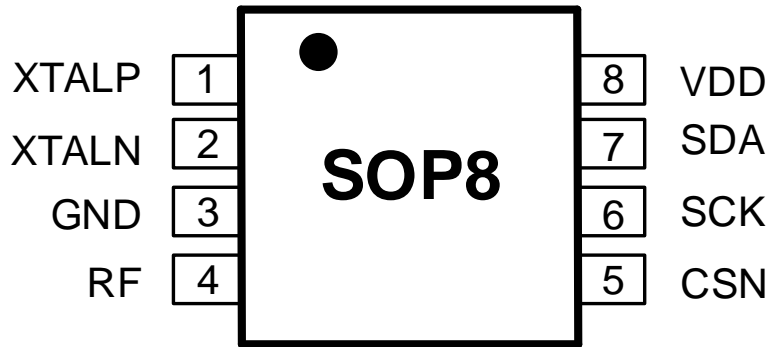


图 2-1: SOP8 封装管脚分布图

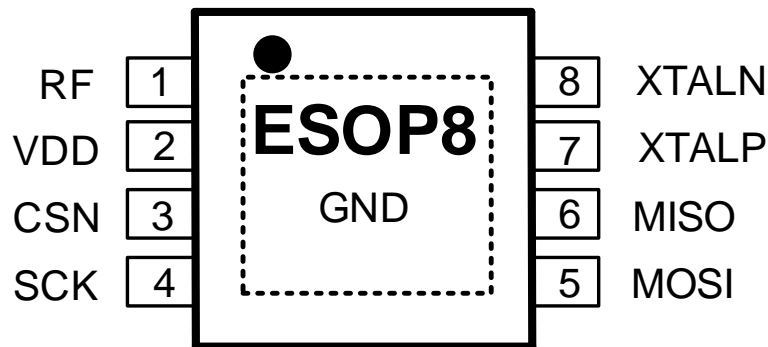


图 2-2: ESOP8 封装管脚分布图

2.2 引脚功能描述

表 2-1: 引脚功能说明

封装引脚编号		管脚名称	IO Type	功能描述
SOP8	ESOP8			
1	7	XTALP	AI	Crystal+
2	8	XTALN	AI	Crystal-
3	9	GND	G	地
4	1	RF	AIO	射频信号输入输出
5	3	CSN	DI	SPI 片选脚
6	4	SCK	DI	SPI 时钟脚
7	-	SDA	DIO	SPI 数据脚
8	2	VDD	P	电源供电 (+1.9~3.6V)
-	5	MOSI	DI	SPI 输入数据脚 (SPI 从设备)
-	6	MISO	DO	SPI 输出数据脚 (SPI 从设备)

说明: RF-射频信号; A-模拟信号; D-数字信号; I-Input; O-Output; G-Ground; P-Power。

3 射频控制

本章节介绍 UM2052 无线收发器的工作模式和用于控制射频工作的参数。

UM2052 内置状态机用于控制芯片工作模式。状态机受控于寄存器配置的值和内部信号。

3.1 工作模式

UM2052 有 5 种工作模式，分别是 Power Down 模式、Sleep 模式、Standby 模式、RX 模式和 TX 模式，本章将详细介绍这些模式。

3.1.1 状态图

工作模式及原理如图 3-1 所示。工作模式可分为以下三种：

- 推荐工作模式：正常工作状态。
- 可能工作模式：一种可能的工作状态，但正常工作时不会使用。
- 过渡状态：在晶体起振和 PLL 锁定过程中使用的状态。

当 VDD 达到 1.9V 或更高时，UM2052 进入上电复位状态，并保持复位状态直到进入 Power Down 模式。

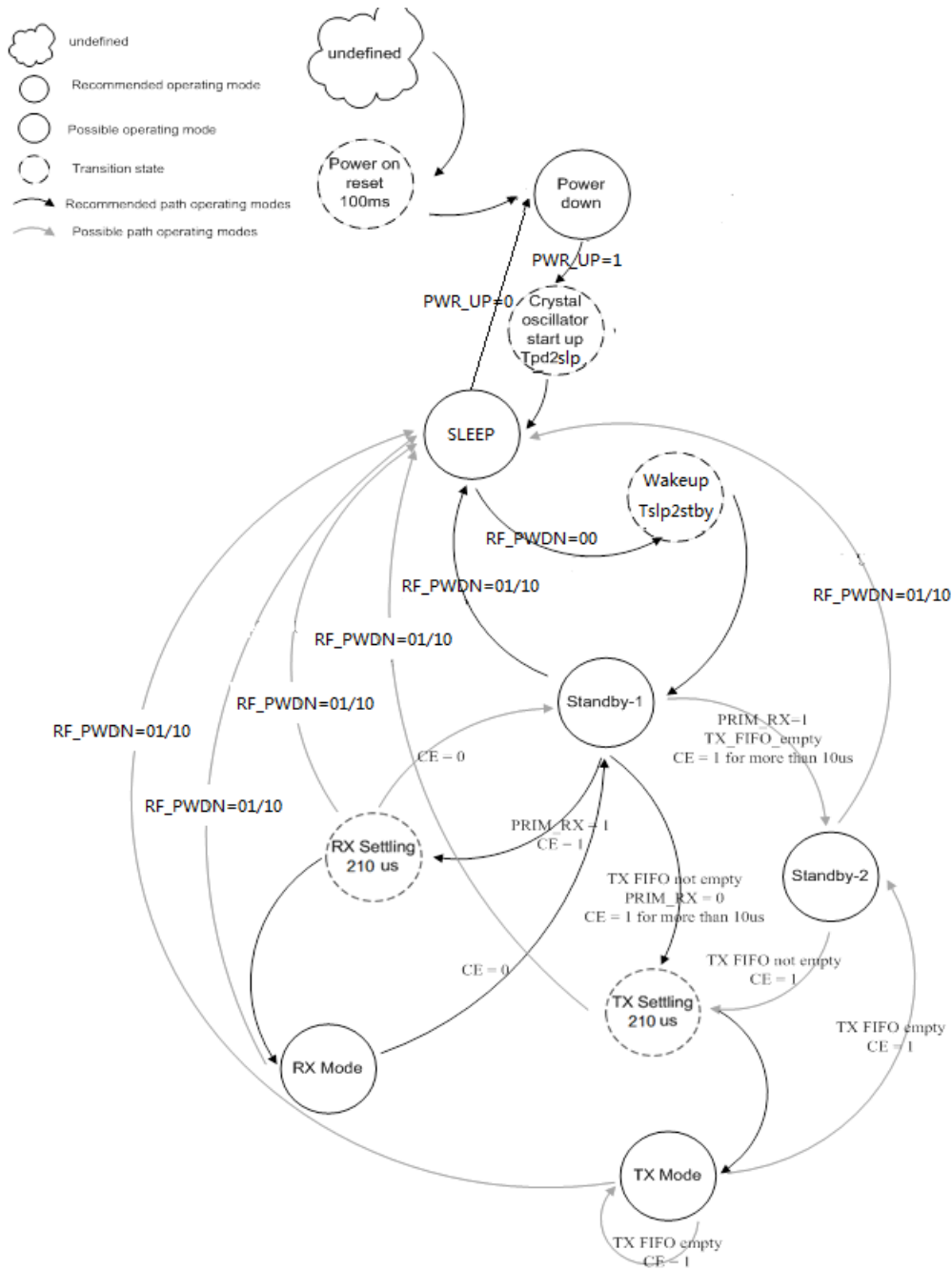


图 3-1：射频控制状态图

3.1.2 Power-Down 模式

在 Power-Down 模式下，UM2052 所有功能关闭，保持电流消耗最小，但寄存器值保持不变且 SPI 正常工作，从而可以进行配置更改及数据寄存器的上传/下载。启动时序如表 3-2 所示。当 UM2052 处于 sleep 模式时，将 CONFIG 寄存器的 PWR_UP 位设置为 low 即可进入 Power down 模式。

3.1.3 Sleep 模式

在 sleep 模式下，UM2052 所有功能关闭，保持电流消耗最小，但寄存器值保持不变且 SPI 和晶振正常工作。

3.1.4 Standby 模式

3.1.4.1 Standby-I 模式

通过设置 CONFIG 寄存器中的 PWR_UP bit 为 1，芯片进入 sleep 模式，然后清除 RF_PWDN 2bit，芯片进入 Standby-I 模式。在 Standby-I 模式下，保持电流消耗最小的同时芯片能够在较短时间内启动。在这种模式下，只有部分晶振是工作的。只有当 CE 为高时，晶振可变换为工作状态，当 CE 设为低，UM2052 将返回 Standby-I 模式。

3.1.4.2 Standby-II 模式

当发送端 TX FIFO 寄存器为空，并且 CE 为高电平时，进入 Standby-II 模式。如果数据包送入 TX FIFO，此时 PLL 立刻启动工作且待其频率锁定延时（210 μ s）后，发射机将数据包发射出去。在 Standby-I 及 Standby-II 模式下，寄存器值保持不变且 SPI 处于工作状态。

注意：从 Standby-I 模式到 standby-II 模式，CE 需要保持至少 20us。

3.1.5 RX 模式

RX 模式是 UM2052 被用作接收器时处于的工作模式。当 PWR_UP 位、PRIM_RX 位和 CE 位设为高时进入 RX 模式。

在 RX 模式下，接收机解调来自射频信道的信号，不断地将解调后的数据呈现给基带协议引擎。基带协议引擎不断地搜索有效的数据包。如果一个有效的数据包被发现(通过匹配地址和 CRC 校验)，该数据包所携带的有效信息会被放在 RX FIFO 的空位置上。如果 RX FIFO 是满的，接收的数据包就会被丢失。

芯片一直处在 RX 模式直到 MCU 将其设置为 Standby-I 模式或 Power Down 模式。但是，如果芯片配置成自动应答模式（使能基带协议引擎），芯片根据协议命令可以进入其他模式。

3.1.6 TX 模式

TX 模式是 UM2052 发送数据包时处于的工作模式，当 PWR_UP 位设为高，PRIM_RX 位设为低，TX FIFO 中有数据且 CE 维持 20 μ s 以上的高电平时进入 TX 模式。

UM2052 在数据包发送完之前都会保持在发送模式。如果 CE=0，芯片返回 standby-I 模式。如果 CE=1，TX FIFO 的状态决定下一个工作状态。如果 TX FIFO 不为空，UM2052 保持 TX 模式并传输下一个数据包。如果 TX FIFO 为空，则芯片进入 Standby-II 模式。

3.1.7 工作模式配置

工作模式的配置说明如下表所示。

表 3-1: UM2052 工作模式

模式	PWR_UP 寄存器	RF_PWDN 寄存器	PRIM_RX 寄存器	CE 命令	FIFO 状态
RX 模式	1	00	1	1	-
TX 模式	1	00	0	1	TX FIFO 数据。在 TX FIFO 模式下将为空
TX 模式	1	00	0	最小 20us 高电平	TX FIFO 数据。在 TX FIFO 模式下将为空
Standby-I 模式	1	00	-	0	没有持续的数据包
Standby-II 模式	1	00	0	1	TX FIFO 为空
Sleep 模式	1	01/10	-	-	-
Power 模式	0	01	-	-	-

3.1.8 时间

本节描述的是各种工作模式下切换所需的时间及 CE 引脚的时序。从 TX 模式切换到 RX 模式，或者从 Standby 模式到 TX 模式或 RX 模式所需的时间（最大 210 μ s），如下表所示。

