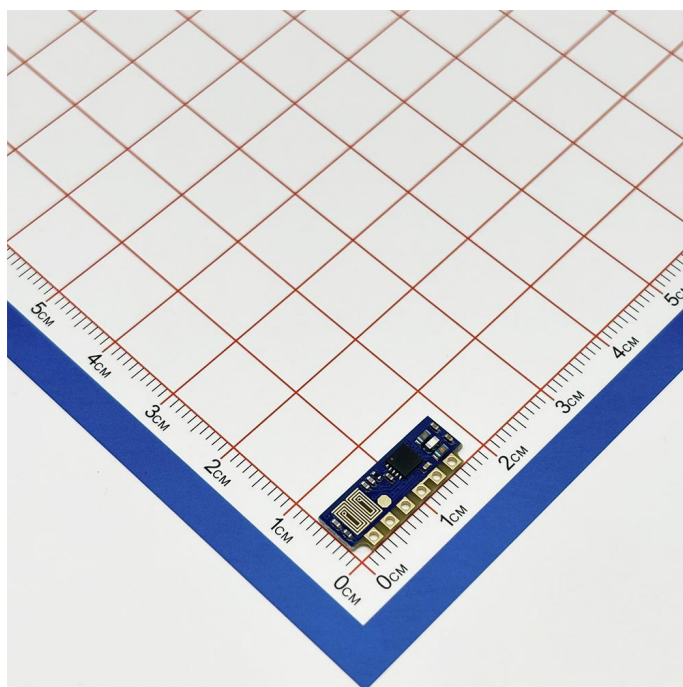




微-R6 低功耗接收模块规格书

Wei-R6 Low Power Receiver Module Specification



项目	内容
产品型号	微-R6 / Wei-R6
文档版本	V1.1
发布日期	2026-07-04
品牌	FNR / 蜂鸟无线



目录

微-R6 低功耗接收模块规格书1

1 产品概述 3

2 产品特点 3

3 典型应用 3

4 技术参数 4

5 引脚定义 5

6 输出接口与休眠控制 5

7 用法与接线 6

8 MCU 接收与例程建议 7

9 电源设计建议 9

10 EMC 与抗干扰建议 10

11 PCB 布线建议 10

12 天线与结构建议 10

13 回流焊与生产建议 10

14 调试流程建议 11

15 尺寸 12

16 资料与技术支持 12

附录 A 天线、封装、开发工具、服务、购买 13



1 产品概述

微-R6 是一款微型低功耗 ASK/OOK 超外差接收模块, 支持 315MHz 或 433MHz 频点。与上一代产品相比, 模块尺寸进一步减小, 适合空间受限的低功耗遥控接收端。

模块采用 RF 接收集成芯片和 Q 介质谐振稳频技术, 提供 S 休眠控制、D 数据输出、V/G 电源引脚和 A 天线接口。D 输出为解调后的数字脉冲信号, 可连接 MCU 或解码 IC。

2 产品特点

- 2.1-5.5V 宽电压供电, 典型 3.3V。
- 工作电流典型 0.18mA。
- 休眠电流典型 0.008mA, 唤醒时间约 1ms。
- 315MHz / 433MHz 可选。
- ASK/OOK 调制方式, 最大传输速率 4.8kbps。
- ESD 指标 2000V。
- 7.00 x 18.00 x 1.95mm 小尺寸封装。

3 典型应用

- 单火线开关
- 电池供电遥控接收端
- 小尺寸无线门铃
- 智能家居控制
- 安防报警
- 无线灯控
- 小家电遥控

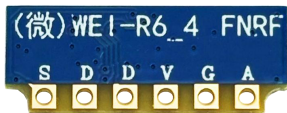


4 技术参数

参数	最小值	典型值	最大值	单位	条件/说明
工作频率	-	315 / 433	-	MHz	订购选择。
工作电压	2.1	3.3	5.5	V	VCC 与 GND 之间。
工作电流	-	0.18	-	mA	接收工作状态。
休眠电流	-	0.008	-	mA	休眠状态。
唤醒时间	-	1	-	ms	S 引脚唤醒。
调制方式	-	ASK/OOK	-	-	遥控接收。
接收灵敏度	-	-100	-	dBm	典型值。
传输速率	-	-	4.8	kbps	最大值。
接收带宽	-	2.5	-	MHz	典型接收带宽。
天线阻抗	-	50	-	Ω	A 引脚外接天线。
ESD	-	2000	-	V	静电防护指标。
外形尺寸	-	7.00 x 18.00 x 1.95	-	mm	宽 x 长 x 厚。
工作温度	-35	-	85	摄氏度	应用需按整机环境验证。



