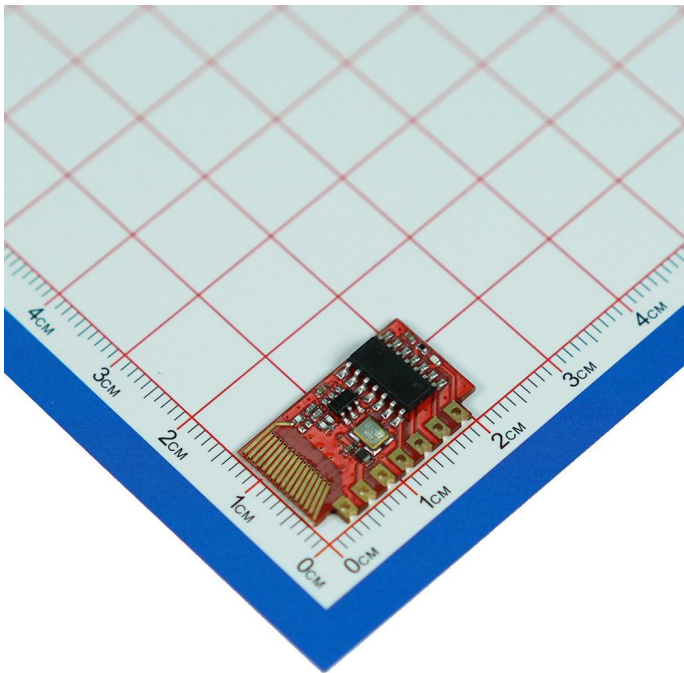




灵-R1A 接收模块规格书

Ling-R1A Remote Receiver Module Specification



项目	内容
产品型号	灵-R1A / Ling-R1A
文档版本	V2.1
发布日期	2026-07-04
品牌	FNRF / 蜂鸟无线



目录

灵-R1A 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 引脚与用法	5
6 订购型号与工作模式	7
7 输出与键值对应	7
8 串口输出格式	7
9 串口解析例程框架	8
10 学习、清码与急停	10
11 连接与使用流程	11
12 电源设计建议	12
13 EMC 与抗干扰建议	12
14 PCB 布线建议	12
15 天线与结构建议	13
16 回流焊与生产建议	13
17 调试流程建议	13
18 尺寸	14
19 资料与技术支持	14
附录 A 封装、开发工具	15



1 产品概述

灵-R1A 是一款 ASK/OOK 遥控接收模块, 工作频点为 433.92MHz。模块自带 PCB 天线, 内置解码逻辑, 可自动识别 1527 等常用编码方式, 并可学习常见射频遥控器。

模块提供 4 路开关量输出或 1 路串口输出模式。包括 M1 翻转款、M3 点动款、M4 互锁款和 M5N 串口款等。默认常用模式为 M3 点动款。

灵-R1A 可与灵-T1PRO、灵-T3A、灵-T3MAX 等发射模块组成遥控控制系统。实际接收距离和稳定性与发射端、天线结构、供电、电路板、外壳、安装位置和现场同频干扰有关, 整机应用时应按最终结构验证。

2 产品特点

- 433.92MHz ASK/OOK 遥控接收模块。
- 自带 PCB 天线, 适合小型遥控接收端集成。
- 支持 1527 等常用编码方式自动识别。
- 宽范围解码, 自动适应常规脉宽。
- 支持 M1 翻转、M3 点动、M4 互锁、M5N 串口输出等模式, 按订购型号区分。
- 可学习 80 个遥控, 学习信息掉电保存。
- 板载 LED 状态指示。
- 支持一键紧急关闭输出功能。
- 小体积封装, 支持贴片和插件兼容安装。
- 串口款固定 9600bps, ASCII 明文输出解码结果。

3 典型应用

- 灯控开关
- 无线门铃
- 智能家居
- 卷闸门与电机控制
- 遥控玩具
- 防盗报警
- 小家电遥控接收
- MCU 地址码和键值识别

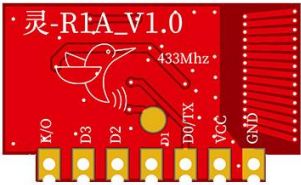


4 技术参数

参数	最小值	典型值	最大值	单位	条件/说明
工作频率	-	433.92	-	MHz	ASK/OOK 遥控接收。
工作电压	2.6	3.0	5.5	V	VCC 与 GND 之间。
工作电流	-	4.7	-	mA	接收工作状态。
调制方式	-	ASK/OOK	-	-	遥控接收。
接收灵敏度	-	-110	-	dBm	典型值。
数据接口	-	开关量/串口	-	-	按订购型号指定。
串口速率	-	9600	-	bps	M5N 串口款, ASCII 明文输出。
可学习遥控数量	-	80	-	个	掉电保存。
输出路数	-	4	-	路	D0-D3 开关量输出。
IO 驱动能力	-	-	15	mA	D0-D3 输出。
外形尺寸	-	22.35 x 11.30 x 2.50	-	mm	以尺寸图为准。
工作温度	-30	-	80	摄氏度	应用需按整机环境验证。