表 3-2: UM2052 模式切换时间

名称	芯片工作模式	最大值	最小值	备注
T _{pd2slp}	Power Down 模式 → Sleep 模式	150 μ s	-	使用外部时钟
		2ms	-	外部晶体, Ls<30mH
		3ms	-	外部晶体, Ls<60mH
		4.5ms	-	外部晶体, Ls<90mH
T _{slp2stby}	Sleep 模式 → standby 模式	1ms	-	-
T _{stby2a}	Standby 模式 → TX/RX 模式	210 μ s	-	-
T _{hce}	CE 为高电平最小持续时间	-	20 μ s	-
T _{pece2csn}	从 CE 有效到 CSN 拉低需要延时	-	4 μ s	-

UM2052 从 Power down 模式到 TX 或 RX 模式，它必须先经过 Sleep 和 Standby 模式。在 UM2052 离开 Power down 模式后，CE 设为高之前，必须有 T_{pd2slp} 和 T_{slp2stby} 的延迟(见表 3-2)。

注意：如果 V_{DD} 关闭，寄存器值丢失，必须在进入 TX 或 RX 模式前配置芯片。

3.2 数据率

数据率是芯片在传输和接收数据时使用的调制信号速率。UM2052 支持 1Mbps 和 2Mbps 数据率。使用低数据率比高数据率能实现更好的接收灵敏度，但是，高数据率可以实现低电流功耗。RF_SETUP 寄存器中的 RF_DR 位设置数据率，发射机和接收机必须使用相同的数据率才能相互通信。

3.3 RF 信道频率

RF 信道频率决定了芯片所使用的信道的中心频率。在速率为 1Mbps 时，信道所占宽带小于 1MHz。在速率为 2Mbps 时，信道所占宽带小于 2MHz。芯片工作频率为 2.400GHz ~ 2.525GHz。RF 信道频率的分辨率为 1MHz。

在速率为 2Mbps 时，信道所占宽带比信道频率的分辨率更宽。为了确保 2Mbps 模式下的非重叠信道，信道间距必须为 2MHz 或以上。在速率为 1Mbps 时，信道所占宽带等于或低于射频信道频率的分辨率。

RF 信道频率由 RF_CH 寄存器根据以下公式设置：

$$F_0 = 2400 + \text{RF_CH [MHz]}$$

接收机和发射机必须使用相同的信道频率才能相互通信。

3.4 PA 控制

PA(Power Amplifier)控制用于设置芯片功率放大器的输出功率。TX 模式有 4 个等级，可以通过 RF_SETUP 寄存器中的 PA_PWR 位进行设置。具体如下表所示。

测试条件：V_{DD} = 3.0V, V_{SS} = 0V, T_A = 27°C

表 3-3: 输出功率设置

SPI RF-SETUP (PA_PWR[3:0])	RF 输出功率	DC 电流功耗
1000	0dBm	17mA
0100	-6dBm	14.5mA
0010	-12dBm	12.8mA
0001	-18dBm	10.5mA

注意：设置 PA_PWR[3:0]为 1111 可获得最大+8dBm 输出功率。

3.5 RX/TX 控制

RX/TX 由 CONFIG 寄存器中的 PRIM_RX 位控制，可设置 UM2052 芯片为发送或接收模式。

4 协议引擎

协议引擎是在数据链路层规定数据包的通信格式，实现数据包的自动装载与定时、自动应答和自动重传。协议引擎可实现超低功耗和高性能的通信，显著的改善单工和双工通信系统的功率效率，不需要 MCU 的复杂控制。

4.1 特性

协议引擎的主要特性有：

- 1 到 32 字节的动态数据包长度
- 自动包处理
- 自动包交互处理
 - 自动应答
 - 自动重传
- 3 个数据通道，可实现 1:3 星型网络

4.2 协议引擎概述

协议引擎定义了自动包处理和定时。在发射时，装载数据包和传输包的时钟 bits 以供传输。在接收时，在解调信号中不断查询一个有效的地址。如果收到有效地址，则对数据包的剩下部分处理和进行 CRC 校验。如果数据包是有效的，则将数据包搬进 RX FIFO 的空数据段。所有的高速 bit 处理和定时都由协议引擎控制。

协议引擎的自动包传输处理功能，使得可靠的双向链路通信非常容易实现。双向通信的数据包交换在两个收发机之间实现，一个是接收机（PRX），一个是发射机（PTX）。协议规定，由发射机（PTX）发起数据包传输，接收机（PRX）接收到数据包后，回一个 ACK 数据包，由此完成一次双向通信。

自动包传输处理流程如下：

1. 用户通过从发射机（PTX）向接收机（PRX）传输一个数据包开始通信。发射机（PTX）发送完成后自动切到接收模式，等待 ACK 数据包。
2. 如果接收机（PRX）接收到数据包，协议引擎自动装载和发射 ACK 包到发射机（PTX），然后又切回接收模式。
3. 如果发射机（PTX）没有即时收到 ACK 数据包，则在一定延时后，自动重传原始数据包，然后又切回接收模式等待 ACK 数据包。

在协议引擎中，规定了最大重传次数和本次发送到重传之间的延时。所有的自动处理都不需要 MCU 参与。

4.3 协议引擎数据包格式

本节将介绍协议引擎的数据包格式。数据包格式包含一个前导码字段、地址字段、保护位字段、数据标识字段、Payload 字段和 CRC 字段。如下表所示。

表 4-1: 数据帧格式

前导码	地址	保护位	数据标识	Payload	CRC
1Byte	4~5Byte	2Byte	9bit	0~32Byte	1~2Byte

4.3.1 前导码

前导码用于同步接收机解调的比特流。前导码是一个字节,要么是 01010101,要么是 10101010。当地址的第一位是“1”时,前导码自动设置为 10101010,当地址的第一位是“0”时,前导码自动设置为 01010101。

4.3.2 地址

地址是针对接收机来说的。地址能确保数据包被正确的接收机检测和接收,防止多个 UM2052 通信系统中的相互冲突。地址可通过 AW 寄存器配置成 4 字节或者 5 字节。

4.3.3 保护位

2 字节的保护位有更好的同步特性。

4.3.4 数据标识(PCF)

如下表所示,数据标识有 9 个 bit,包含 6bit Payload 长度字段、2bit PID (packet Identity)字段和 1bit NO_ACK 标志。

表 4-2: PCF (Packet control field)

Payload 长度	PID	NO_ACK
6bit	2bit	1bit

4.3.5 Payload 长度

6bit 的数据长度标识数据的字节数。数据字节数为 0~32 字节。000000 表示 0 字节（只用在空 ACK 数据包），数据长度为 0 可以用 R_RX_PAYLOAD 读出无数据。100000=32 字节。100001=忽略。

只有在动态数据长度功能使能下才能使用此字段。

4.3.5.1 PID (数据包识别)

2bit PID 字段用于检查接收的数据包是新的还是重传的。PID 防止 PRX 送给 MCU 相同的数据包。在 PTX 端，通过 SPI 接收到的每一个新数据包，PID 自动增加。PID 和 CRC 字段被用于确定数据包是新的还是重传的。当发射的数据包丢失时，PID 会保持上一个数据包的 PID 号。如果收到的数据包与上一个数据包有相同的 PID 号，射频收发器进行 CRC 校验，如果 CRC 校验也相同，此数据包为重传数据包，将被丢弃。

4.3.5.2 NO_ACK 位

NO_ACK 位在自动应答功能下使用。设置 NO_ACK 位为高，告知接收机数据包不需要自动应答。

在 PTX 端，您可以通过 **W_TX_PAYLOAD_NOACK** 命令设置数据标识字段的 NO_ACK 位，然而，该功能必须首先被 FEATURE 寄存器中 EN_DYN_ACK 位使能。在发射完数据后，PTX 直接进入 standby-I 模式，PRX 接收数据包时不发送 ACK。

4.3.6 Payload

Payload 数据是用户定义的数据包内容，长度为 0~32 字节。

协议引擎提供两种数据长度：静态和动态。默认为静态数据长度，在此模式下，发射机和接收机之间的所有数据包长度都是一样的。通过设置 RX_PW_Px 寄存器可配置接收机端的静态数据长度。在发射机端，通过设置 TX_FIFO 的时钟字节数可配置数据长度，和接收机的数据长度必须一致。

动态数据长度（DPL）使能后，发射机发送给接收机的数据长度可变，这意味着，对于一个系统有不同的数据长度。UM2052 的 DPL 功能可以自动解码收到数据包的有效数据长度而不使用 RX_PW_Px 寄存器。MCU 可以通过 R_RX_PL_WID 命令读出接收的 Payload 数据长度。

注意：使用 R_RX_PL_WID 命令时，需要一直检测数据包长度是否为 32 字节或是更短。如果宽度大于 32 字节，则数据包包含错误信息必须被丢弃，使用 Flush_RX 命令丢弃数据包。

启用 DPL 必须使能 FEATURE 寄存器中的 EN_DPL 位。在 RX 模式下，DYNPD 寄存器必须被设置。数据从 PTX 发送到启用了 DPL 的 PRX，必须在 DYNPD 寄存器设置 DPL_P0 位。

4.3.7 CRC 校验

CRC 是数据包的错误检测机制，它可以是 1 或 2 字节，计算地址、数据标识和数据。

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

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

CRC 字节数通过 CONFIG 寄存器中的 CRCO 位设置。如果 CRC 检测错误，数据包不被接收。

4.3.8 自动打包

自动打包将前导码、地址、数据标识、Payload 和 CRC 打包后发送出去，具体流程如下图所示。

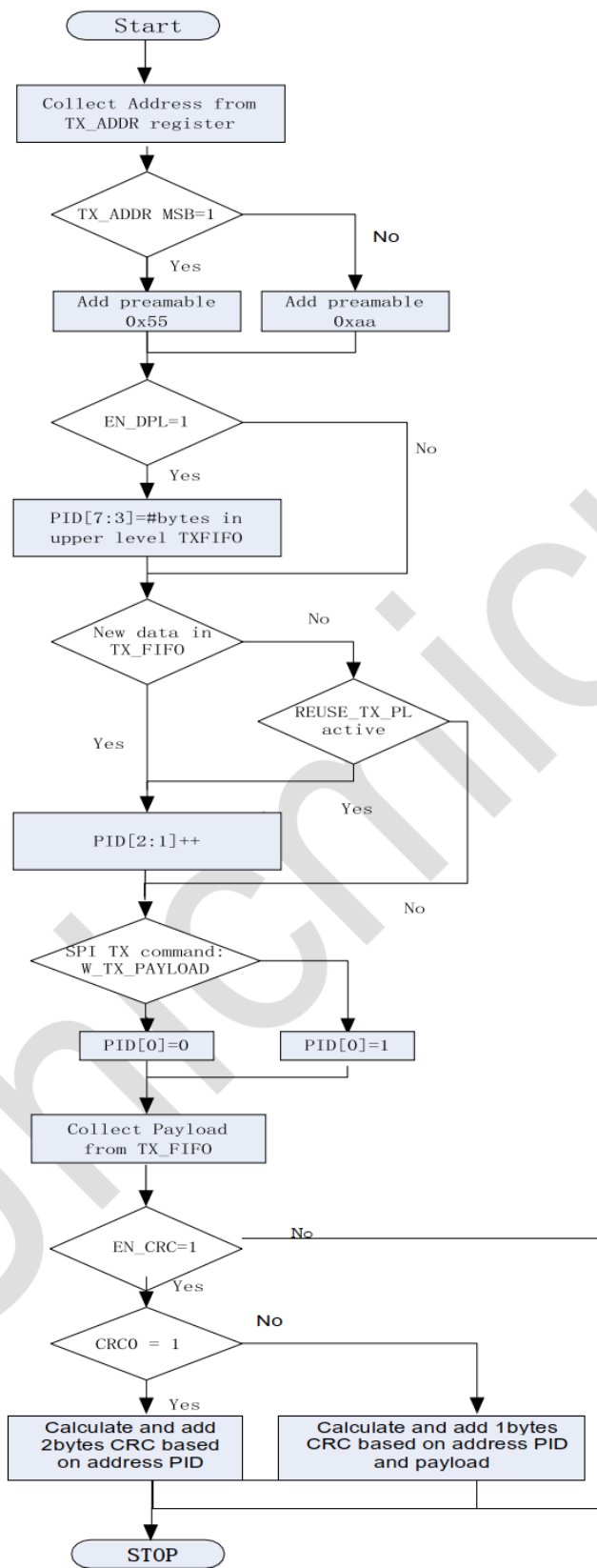


图 5-3: 自动打包流程图

4.3.9 自动拆包

收到的数据包被验证后,进行拆包,并取出数据,然后送给 RX FIFO,同时产生接收中断(RX_DR IRQ), 具体流程如下图所示:

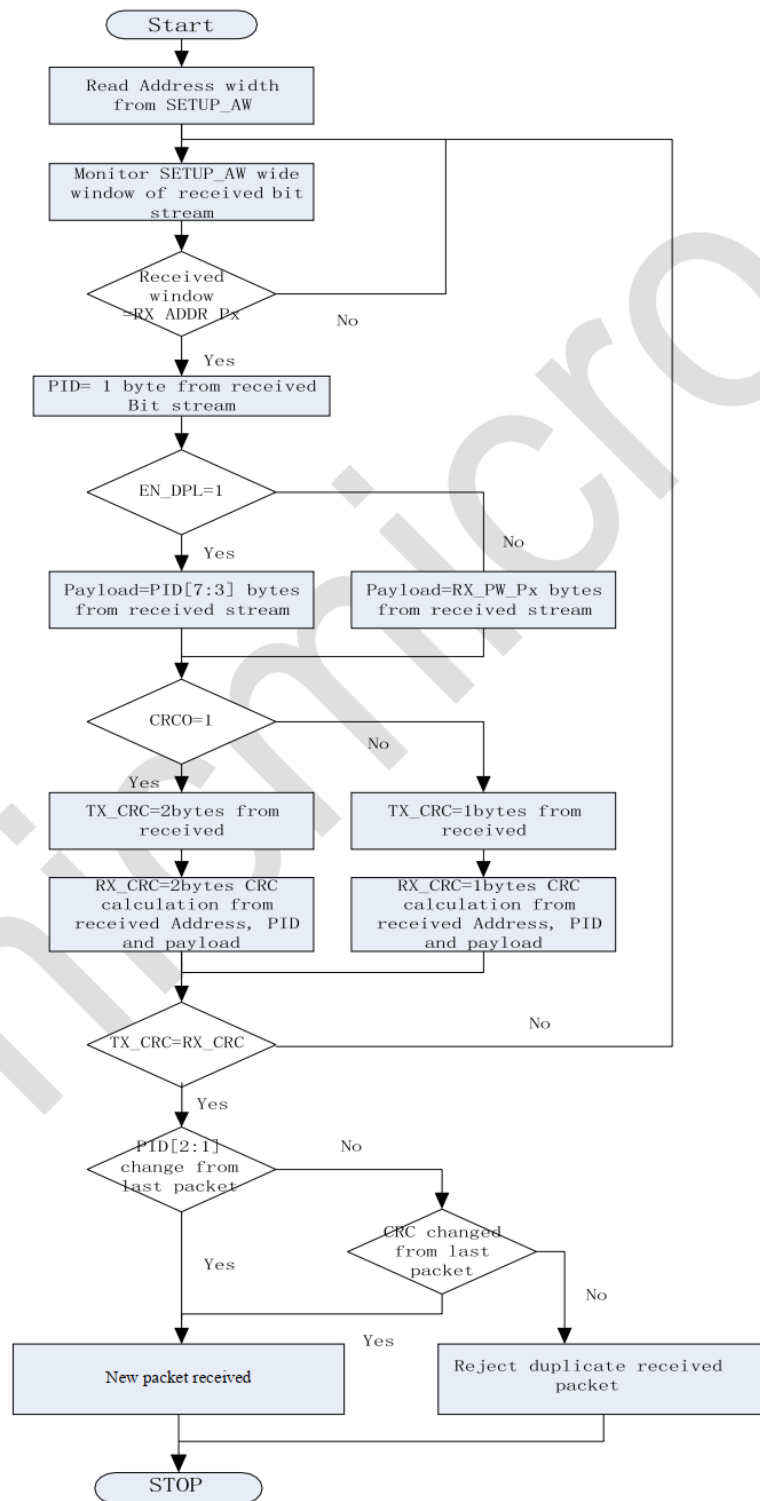


图 4-1: 自动数据包解析

4.4 自动包交互处理

协议引擎用于自动包交互处理主要有两个功能：自动应答和自动重传。

4.4.1 自动应答

自动应答是指接收机（PRX）收到有效的数据包后，自动回给发送机（PTX）一个 ACK 包。自动应答功能减少了 MCU 系统的负载和降低了平均电流消耗。通过设置 EN_AA 寄存器使能自动应答功能。

注意：如果收到的数据包设置了 NO_ACK 标志，则不会执行自动应答。

接收机（PRX）发送给发射机（PTX）的自动应答包，其有效数据是可选的。为了使用该功能，动态数据包长度（Dynamic Payload Length, DPL）必须被使能。接收机（PRX）使用 W_ACK_PAYLOAD 命令打包数据，并送进 TX FIFO。ACK 包会一直在 TX FIFO 内，直到收到下一个来自发射机的数据包。在同一时刻，接收器可以有三个 ACK 包存在 TX FIFO 内。

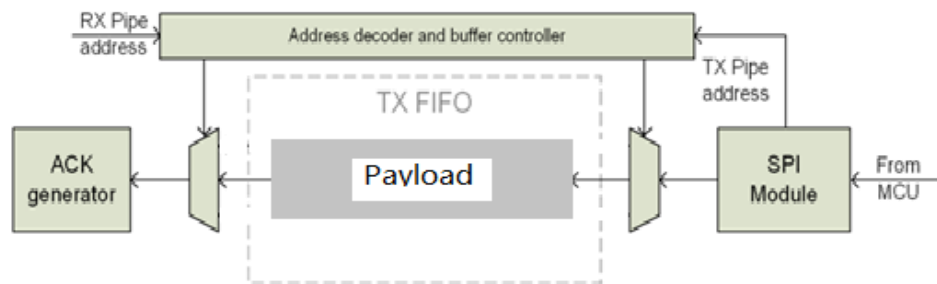


图 3-1: TX FIFO (PRX) 等待包

图 3-1 显示了接收机的 TX FIFO 是如何处理等待的 ACK 数据包的。MCU 通过 W_ACK_PAYLOAD 命令完成 ACK 包加载，地址译码器和缓冲控制器可以确保被通知 PTX 地址，这将确保送给 ACK 发生器的数据是正确的。

如果 PRX 的 TX FIFO 多于一个数据包发送给 PTX，则数据包的处理采用先进先出的原则。如果所有等待的有效数据包写给 PTX 链接丢失，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）。

配置 FEATURE 寄存器中的 EN_ACK_PAY 位才可使能自动应答功能。

4.4.2 自动重传（ART）

自动重传是指发射机(PTX)未收到 ACK 包,进行重新发送上一次的数据包。即被用在发射机(PTX)的自动应答系统。当发射机(PTX)没收到 ACK 包时,可以通过设置 SETUP_RETR 寄存器的 ARC

位来选择重传的次数。PTX 每次发送数据包后，都进入接收模式，等待 ACK 包的返回。PTX 在接收模式下，等待的时间由下面三个因素决定：

- 自动重传延时(ARD)
- 在 256 μ s 内无地址匹配
- 如在 256 μ s 内地址匹配，收到数据包(无论通过 CRC 与否)

当发射机(PTX)收到 ACK 时,会产生一个 TX_DS IRQ 中断。如果在 TXFIFO 内无待发射数据和 CE 被拉低，发射机(PTX)会进行 Standby-I 模式。发射机(PTX)在等待一段延时(ARD)后，会重新发射数据。这个过程一直会被持续，直到收到 ACK 包或者达到最大重传次数。OBSERVE_TX 寄存器的两个计数器 ARC_CNT 和 PLOS_CNT 在每次数据包丢失时，都会自动增加，ARC_CNT 计数是当前交互的重传次数，可以对 ARC_CNT 进行复位，开始新的数据交互。PLOS_CNT 计数是自上次信道变化后的总重传次数，PLOS_CNT 的复位在 RF_CH 寄存器内配置。可以使用 OBSERVE_TX 寄存器中的信道信息对信道总体质量进行评估。

ARD 是指发射包结束到重传开始之间的时间。ARD 的步进是 256 μ s，通过设置 SETUP_RETR 寄存器来改变。

- ARD 不应该小于建立时间和 ACK 包的传输时间的总和。
- 对于 2Mbps 速率，5 字节地址，最大 15 字节 ACK 包，ARD=256 μ s (复位值)。

对于 1Mbps 速率，5 字节地址，最大 5 字节 ACK 包，ARD=256 μ s(复位值)。ARD=512 μ s 时，对于在 2Mbps 或 1Mbps 下，任何长度的 ACK 都可以满足。