5 引脚定义



背面视图

引脚名称	类型	功能	说明
S	输入	休眠控制	内部下拉；悬空或拉低工作，拉高休眠，唤醒时间约 1ms。
D	输出	数据输出	解调后的数字脉冲输出端。
D	输出	数据输出	与上一 D 同功能，接其中一个即可。
V	电源	电源正	2.1~5.5V，典型 3.3V。
G	电源	电源负	系统地。
A	射频	天线端	50Ω 天线接口。

6 输出接口与休眠控制

DAT/D 输出为 ASK/OOK 解调后的数字脉冲信号，不是 UART 串口数据。MCU 或解码 IC 需要根据发射端编码协议解析高低电平宽度，例如 1527、2262 等常见遥控编码。

S 引脚为休眠控制端，内部下拉，悬空或拉低时模块处于工作状态；拉高时模块进入休眠状态。唤醒时间约 1ms。

项目	说明
数据输出类型	解调后的数字脉冲信号。
连接对象	MCU 输入脚或解码 IC 输入脚。
解析方式	按发射端编码协议解析高低电平宽度。
休眠控制	休眠控制脚拉高进入休眠，悬空或拉低工作。
接收边界	MCU 可通过边沿中断记录脉宽，并通过超时判断一帧结束。



7 用法与接线

方案一



方案二



方案三



7.1 接线

模块引脚	连接对象	说明
VCC/V	电源正极	按技术参数供电范围接入。
GND/G	电源负极	与 MCU 或解码 IC 共地。
DAT/D	MCU 输入或解码 IC 输入	接收解调后的数字脉冲信号。
SHUT/S	MCU GPIO 或固定电平	悬空或拉低工作，拉高休眠。
ANT/A	对应频段天线	天线走线短而直接，远离噪声源。



7.2 使用步骤

步骤	操作	说明
1	确认频点	发射端和接收端频点需一致。
2	接入电源	检查电源电压、纹波和共地状态。
3	设置休眠控制	需要接收时保持工作状态; 待机时可拉高进入休眠。
4	连接数据输出	数据输出脚接 MCU 输入脚或解码 IC 输入脚。
5	验证波形	用示波器或逻辑分析仪确认输出波形幅度和脉宽。
6	联调编码	按发射端编码协议解析地址码和键值。

8 MCU 接收与例程建议

数据输出脚输出的是脉冲信号, MCU 通常使用 GPIO 边沿中断记录高低电平持续时间, 再通过接收超时判断一帧结束。下面例程只展示接收框架, 具体协议解析需按发射端编码格式实现。