5 引脚与用法



背面视图

引脚名称	类型	功能	说明
K/O	输入	接对码按键	双击进入对码模式，遥控器按任意键可对码。
D0-D3	输出	4 路开关量输出	M1/M3/M4 模式下为开关量输出；D0 在 M5N 模式下为串口输出。单路 IO 驱动能力最大 15mA。
VCC	电源	电源正	2.6-5.5V，典型 3.0V。
GND	电源	电源负	与主控或外部电路共地。



用法:

方案一



方案二



方案三





6 订购型号与工作模式

序号	型号	名称	输出行为
1	灵-R1A-M1_433/315	翻转款	每次按下遥控, 对应输出状态翻转一次。
2	灵-R1A-M3_433/315	点动款	按键按下时对应输出高电平, 松开后恢复。默认款。
3	灵-R1A-M4_433/315	互锁款	D0-D3 每次只有一路为高, 其他为低。
4	灵-R1A-M5_433/315	串口款	D0 固定 9600bps 串口输出解码数据。
5	灵-R1A-M1N_433/315	翻转款免对码	无需学习对码, 按翻转逻辑输出。
6	灵-R1A-M3N_433/315	点动款免对码	无需学习对码, 按点动逻辑输出。
7	灵-R1A-M4N_433/315	互锁款免对码	无需学习对码, 按互锁逻辑输出。
8	灵-R1A-M5N_433/315	串口款免对码	无需学习对码, D0 输出串口解码数据。

7 输出与键值对应

发射端使用灵-T3、灵-T3A、灵-T3PRO 等产品时, 常见按键与灵-R1A 输出对应关系如下。若需更多按键, 可选择 M5N 串口模式配合灵-T3MAX, 最多支持 25 个按键。

接收输出	键值	常见发射端按键
D0	1	K3
D1	2	K2
D2	4	K1
D3	8	K0

8 串口输出格式

M5/M5N 串口款从 D0 脚输出解码结果。串口固定 9600bps, 输出为 ASCII 明文字符串, 不是 HEX 原始二进制格式。



8.1 输出帧说明

项目	示例	说明
输出示例	LC:DAB10893\r\n	ASCII 字符串形式。
固定帧头	LC:	用于识别一帧开始。
地址编码	DAB10	对应 1527 的 20bit 地址编码。
按键值	8	对应按键值。
校验	93	$DA + B1 + 08 = 0x193$, 取低 8 位为 0x93。
结束符	\r\n	回车换行符, 不可见字符。

8.2 串口接收边界

项目	参数/说明
串口波特率	9600bps, 固定。
数据格式	ASCII 明文输出, 不是 HEX 直接输出。
有效数据	固定帧头后的地址、键值和校验。
应用建议	MCU 应检查帧头、字符合法性、校验和重复帧逻辑, 再执行控制动作。

9 串口解析例程框架

串口款输出为 ASCII 字符串。MCU 解析时应先判断帧头 **LC:**, 再把后续 8 个十六进制字符转为 4 个字节, 其中前 3 个字节为地址/键值数据, 第 4 个字节为校验。校验通过后再执行控制动作。