当使能自动重传功能时，需要设置收发机的数据包重传次数，通过 REUSE_TX_PL 命令实现。使用此命令时，MCU 每次都需要初始化发射数据包（CE 脉冲）。

4.5 协议引擎流程图

本节描述协议引擎中 PTX 和 PRX 的操作流程图。

4.5.1 PTX 操作

图 3-1流程图概述了进入 standby-I 模式后，射频收发器如何配置为发射机（PTX）。

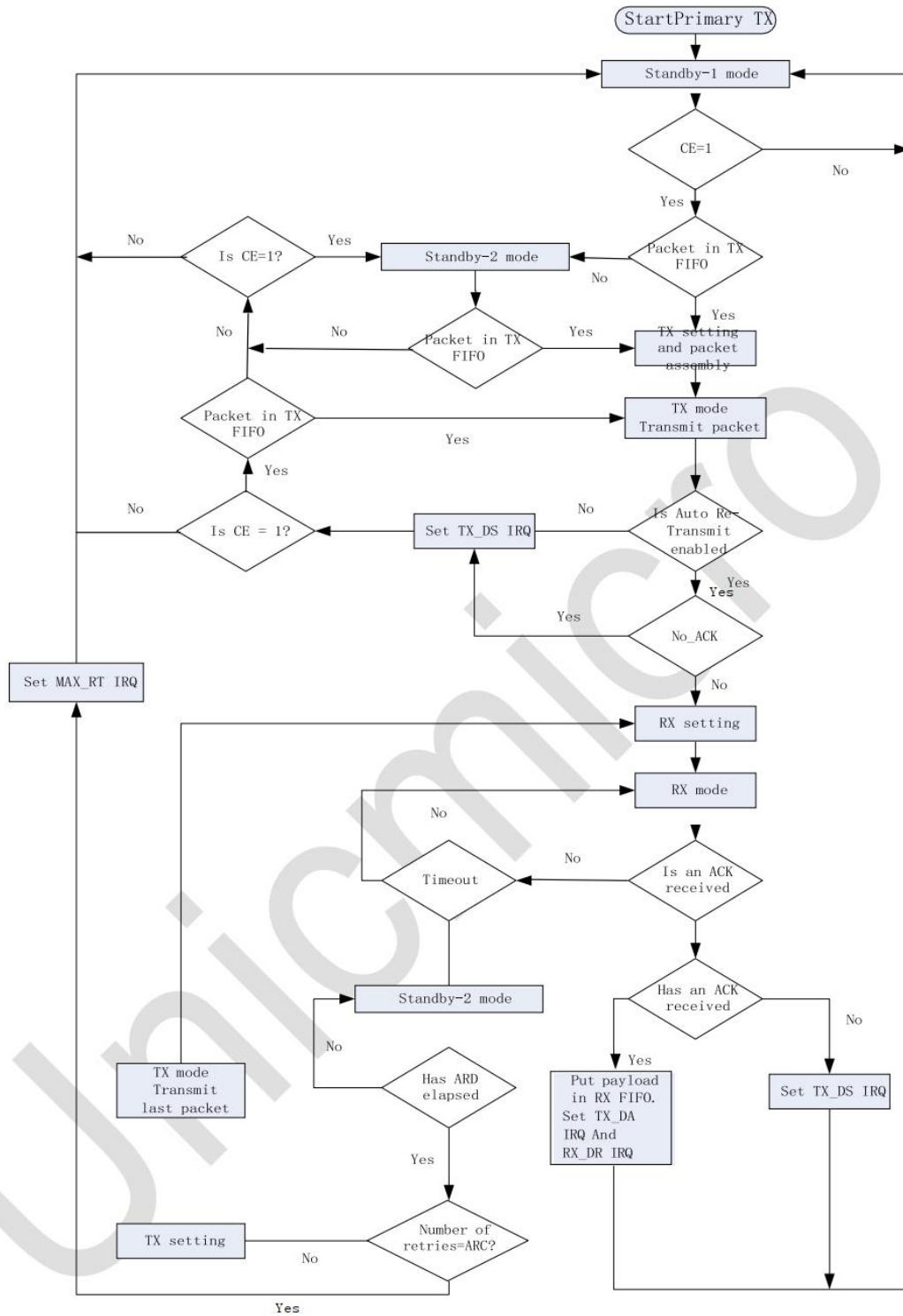


图 3-1: 协议引擎中 PTX 操作流程

通过设置 CE 为高激活 PTX 模式，如果 TX FIFO 内有数据包，RF 收发器进入 TX 模式并传输数据包。如果自动重传被使能，状态机检测是否设置了 NO_ACK 位，如果没有设置，RF 收发器进入 RX 模式接收 ACK 包。如果接收的 ACK 包为空，只产生 TX_DS 中断。如果 ACK 包包含数据，同时产生 TX_DS 和 RX_DR 中断，然后 RF 收发器进入 standby-1 模式。

如果计数超时而未收到 ACK 包, RF 收发器进入 Standby-II 模式。RF 收发器一直处于 Standby-II 模式直到 ARD 结束。如果重传次数未到 ARC, 则 RF 收发器进入 TX 模式并再次发射上一次数据包。

当使能了自动重传功能, 重传次数可以达到 ARC 中定义的最大次数。如果重传次数达到了最大次数, RF 收发器产生 MAX_RT 中断并返回到 Standby-I 模式。

如果 RFCON 寄存器的 CE 位为高且 TX FIFO 为空, 则 RF 收发器进入 Standby-II 模式。

4.5.2 PRX 操作

图 3-2流程图概述了进入 standby-I 模式后, 射频收发器如何配置为接收机 (PRX)。

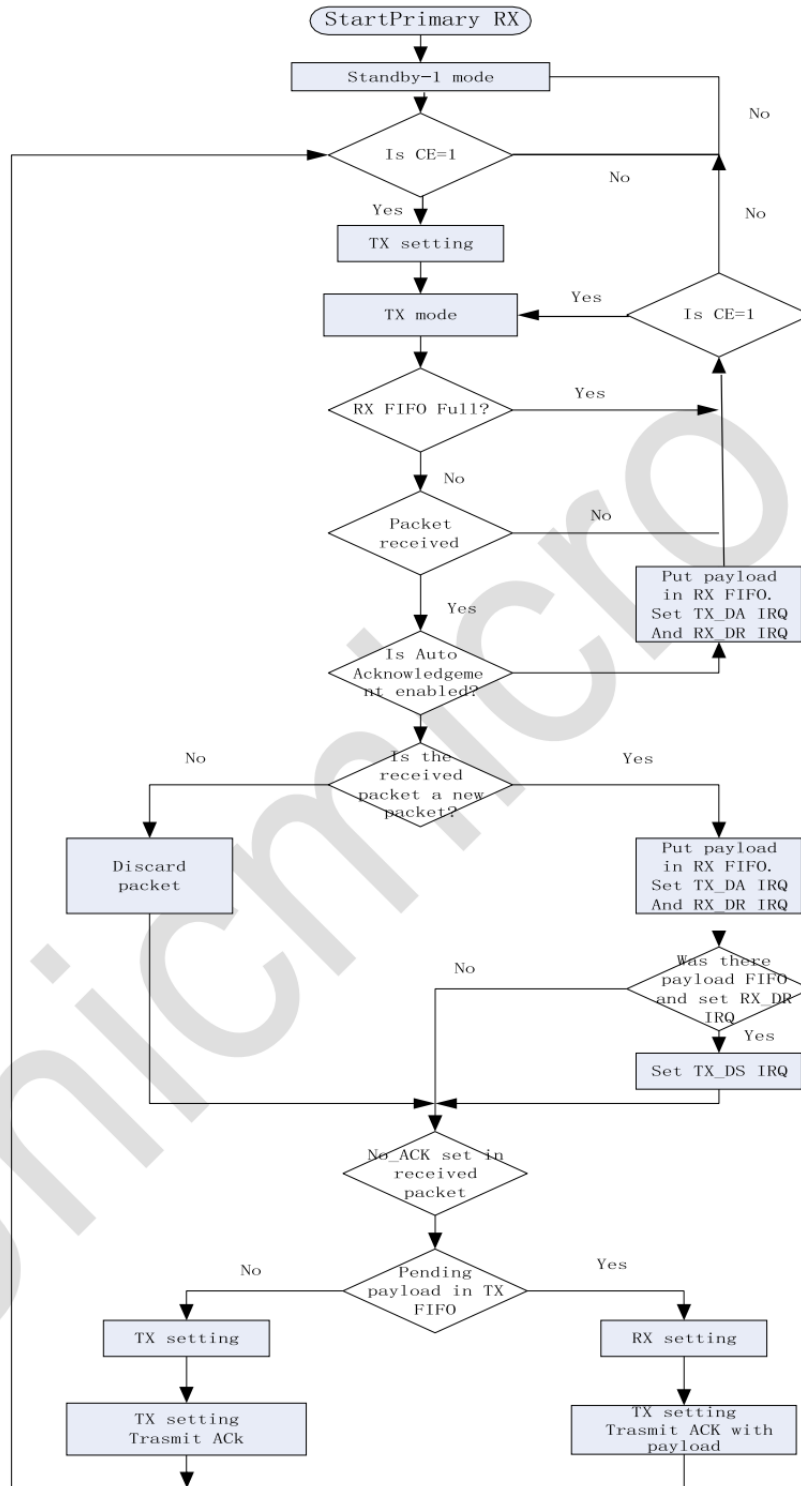


图 3-2: 协议引擎中的 PRX 操作流程图

通过设置 RFRCON 寄存器中的 CE 位为高激活 PRX 模式。RF 收发器进入 RX 模式并开始查询数据包。如果 RF 收发器收到了数据包并使能了自动应答模式，RF 收发器对数据包进行判断，是新的数据包还是之前收到的数据包。如果是新数据包，则送入 RX FIFO，并产生 RX_DR 中断。如果收到的是上一次数据包，并且是带有数据的 ACK 包，TX_DS 中断表示 PTX 收到带有数据的 ACK

包。如果收到的数据包未设置 NO_ACK 位，PRX 进入 TX 模式。如果 TX FIFO 中有待发送的带有数据的 ACK 包，ACK 数据包发射完之后，RF 收发器进入 RX 模式。

如果 ACK 包丢失，PRX 可能会收到重复的数据包。在这种情况下，PRX 丢掉重复的数据包，发射 ACK 数据包后切换到 RX 模式。

4.6 多从机

多从机用于 RX 模式，其中包含了三个地址独立的数据管道。一个数据管道在物理 RF 信道里有一个逻辑信道，每一个数据管道有自己在 RF 收发器中解码的物理地址（数据管道地址）。

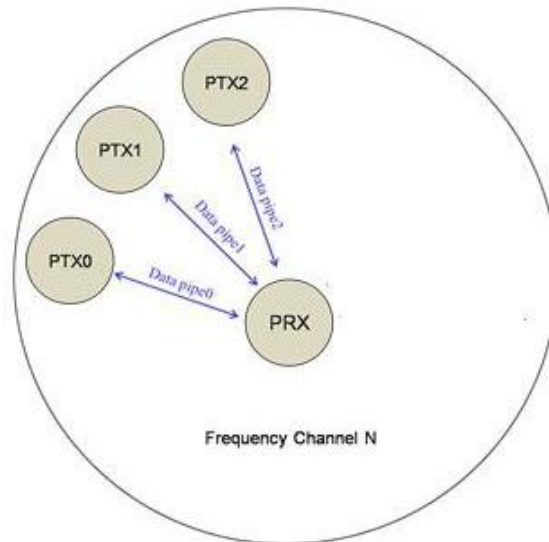


图4-3: PRX 用于多从机模式

RF 收发器配置成 PRX 模式，可以收到来自同一个频率信道的 3 个不同管道的数据地址。如图 4-3 所示，每一个数据管道都有自己唯一的地址，并且是相互独立的。

最多 3 个 PTX 和一个 PRX 通信。所有的数据管道地址都是被同时查寻，在同一时刻，只有一个数据管道能接收到数据包。所有数据管道都遵循协议引擎功能。

以下为所有数据管道常用的设置：

- CRC 使能/禁止 (当协议引擎使用时 CRC 被使能)
- CRC 编码方案
- RX 地址宽度
- 射频信道
- 据速率
- LNA 增益

通过 EN_RXADDR 寄存器使能数据管道，默认数据管道“0”和数据管道“1”被使能。每个数据管道的地址通过 RX_ADDR_PX 寄存器配置。

注意：始终确保每个数据管道都具有不同的地址。

每个数据管道地址最大为 5 字节数。数据管道 0~2 共享 4 个高地址字节，低字节是唯一的。图 4-2 描述了数据管道 0~2 地址是如何分配的，只有数据管道 0 有 5 字节地址可配，其他数据管道只有 1 字节地址可配。

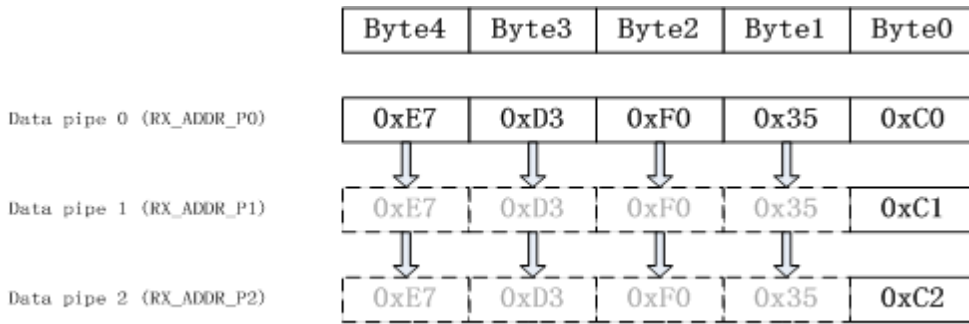


图 4-2: 数据管道 0-2 地址分配

PRX 用于多从机和通信引擎，能接收来自多于一个 PTX 发送的数据包，为了确保 PRX 的 ACK 包发送给正确的 PTX，当传输 ACK 包，PRX 将收到的数据包的管道地址作为发射地址。图 4-3 描述了 RTX 和 PRX 的地址分配，PRX 地址在 RX_ADDR_Px 寄存器中，作为数据管道地址，地址必须是唯一的。PTX 地址在 TX_ADDR 寄存器中，必须与 RX_ADDR_P0 相同，