8.1 脉宽采样例程框架

```
// 微-R6 数据输出接收示例框架
// 思路: 模块处于工作状态时, DAT/D 引脚输出解调后的脉冲信号。
// MCU 在任意边沿中断中记录脉宽, 一段时间无新边沿后再解析一帧。
// 注意: 不同发射端编码的同步头、0/1 脉宽和重复帧规则不同,
// RX_DecodeFrame() 需要按实际协议单独实现。
#include <stdint.h>
#include <stddef.h>
#define RX_MAX_PULSE_COUNT 96
typedef struct {
    uint16_t level;    // 当前脉冲电平: 0 表示低电平, 1 表示高电平。
    uint16_t width_us; // 当前电平持续时间, 单位为微秒。
} RX_Pulse;
static RX_Pulse rx_pulses[RX_MAX_PULSE_COUNT];
static size_t rx_pulse_count = 0;
static uint32_t rx_last_edge_us = 0;
// 由平台定时器提供当前时间, 单位为微秒。
extern uint32_t Board_GetTimeUs(void);
```



```
// 控制休眠引脚: 0 为工作, 1 为休眠。
extern void Board_SetRxSleepPin(uint8_t level);

// 重新启动接收超时定时器, 例如 10-20ms 无新边沿后触发超时回调。
extern void Board_RestartRxTimeoutTimer(uint16_t timeout_ms);

// 按实际编码协议解析一帧脉冲, 例如 1527、2262 或自定义协议。
extern void RX_DecodeFrame(const RX_Pulse *pulses, size_t count);

// 接收前调用: 拉低休眠控制脚, 使模块进入工作状态。
void RX_StartReceive(void)
{
    Board_SetRxSleepPin(0);
    rx_pulse_count = 0;
    rx_last_edge_us = 0;
}

// 待机时调用: 拉高休眠控制脚, 使模块进入休眠状态。
void RX_StopReceive(void)
{
    Board_SetRxSleepPin(1);
    rx_pulse_count = 0;
    rx_last_edge_us = 0;
}

// 数据输出脚任意边沿中断回调。
// level_after_edge 为边沿后的电平, 用于推算上一段脉冲电平。
void RX_OnDataEdge(uint16_t level_after_edge)
{
    uint32_t now_us = Board_GetTimeUs();
    uint16_t width_us = (uint16_t)(now_us - rx_last_edge_us);
    // 第一次进入时 rx_last_edge_us 可能为 0, 可从第二个边沿开始记录有效脉宽。
    if (rx_last_edge_us != 0 && rx_pulse_count < RX_MAX_PULSE_COUNT) {
        rx_pulses[rx_pulse_count].level = (uint16_t)(!level_after_edge);
        rx_pulses[rx_pulse_count].width_us = width_us;
        rx_pulse_count++;
    }
    rx_last_edge_us = now_us;
    // 每来一个新边沿都刷新超时定时器。
    // 超时后再统一解析, 可以避免半帧数据被提前处理。
    Board_RestartRxTimeoutTimer(15);
}

// 接收超时回调: 一段时间没有新边沿, 认为当前帧结束。
```



```
void RX_OnRxTimeout(void)
{
    if (rx_pulse_count == 0) {
        rx_last_edge_us = 0;
        return;
    }

    // 先把完整脉冲序列交给协议解析层, 解析通过后再执行业务动作。
    RX_DecodeFrame(rx_pulses, rx_pulse_count);

    // 清空接收状态, 准备下一帧。
    rx_pulse_count = 0;
    rx_last_edge_us = 0;
}
```

8.2 MCU 处理建议

项目	建议
输入脚配置	数据输出脚按 MCU 电平标准配置, 避免外部强上拉或强下拉影响波形。
休眠控制	由 MCU 明确控制工作和休眠状态, 避免状态不确定。
计时精度	脉宽计时建议使用微秒级定时器。
重复帧	遥控器长按可能连续发帧, 业务层应处理重复帧。
抗误触发	地址码、键值和脉宽范围均通过后再执行控制动作。

9 电源设计建议

微-R6 工作电流典型 0.18mA, 休眠电流典型 0.008mA。电源端应靠近模块放置 0.1uF 去耦电容; 整机有长线、继电器、电机或开关电源时, 可结合系统要求增加 TVS、防反接和输入滤波。

项目	建议
去耦电容	VCC/V 与 GND/G 之间靠近模块放置 0.1uF 陶瓷电容。
储能电容	根据整机供电情况增加 1-4.7uF 电容。
休眠控制	低功耗产品应明确控制休眠脚状态, 并验证唤醒后接收稳定性。
电源噪声	若系统有 DC-DC、电机或继电器, 应检查数据输出误触发和接收距离变化。



10 EMC 与抗干扰建议

- 模块应远离开关电源电感、MOSFET 开关节点、继电器触点、电机线和大电流回路。
- 数据输出线不宜过长; 若必须跨板连接, 应靠近 MCU 端增加必要的串阻或滤波设计。
- 休眠控制线应避免与高噪声线束并行长距离走线, 必要时可在 MCU 端增加滤波或默认电平约束。
- 整机有浪涌、ESD 或 EFT 要求时, 应在电源入口、外部按键线和长线接口处设置防护。
- 接收端软件应对异常短脉冲、过长脉冲和不完整帧做过滤。
- 批量生产前应在最终外壳、线束、电源和安装方向下做接收稳定性验证。