9.1 ASCII 转 HEX 参考框架

```
// 灵-R1A M5/M5N 串口输出示例: LC:DAB10893\r\n
// 含义: LC: 为固定帧头; DA B1 08 为有效数据; 93 为校验和低 8 位。
// 注意: 本例为 MCU 解析框架, 实际产品还应加入接收超时、重复帧抑制和业务安全判断。
#include <stdint.h>
#include <stddef.h>
// 将 1 个 ASCII 十六进制字符转换为 0x0-0xF。
```



```
// 返回 0xFF 表示字符非法, 调用方应丢弃当前帧。
static uint8_t R1A_HexCharToNibble(uint8_t ch)
{
    if (ch >= '0' && ch <= '9') {
        return (uint8_t)(ch - '0');
    }
    if (ch >= 'A' && ch <= 'F') {
        return (uint8_t)(ch - 'A' + 10);
    }
    if (ch >= 'a' && ch <= 'f') {
        return (uint8_t)(ch - 'a' + 10);
    }
    return 0xFF;
}

// 解析一帧灵-R1A 串口数据。
// frame: 接收到的 ASCII 字符串缓冲区, 例如 "LC:DAB10893\r\n"。
// len: 当前缓冲区长度。
// out0-out2: 输出 3 个有效数据字节, 例如 DA B1 08。
// 返回 1 表示帧头、字符格式和校验均通过; 返回 0 表示数据无效。
uint8_t R1A_ParseAsciiFrame(const uint8_t *frame, size_t len,
                             uint8_t *out0, uint8_t *out1, uint8_t *out2)
{
    uint8_t n[8];
    uint8_t data0, data1, data2, checksum;
    // 最短需要包含 "LC:" + 8 个十六进制字符。
    if (len < 11) {
        return 0;
    }
    // 检查固定帧头。任一字符不匹配, 都应判定为无效帧。
    if (frame[0] != 'L' || frame[1] != 'C' || frame[2] != ':') {
        return 0;
    }
    // 将 DAB10893 这 8 个 ASCII 十六进制字符转换成 8 个半字节。
    for (size_t i = 0; i < 8; i++) {
        n[i] = R1A_HexCharToNibble(frame[3 + i]);
        if (n[i] == 0xFF) {
            return 0;
        }
    }
```



```
}  
// 两个半字节组合成一个字节: DA B1 08 93。  
data0 = (uint8_t)((n[0] << 4) | n[1]);  
data1 = (uint8_t)((n[2] << 4) | n[3]);  
data2 = (uint8_t)((n[4] << 4) | n[5]);  
checksum = (uint8_t)((n[6] << 4) | n[7]);  
// 校验规则: 前三个字节累加, 取低 8 位。  
if ((uint8_t)(data0 + data1 + data2) != checksum) {  
    return 0;  
}  
*out0 = data0;  
*out1 = data1;  
*out2 = data2;  
return 1;  
}
```

9.2 控制类应用建议

项目	建议
重复帧	遥控器长按可能连续输出多帧, MCU 应按业务逻辑做重复帧抑制。
动作确认	电机、门锁、报警等控制动作应增加状态确认或超时保护。
安全边界	接收端不宜只凭一次未校验数据立即执行高风险动作。
N 款免对码	免对码适合快速兼容应用, 但实际产品仍需评估误触发和同频遥控环境。

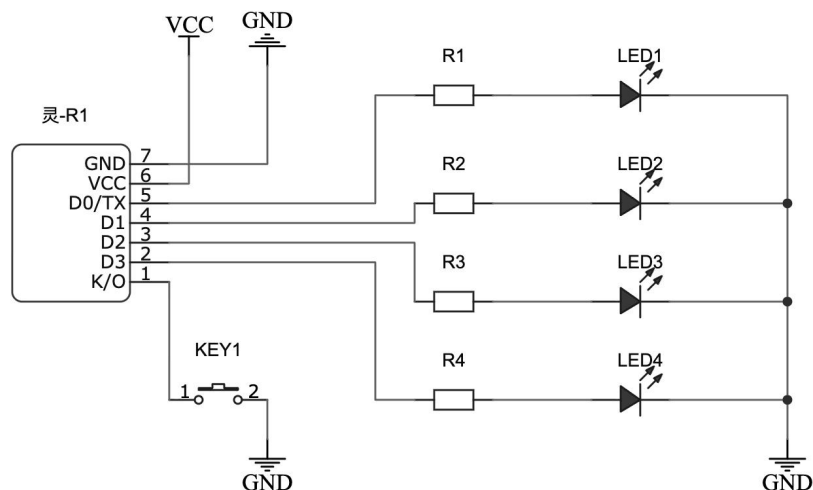
10 学习、清码与急停

功能	操作方法	说明
学习对码	上电后快速双击 KEY1, 工作指示灯闪烁进入对码模式; 触发灵-T1L、灵-T3A、灵-T3MAX 或遥控器任意键。	工作指示灯约亮 1 秒表示对码成功; N 款无需对码。
清除所有遥控	上电后长按 KEY1 8 秒以上, 松开按键。	已学习遥控全部清空。
紧急停止	模块工作期间短按对码键。	D0-D3 全部变低电平输出, 适合卷闸门电机等需要急停的场景。
学习容量	-	可学习 80 个遥控, 掉电保存。