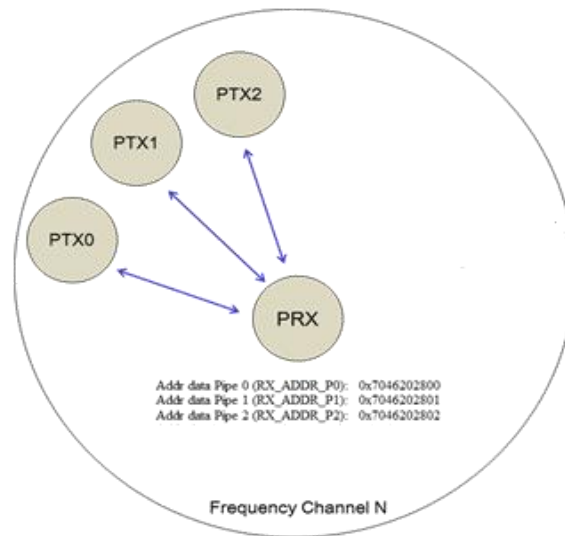


图 4-3: 多从机地址分配

只有当一个数据管道接收完数据包后，其他数据管道才能开始接收数据。
 当多个 PTXs 与 PRX 通信时，ARD 用于规划先后发射次序，防止相互碰撞。

4.7 协议引擎时序

本节描述协议引擎的时序及所有模式的启动和操作。通过控制数据和接口控制协议引擎的时序，RF 收发器可设置成静态模式和自动模式（内部状态机控制事件）。每个自动模式/时序结束产生 RFIRQ 中断，在时序图中，所有的中断都是 IRQ 事件。

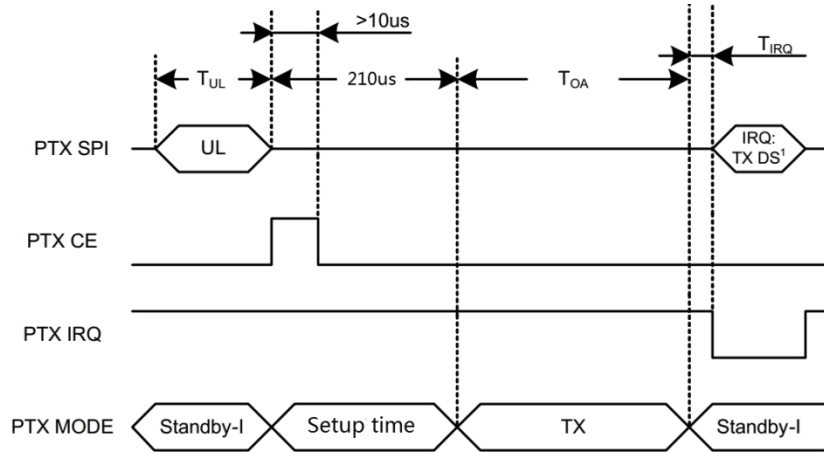


图 4-4: NO_ACK 使能的发射时序图

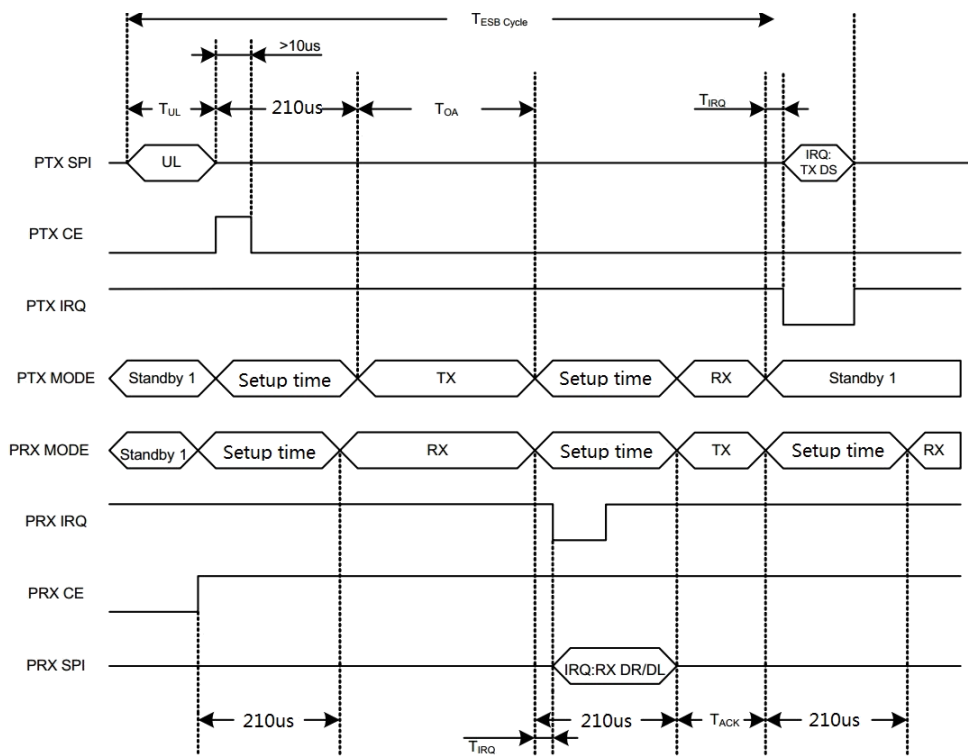


图 4-5: 协议引擎上传数据包的时序图 (2Mbps)

图 4-5描述了数据包的传输和应答。PRX 工作在接收模式（CE=“1”），PTX 工作在发射模式（CE=“1”最少 20μs），在 210μs 后，传输开始并在 TOA 时间后结束。

当 PTX 完成发射后，自动切到 RX 模式等待 PRX 回 ACK 包，PRX 收到数据包后产生中断，告知 MCU，切换到 TX 模式发射 ACK 包。PTX 收到 ACK 包后产生中断，告知 MCU，清空 TX FIFO。

第一个 ACK 未收到的协议引擎时序图如图 4-6所示：

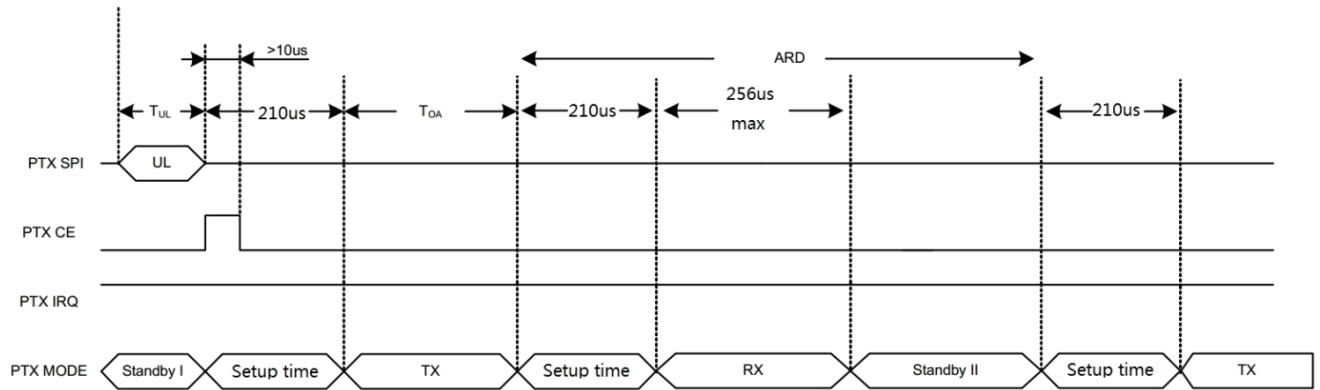


图 4-6：第一个 ACK 未收到的协议引擎时序图(2Mbps)

4.8 协议引擎交互框图

本节描述了协议引擎自动交互的几个场景，本节框图表示 IRQ 及其他事件，MCU 在不同时间处理的事件。

4.8.1 带 ACK 包和中断事件的单一数据包处理

图 4-7显示了最基本的自动应答模式。PTX 发射数据包后，接收来自 PRX 的 ACK 包。在 PRX 接收数据包后产生 RX_DR 中断，当 PTX 收到 ACK 包后产生 TX_DS 中断。

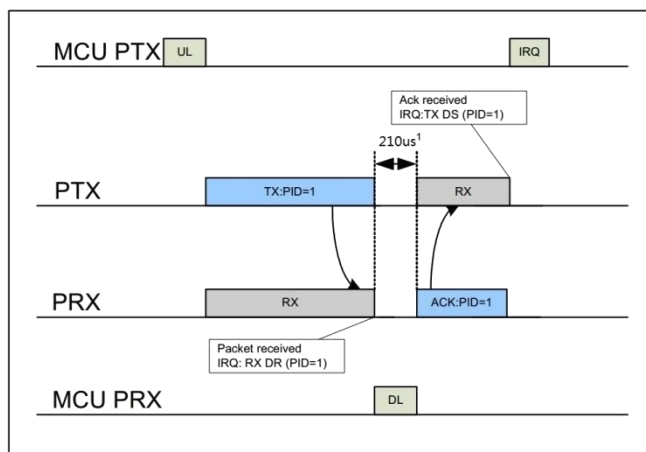


图 4-7：带 ACK 和中断的 TX/RX 时序图

4.8.2 未收到数据包的单一数据包处理

图 4-8显示了丢失第一个数据包时，重传的场景。在 PTX 发射数据包后，进入 RX 模式接收 ACK 包。PTX 第一次发射之后，等待一段时间（包含建立时间，RX_TM 和 ARD 时间），如果在此时间段内未收到 PTX 重传的数据包，当数据包发射时，在 210μs 的建立时间后，PTX 切到 RX 模式。在 130μs 的接收时间超时后（RX_TM 的超时时间，可以被设置更短的时间），PTX 切到 ARD（ARD 可以设置为 0μs、256μs、512μs 到 3840μs）