11 PCB 布线建议

- VCC/V 与 GND/G 去耦电容放在模块电源脚附近, 回路面积尽量小。
- 数据输出线远离天线端、DC-DC 开关节点、晶振、高速时钟和继电器驱动线。
- 休眠控制线可由 MCU GPIO 直接控制, 走线应短而清晰, 避免靠近高压或大电流节点。
- ANT/A 走线应短、直, 避免长距离绕线。
- 模块和天线附近的铜皮、金属结构会影响接收效果, 应结合尺寸图和结构要求预留净空。

12 天线与结构建议

天线对接收距离和稳定性影响较大。不接天线或天线不匹配会明显影响接收效果; 实际效果还与 PCB、外壳、结构、安装位置和现场干扰有关。

项目	建议
频点匹配	315MHz 与 433MHz 天线不能简单混用。
安装位置	天线区域尽量靠近外壳非金属区域。
周边结构	远离金属件、电池、屏蔽罩、线束和大面积铜皮。
验证方法	用实际发射端在目标距离、方向和安装环境下测试。

13 回流焊与生产建议

- 模块支持贴片安装, 工艺参数应按 PCB、焊膏和整机生产条件设定。
- 回流焊前检查模块方向、焊盘锡量和定位偏移。
- 波峰焊或手工焊时应避免长时间高温加热同一焊盘。

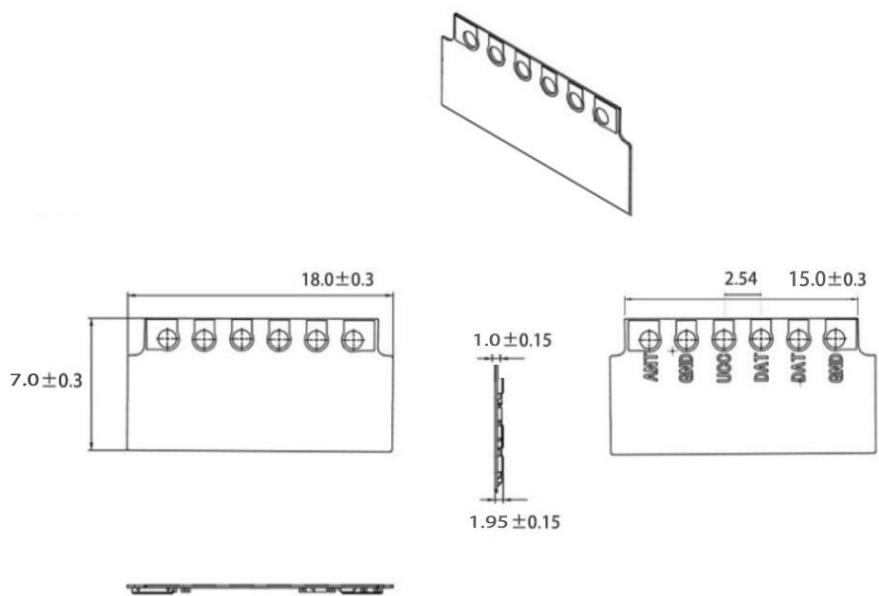


- 焊后检查数据输出、休眠控制、电源、地和天线焊点是否连锡、虚焊或偏移。
- 生产测试应包含上电电流、休眠电流、休眠控制、数据输出波形、频点匹配和实际遥控触发测试。

14 调试流程建议

步骤	检查项	说明
1	电源电压	确认供电在允许范围内, 纹波不过大。
2	工作电流	模块工作状态下检查电流是否接近典型值。
3	休眠电流	休眠控制脚拉高时检查待机电流。
4	数据输出波形	发射端按键时, 数据输出脚应有可识别脉冲输出。
5	编码解析	MCU 按目标协议解析地址码和键值。
6	天线效果	对比不同天线、外壳方向和安装位置。
7	干扰测试	在电机、继电器、开关电源动作时观察误触发情况。

15 尺寸



单位: mm



附录 A 天线、封装、开发工具、服务、购买

天线非常重要, 推荐TT02, 不接天线或天线不当会严重影响效果, 实际效果还和PCB、外壳、结构等有关, 建议多购买几种天线方便实测

推荐使用立创 EDA (www.lceda.cn)

直接搜索“蜂鸟无线”或“产品型号”即可找到