11 连接与使用流程

11.1 接线



连接图

模块引脚	连接对象	说明
VCC	电源正极	2.6-5.5V, 典型 3.0V。
GND	电源负极	与主控或外部电路共地。
D0-D3	MCU 输入或负载驱动输入	开关量模式下输出对应按键状态; 驱动负载时需增加外部驱动电路。
K/O, K1	对码按键	用于学习对码、清码和急停。

11.2 使用流程

步骤	操作	说明
1	选择工作模式	按输出逻辑选择 M1、M3、M4 或 M5N。
2	接入电源	按 2.6-5.5V 范围供电, 典型 3.0V。
3	接入输出端	D0-D3 接 MCU 输入或外部驱动电路。



步骤	操作	说明
4	学习遥控	非 N 款按学习流程对码; N 款无需对码。
5	验证输出	按键触发后检查输出逻辑、串口帧或下游动作。
6	整机实测	在最终外壳、供电和安装环境下验证接收距离和稳定性。

12 电源设计建议

- VCC 与 GND 走线应尽量短、宽, 并靠近模块放置 0.1uF 去耦电容和合适的储能电容。
- 模块电源不宜与继电器、电机、蜂鸣器、DC-DC 电感等高噪声负载共用细长供电路径。
- 输出端驱动继电器、电机或大电流负载时, 应使用外部驱动器件, 并增加续流、浪涌和地回流设计。
- 调试偶发丢码或误动作时, 应使用示波器在模块 VCC/GND 引脚处观察负载动作瞬间的电压变化。
- 电池供电应用应评估低电压、冷启动、温度和放电后期的接收稳定性。

13 EMC 与抗干扰建议

- 模块和 PCB 天线区域应远离继电器触点、电机线、开关电源、电感、长线束和高频时钟区域。
- 继电器、电机、螺线管等负载动作时, 应测试接收成功率和输出误触发情况。
- 外壳、电池、金属面板、线束和安装位置会影响接收效果, 应在最终结构中复测。
- 产测时不宜仅测试极限距离, 应在固定距离、固定方向、固定供电状态下重复触发验证成功率。
- 如整机需要 FCC/CE/SRRC/EMC 等认证, 应以整机测试和正式报告为准。

14 PCB 布线建议

- 模块电源和地线应保证低阻抗回路, 避免负载动作引起 VCC 抖动。
- PCB 天线正下方和附近区域应保持净空, 不铺铜, 不放金属结构和高噪声器件。
- D0-D3 与对码按键线不宜过长, 长线输入/输出应结合 ESD、防抖和抗干扰测试评估。
- 模块附近应避免强开关电流回路、继电器触点走线和电机线束。
- RF 区域、数字输出区域和负载驱动区域应分区布置, 降低相互耦合。



15 天线与结构建议

- 灵-R1A 自带 PCB 天线, 天线区域应保持净空。
- PCB 天线附近不应放置铜皮、电池、屏蔽罩、金属外壳、螺丝柱或长线束。
- 模块安装方向、外壳材料和整机内部金属件会影响接收距离, 应按最终结构验证。
- 同一产品应固定模块位置和方向, 避免生产批次中天线环境变化导致接收性能不一致。

16 回流焊与生产建议

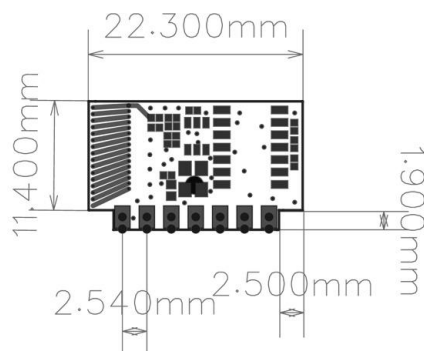
- 模块支持贴片和插件兼容封装, 焊接工艺应按 PCB、焊膏、器件耐温和工厂工艺确认。
- 波峰焊或回流焊前应确认模块方向、焊盘开窗和天线净空。
- 返修时应避免局部长时间高温加热, 防止模块和 PCB 天线区域受损。
- 批量生产前应做小批量焊接验证, 检查虚焊、连锡、偏移和输出端焊点状态。
- 生产测试建议记录频点、工作模式、学习状态、遥控器型号、供电电压、测试距离和输出结果。

17 调试流程建议

阶段	检查项目	说明
裸板验证	频点、编码、输出	使用开发助手或发射端确认遥控信号和输出是否匹配。
模式验证	M1/M3/M4/M5N	按订购型号验证翻转、点动、互锁或串口输出逻辑。
对码验证	学习、清码、急停	检查学习容量、掉电保存和急停输出。
供电验证	VCC 波形	观察负载动作和接收瞬间的电源稳定性。
外壳验证	距离和成功率	装壳后复测接收距离、方向性和成功率。
批量验证	固定距离重复测试	用固定工装、固定方向和固定电源状态测试通过率。