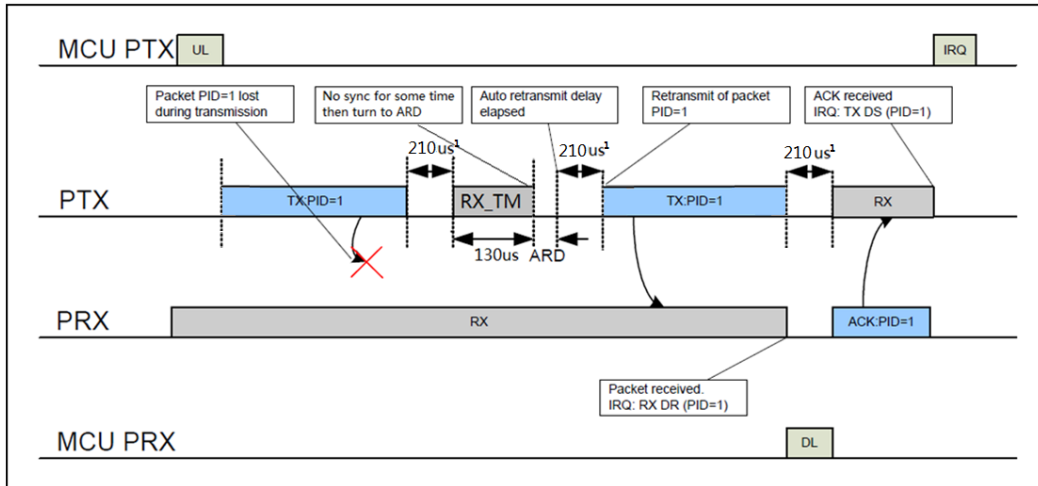


图 4-8：当第一个包发送失败时，带 ACK 和中断的 TX/RX 时序图

4.8.3 未收到 ACK 包的单一数据包处理

图 4-9显示了 ACK 包丢失后重传的场景，并显示了相关的中断。

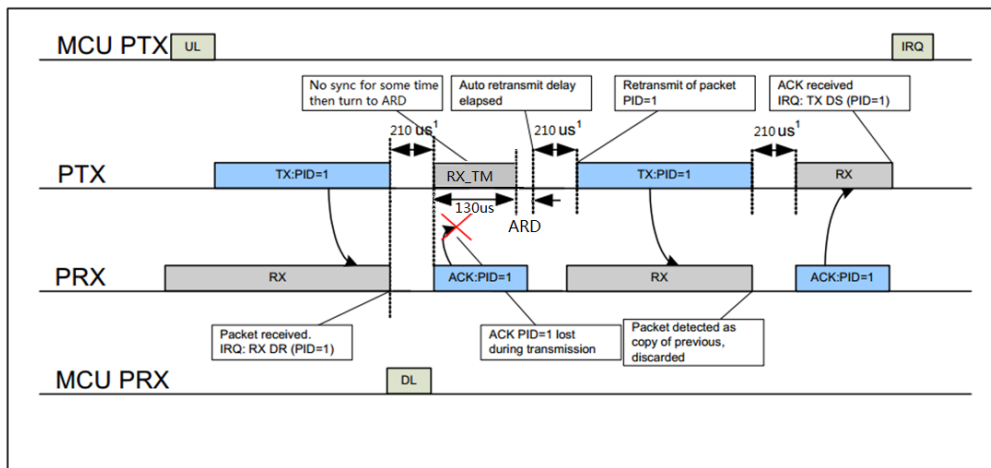


图 4-9：T 未收到 ACK 包时，带 ACK 和中断的 TX/RX 时序图

4.8.4 ACK 数据包的单一数据包处理

图 4-10显示最基本的带有数据的自动应答场景。PTX 发射完数据包后，接收来自 PRX 的 ACK 数据包。在 PRX 接收数据之后产生 RX_DR 中断，当 PTX 收到 ACK 包时，产生 TX_DS 中断。当 PRX 收到来自 PTX 的 ACK 数据包后，产生 TX_DS 中断。

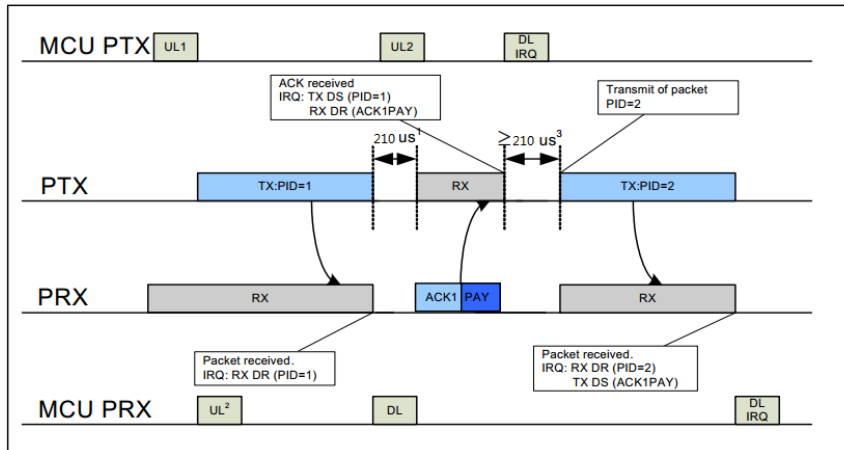


图 4-10: ACK 数据包的 TX/RX 时序图

4.8.5 ACK 数据包的单一数据包处理和数据包丢失

图 4-11在 PRX RX_DR 中断产生之前，第一个数据包丢失和重传的场景。在 PTX 收到 ACK 包后，产生 TX_DS 和 RX_DR 中断。在 PRX 收到第二个数据包 (PID=2) 后，产生 RX_DR (PID=2) 和 TX_DS (ACK 数据包)中断。

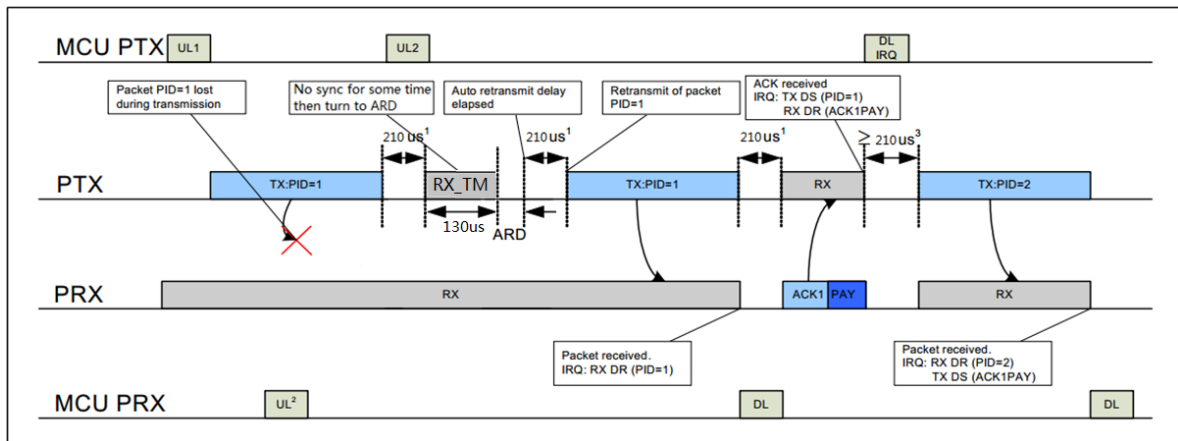


图 4-11: 未收到数据包, TX/RX 时序图和相关的中断

4.8.6 ACK 数据包的双数据包处理及第一个 ACK 包丢失

图 4-12描述了在 TX_DS 中断产生前，ACK 包丢失和重传的情况下，产生了 RX_DR 中断。PTX 重发上次数据包 (PID=1)，在 PTX 发射第二个 ACK 包时，产生 TX_DS 和 RX_DR 中断。在 PRX 收到第二个数据包 (PID=2) 后，产生 RX_DR (PID=2) 和 TX_DS (ACK1PAY)中断。

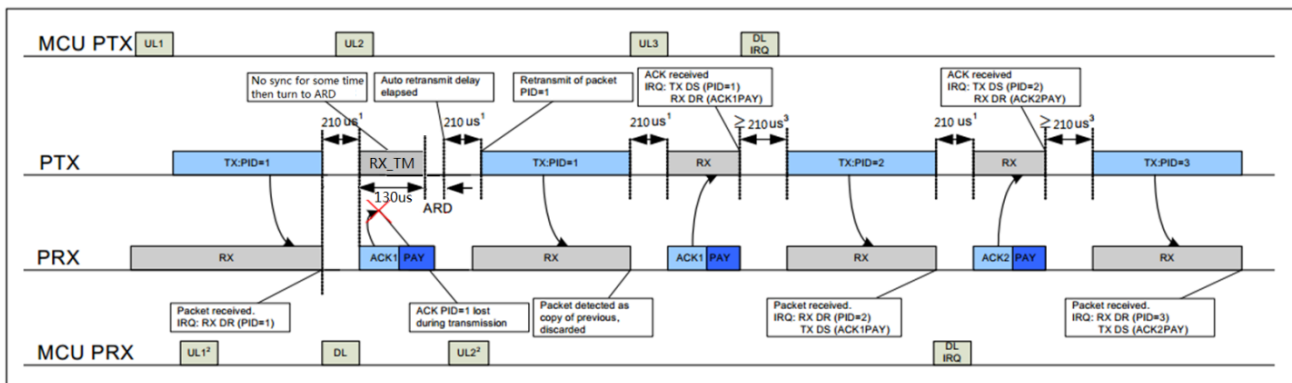


图 4-12: 未收到 ACK 数据包，携带 ACK 数据的 TX/RX 时序图和相关中断

4.8.7 最大重传下的双数据包处理

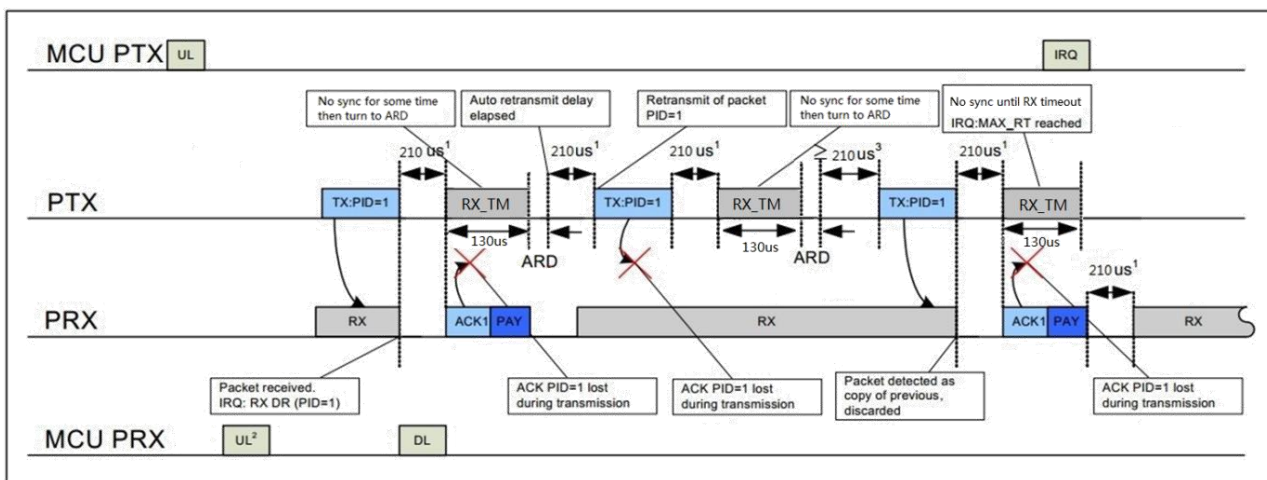


图 4-13: 产生最大重传的 TX/RX 时序图

ARC 计数器达到设置的最大值时，产生 MAX_RT 中断，如图 4-13，MAX_RT 中断产生时，数据包传输终止。TX FIFO 中的数据不会被清除，MCU 执行下一个步骤时，RFCON 寄存器中 CE 变换，重新传输数据包，可通过 FLUSH_TX 命令清 TX FIFO。

5 数据和控制接口

数据和控制接口使您可以访问 RF 收发器的所有功能。

耐压 5V，4 线 SPI 信号如下：

- IRQ（低电平有效，由三个屏蔽的中断源控制）
- CSN（4 线 SPI 信号）
- SCK（4 线 SPI 信号）
- MOSI（4 线 SPI 信号）
- MISO（4 线 SPI 信号）

耐压 5V，3 线 SPI 信号如下：

- IRQ（低电平有效，由三个屏蔽的中断源控制）
- CSN（3 线 SPI 信号）
- SCK（3 线 SPI 信号）
- SDA（3 线 SPI 信号）

5.1 特征

- 特殊的 SPI 指令，适合快速频繁的使用
- 0~18Mbps，3/4 线 SPI
- 8bit 指令集
- 简单的寄存器配置映射

5.2 功能描述

SPI 是一个最大数据率为 10Mbps 的标准 SPI。

5.3 SPI 操作

本节介绍 SPI 命令和时序。

5.3.1 SPI 命令

SPI 命令如[错误!未找到引用源。](#)所示，每一条命令的执行必须开始于一次 CSN 由高到低的变化。

SPI 命令格式如下：

<命令字：由高位到低位(每字节)>

<数据字节：低字节到高字节，每一字节高位在前>

表 5-1：SPI 命令

命令名称	命令字 (二进制)	数据字节	操作
R_RESISTER	000A AAAA	1 to 5低字节在前	读命令寄存器和状态寄存器。 AAAAA=5 bit 寄存器地址
W_RESISTER	001A AAAA	1 to 5低字节在前	写命令寄存器和状态寄存器。 AAAAA=5 bit 寄存器地址, 仅在power down和standby 模式下可执行。
R_TX_PAYLOAD	0110 0001	1 to 32低字节在前	读接收数据：1-32 字节。 读操作通常由0字节开始，读完后数据将从FIFO中删除，在接收模式中使用。
W_TX_PAYLOAD	1010 0000	1 to 32低字节在前	写发射数据：1 – 32 字节。 写操作通常由0字节开始，在发射模式中使用。
FLUSH_TX	1110 0001	0	清 TX FIFO，TX模式中使用。
FLUSH_RX	1110 0010	0	清RX FIFO，RX 模式中使用。 在传输应答时不执行，否则会使应答数据不完整。
REUSE_TX_PL	1110 0011	0	用于 PTX 操作.重用最后传送的数据. TX 数据在W_TX_PAYLOAD或 FLUSH TX 命令被执行后可用。在数据传输时不能执行此命令。
R_RX_PL_WID	0110 0000	1	读RX FIFO 最顶部R_RX_PAYLOAD 的数据宽度。 注意：如果读的值大于32字节，清除RX FIFO。
W_ACK_PAYLOAD	1010 1PPP	1 to 32 低字节在前	在RX模式中使用。写PIPE PPP（PPP的值从000到101）响应ACK时同时回传的数据。最多可设置3个ACK数据包。相同PPP的数据将以先进先出的原则发送。写操作通常从0字节开始。
W_TX_PAYLOAD_NO_ACK	1011 0000	1 to 32低字节在前	在TX模式中使用。 使用该命令发送数据将使AUTOACK不可用。
CE_HIGH	1101 0101	0	CE为高命令
CE_LOW	1101 0110	0	CE 为低命令

W_REGISTER 和 R_REGISTER 命令可操作在单字节或多字节寄存器。当访问多字节寄存器时，首先读或写的是低字节的高位。在多字节寄存器中的所有字节被写完之前可以结束写操作，使得未写完的高字节内容保持不变。例如，可以通过写一个字节修改 RX_ADDR_P0 寄存器低字节内容。在 CSN 状态由高变低后可以通过 MISO 来读取状态寄存器的内容。

注意：在 IRQ 由高到低的转换期间，STATUS 寄存器中的 2 位管道信息会被更新，在此期间读取的 STATUS 寄存器的管道信息内容是不可靠的。

5.3.2 SPI 时序

SPI 操作和时序如图 5-1~图 5-4所示。在写配置寄存器之前，UM2052 必须处于 Standby、Sleep 或 Power down 模式。

表 5-2：图 5-1~图 5-4使用的缩写名称表

缩写名称	描述
Cn	SPI 命令位
Dn	数据位(说明:低字节 到高字节, 每字节高位优先)

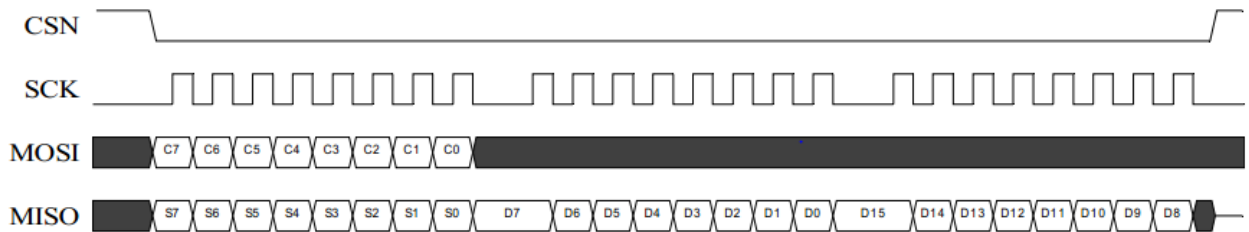


图5-1：4 线 SPI 读操作

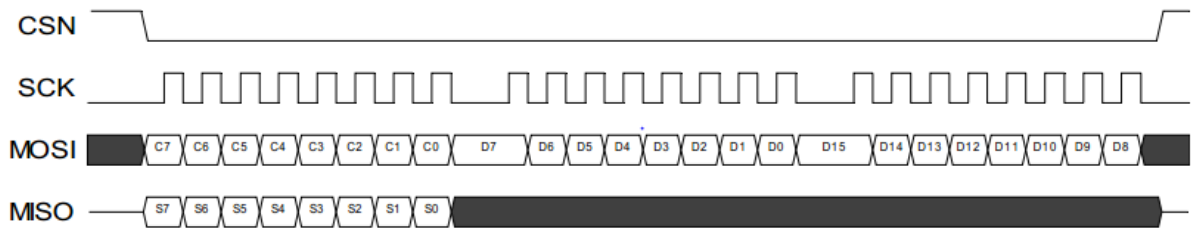


图5-2：4 线 SPI 写操作

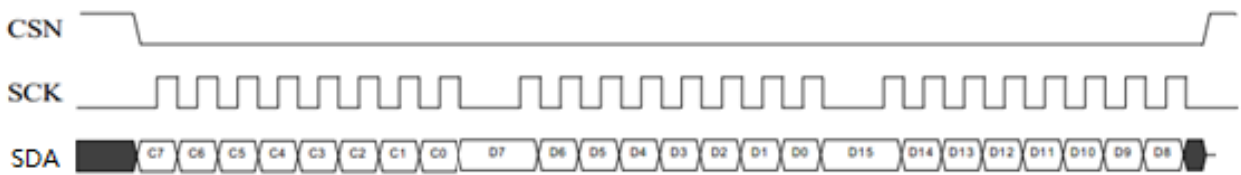


图5-3：3 线 SPI 读操作

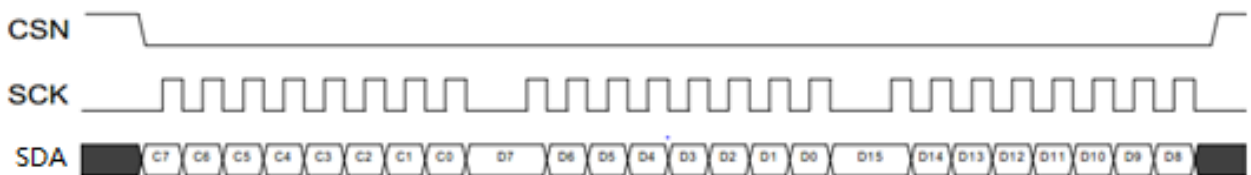


图5-4：3 线 SPI 写操作

5.4 数据 FIFO

数据 FIFO 存储传输的数据在 TX FIFO，或者接收时钟输出的数据到 RX FIFO。FIFO 支持 PTX 模式和 PRX 模式。RF 收发器存在以下 FIFO：

- TX，32 字节 FIFO
- RX，32 字节 FIFO

两个 FIFO 都有一个控制器，可以通过专用的 SPI 命令访问。在 PRX 中的 TX FIFO 将存储的 ACK 数据包发送给不同的 PTX 设备，如果 TX FIFO 包含了一个数据管道，数据的处理遵循先进先出的原则。如果所有等待的有效数据包写给 PTX 链接丢失，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)。

PRX 中的 RX FIFO 可以包含来自最多一个 PTX 设备的数据，PTX 中的 TX FIFO 可以存储最多一个数据。您可以使用以下 3 个命令写入到 TX FIFO：PTX 中的 W_TX_PAYLOAD、W_TX_PAYLOAD_NO_ACK 命令和 PRX 中的 W_ACK_PAYLOAD 命令。使用这三个命令可以对 TX_PLD 寄存器进行访问。

在 PTX 和 PRX 中，可以通过 R_RX_PAYLOAD 命令读取 RX FIFO 中的内容，通过该命令可以访问 RX_PLD 寄存器。

如果 MAX_RT 中断产生，不会移除在 PTX TX FIFO 中的数据。

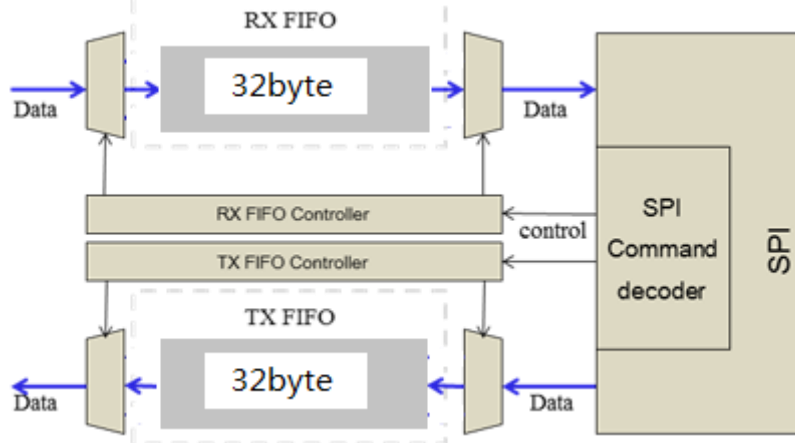


图5-5: FIFO (RX 和 TX)功能框图

用户可读出 FIFO_STATUS 寄存器中 TX 和 RX 是满或空的状态。

5.5 中断

UM2052 的中断引脚 (IRQ) 为低电平触发，当状态寄存器中的 TX_DS、RX_DR 或 MAX_RT 为高时触发中断，当 MCU 给 STATUS 寄存器中的中断源写“1”时，IRQ 引脚复位。CONFIG 寄存器中的 IRQ 掩码用于选择中断源或者触发中断。通过设置 MASK 位为高，中断响应被禁止，默认状态下所有中断源是被使能的。

注意：在 IRQ 由高到低的转换期间，STATUS 寄存器中的 2bit 管道信息会被更新，在此期间读取的 STATUS 寄存器的管道信息内容是不可靠的。

6 寄存器映射

您可以通过 SPI 访问寄存器来配置和控制 UM2052。寄存器映射信息请参见《UM2052 2.4G Register Map》。

7 电气参数

7.1 绝对最大额定值

外部条件如果超过“绝对最大额定值”列表中给出的值，可能会导致器件永久性地损坏。这里只是给出能承受永久性损坏的最大载荷，并不意味着在此条件下器件的功能性操作无误。器件长期工作在最大值条件下会影响器件的可靠性。

表 7-1：芯片绝对最大额定值

操作条件	最小值	最大值	单位
供电电压			
VDD	-	3.6	V
VSS	-	0	V
输入电压			
V_i	-0.3	5	V
输出电压			
V_o	VSS to VDD	VSS to VDD	V
温度			
操作温度	-40	85	°C
储存温度	-40	125	°C

7.2 运行条件

表 7-2：运行条件参数

符号	参数（条件）	备注	最小值	典型值	最大值	单位
VDD	供电电压	-	1.9	3.0	3.6	V
VDD	供电电压（输入信号>3.6V）	-	1.9	3.0	3.6	V
TEMP	运行温度	-	-40	27	85	°C

7.3 电流功耗

表 7-3：电流功耗参数

参数	符号	条件	最小值	典型值	最大值	单位
Power Down 电流	I_{pd}	寄存器保持	-	2	-	μA
Sleep 电流	I_{slp}	晶体工作	-	50	-	μA
发射电流@0dBm	I_{tx}	PA 0dBm	-	17	-	mA
接收电流@2Mbps	I_{rx}	RX Mode	-	18	-	mA