18 尺寸



19 资料与技术支持

技术资料、设置软件和测试工具可通过蜂鸟无线官网获取:

WWW.FNRF.CC

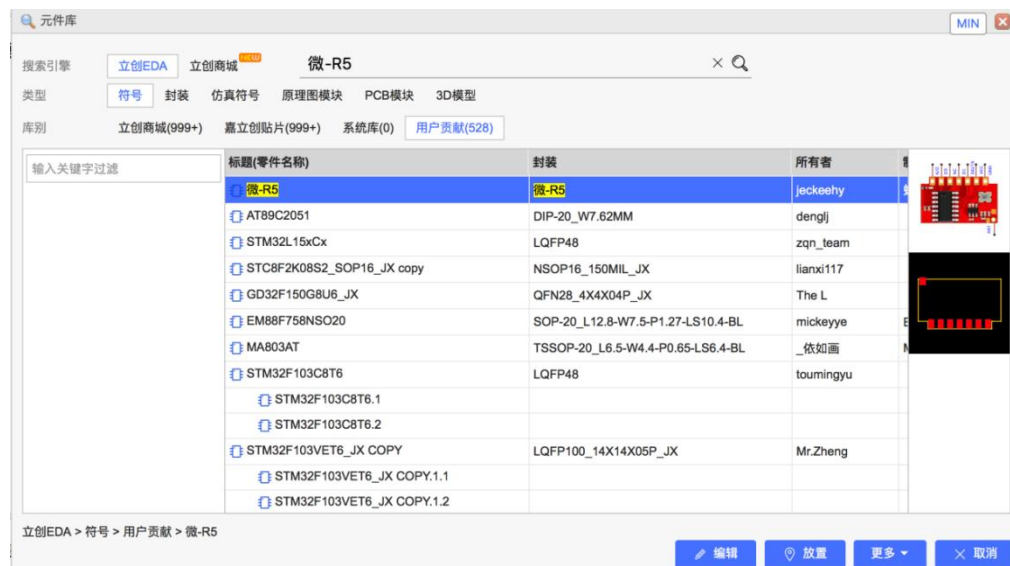
选型和项目导入时, 应确认工作模式、频点、供电方式、输出逻辑、发射端型号、外壳结构、认证需求和批量测试方法。



附录 A 封装、开发工具

封装

推荐使用高效的国产 PCB 设计工具: 立创 EDA (www.lceda.cn) 直接搜索“蜂鸟无线”或“产品型号”即可找到



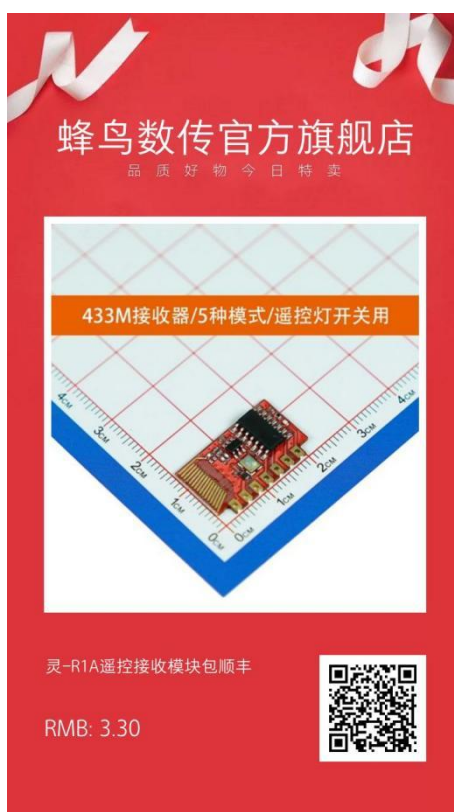


开发工具

开发助手	信号助手
	
不同点: 1、测数据值 2、有编码类型要求(1527 等)	不同点: 1、测信号强度 2、不限编码(ASK/OOK 调制)
用途: 1、显示遥控器/发射模块的地址码和按键值 2、显示遥控频率、脉宽、编码类型 3、遥控产品批量测试	用途: 1、显示遥控器/发射模块信号强弱(相对值) 2、对比天线好坏 3、遥控产品批量测试
供电: TYPE-C	供电: TYPE-C
 微信扫码购买	 微信扫码购买



微信扫码技术咨询+获取开发资料



微信扫码购买送工具