7.4 频率综合器

表 7-4: 频率综合器参数

参数	符号	条件	最小值	典型值	最大值	单位
工作频率	F_{req}	-	2400	-	2525	MHz
PLL 频率步进	Delta F	-	-	1	-	MHz
频偏@1Mbps	Df	-	-	250	-	KHz
频偏@2Mbps	Df	-	-	250	-	KHz

7.5 接收机

表 7-5: 接收机参数

参数	符号	条件	最小值	典型值	最大值	单位
最大接收信号强度	Pin, max	<0.1% BER	-	-10	-	dBm
灵敏度@1Mbps	-	<0.1% BER	-	-90	-	dBm
灵敏度@2Mbps	-	<0.1% BER	-	-87	-	dBm

7.6 发射机

表 7-6: 发射机参数

参数	符号	条件	最小值	典型值	最大值	单位
最大输出功率	P_{max}	50ohm antenna	-	8	-	dBm
最小输出功率	P_{min}	50ohm antenna	-	-34	-	dBm
射频功率可控范围	P_{range}	50ohm antenna	-	42	-	dBm

7.7 晶体

表 7-7: 晶体参数

参数	符号	条件	最小值	典型值	最大值	单位
晶体频率	F_{xtal}	-	16	16	16	MHz
晶体频率容差	Df_{xtal}	-	-60	-	60	ppm
负载电容	C_{xtal}	-	-	12	-	pF

7.8 直流特性

表 7-8: 直流特性

参数	符号	条件	最小值	典型值	最大值	单位
高电平输入	V_{high}	-	0.7VDD	-	3.6	V
低电平输入	V_{low}	-	0	-	0.3VDD	V

8 典型应用原理图

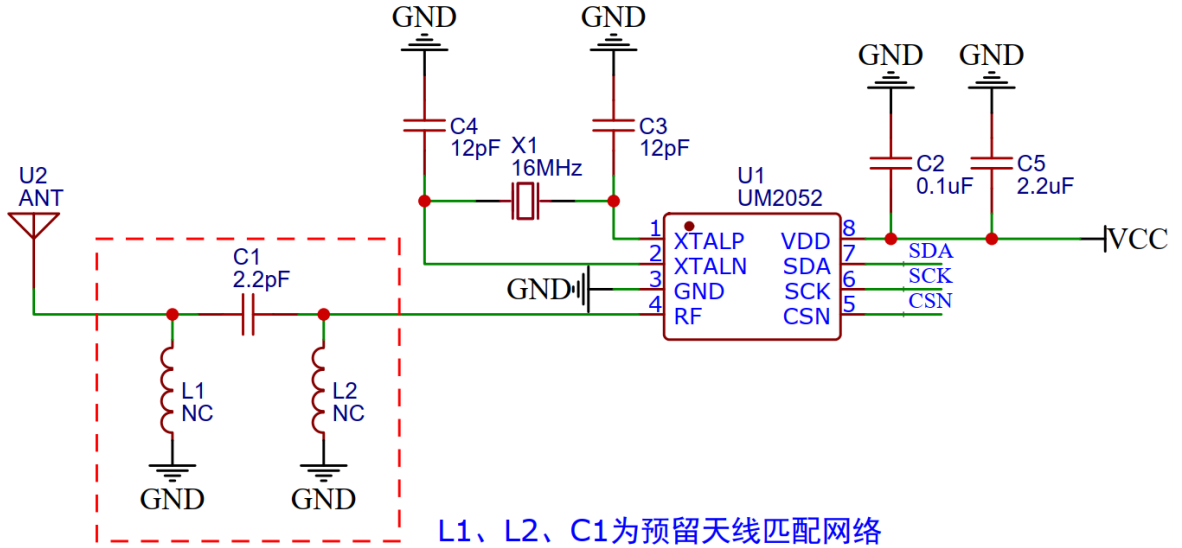


图 8-1：SOP8 封装应用原理图

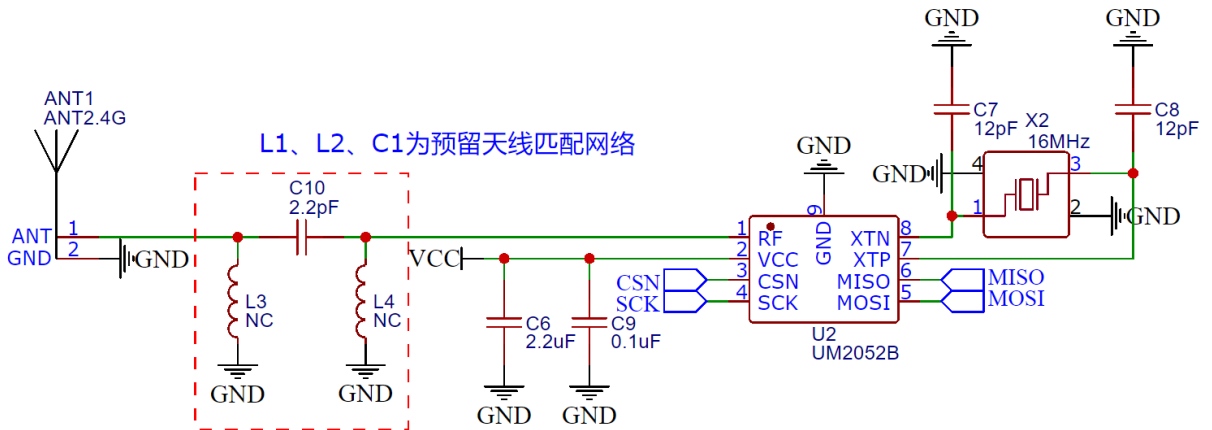


图 8-2：ESOP8 封装应用原理图

注意：

L1、L2、C1 为预留天线匹配网络，如果天线已进行匹配则 L1、L2 可以省略，C1 用导线连接；
C3、C4 晶振外接负载电容，可以省略。

9 封装尺寸

UM2052 采用 SOP8 封装 (4.9*3.9mm) 和 ESOP8 封装 (4.9*3.9mm), 区别在于 IC 底部的 PAD。

9.1 SOP8 (4.9*3.9mm)

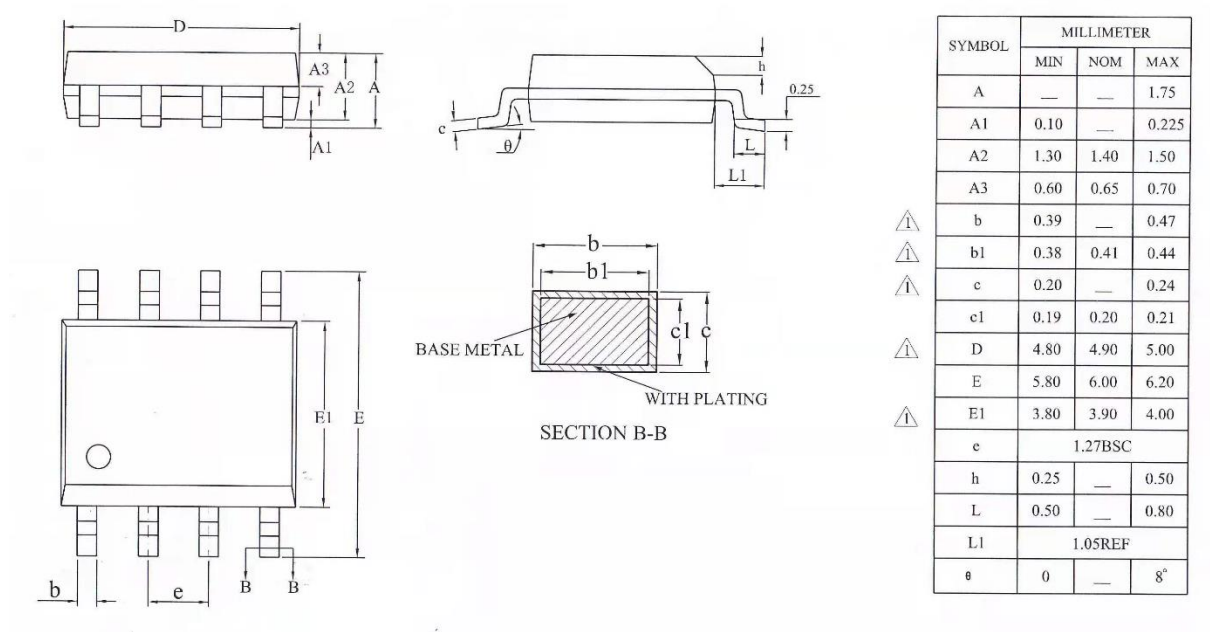
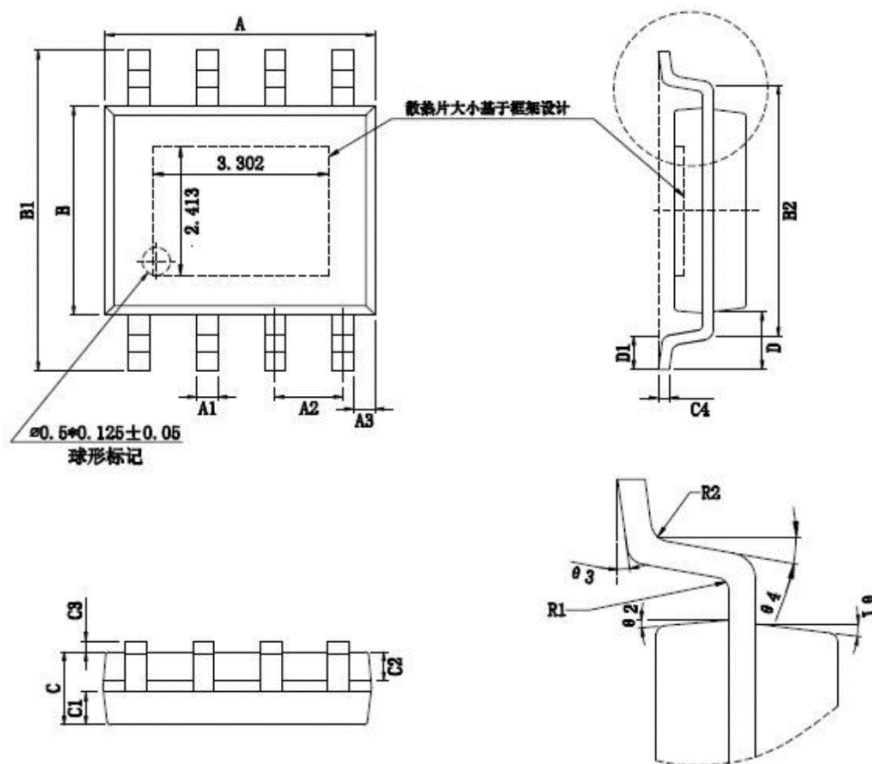


图 9-1: SOP8 封装图

9.2 ESOP8 (4.9*3.9mm)



标注	尺寸	最小 (mm)	最大 (mm)	标注	尺寸	最小 (mm)	最大 (mm)
A		4.80	5.00	C3		0.05	0.20
A1		0.356	0.456	C4		0.203TYP	
A2		1.27TYP		D		1.05TYP	
A3		0.345TYP		D1		0.40	0.60
B		3.80	4.00	R1		0.20TYP	
B1		5.80	6.20	R2		0.20TYP	
B2		5.00TYP		$\theta 1$		17° TYP4	
C		1.30	1.50	$\theta 2$		13° TYP4	
C1		0.55	0.65	$\theta 3$		0° ~ 8°	
C2		0.55	0.65	$\theta 4$		4° ~ 12°	

图 9-2: ESOP8 封装图