

KIR01 万能红外学习控制器使用说明

文件状态：	文件标识：	对外公开
☐ 草稿	当前版本：	V1.2
☒ 正式发布	作 者：	云尚电子
☐ 正在修改	完成日期：	2023-02-18



目录

KIR01 万能红外学习控制器使用说明 1

1. 概述 3

 1.1 简介 3

 1.2 基本参数 3

 1.3 接口描述 3

 1.4 产品应用场景 5

 1.5 控制器的功能简单说明 5

 1.6 上位机简介 5

2 应用场景 6

 2.1 简述 6

 2.2 场景一：直接电脑端上位机调试测试 6

 2.3 场景二：控制器+RS485 电路用于工业领域 7

 2.4 场景三：控制器+单片机实现复杂的功能 7

3 上位机使用说明 8

 3.1 键值学习和测试 8

 3.2 键值学习状态查询 8

 3.3 键值改名 8

 3.4 导出导入红外学习码库文件 9

4 简码通信协议 9

 4.1 通信协议格式 9

 4.2 通信协议，命令详述 10

5 modbus 通信协议 10

 5.1 通信协议简述 10

6 尺寸图 11

7 附件 12

8 版本历史 12

1. 概述

1.1 简介

KIRO1 控制器是一款万能型红外遥控学习控制器，支持普通 NEC 编码的红外遥控器的学习，也支持空调的长码编码的学习，之后根据指令重新将学习的键码发送出去，以实现遥控器的复制或者集成到智能设备中。具有多种控制方式，串口简码或者 485+MODBUS，并配合提供的上位机软件，能轻松的实现产品的开发，大大的降低了新产品的开发难度。

1.2 基本参数

供电方式	DC 7-24V 100mA（静态电流<10mA,发射电流 600mA）
遥控距离	单向发射头 8 米距离,4 向发射头半径 6 米。另外可选配更远距离的发射头，最远可达 20 米
通信方式	485 通信，MODBUS 协议+简码通信协议,8 位数据,1 位停止位,无奇偶校验位
控制方式	485 的 MODBUS 控制、电脑端控制、板内扩展 TTL 串口控制
波特率	2400、4800、 9600(出厂默认) 、19200、115200
编码方式	支持 NEC、空调码、RC-5、RC-6、PWM、PPM 等等绝大部分红外遥控器
载波频率	38KHz
键值数量	可学习并储存 48 个键值

1.3 接口描述

控制器外部接口有电源接口，485 接口，发射头接口。另外还有拨码开关，工作指示灯，红外接收头。

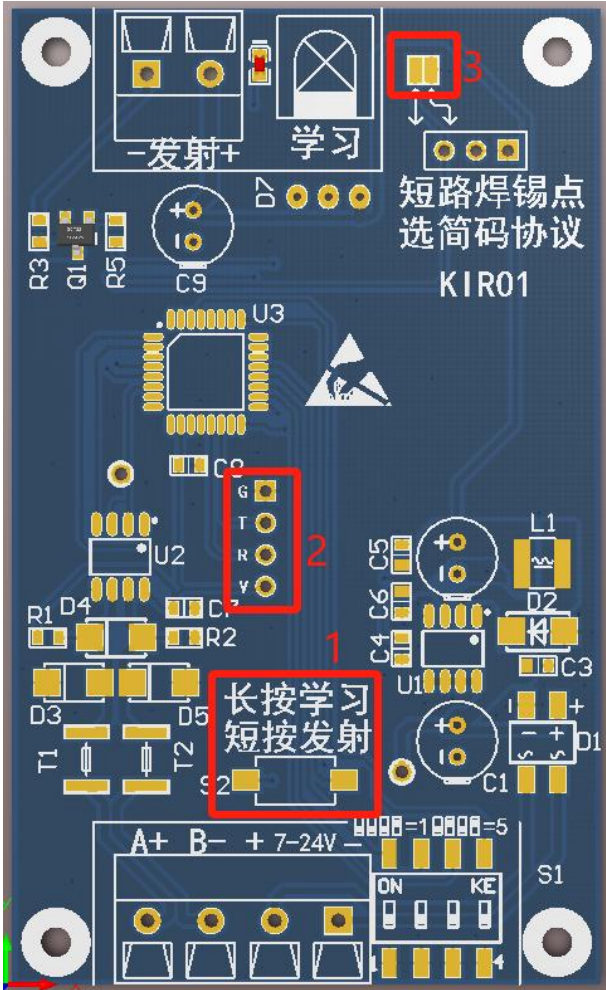


接口	说明
485 接口	用于 485 通信控制的接口，2 线。接线时需要注意线序，A 接 A，B 接 B，线序不对会造成无法通信。
电源接口	用于给设备供电的接口，2 个脚，可以不分正负极，直流或者交流，7-24V 电源(最高输入电压不得超过 26V)，电流不小于 400mA。
地址拨码	MODBUS 通信控制时用到的通信地址=拨码地址+软件设置地址。软件设置地址默认是 1。
红外接收头	进行遥控器学习时，需要用遥控器对准接收头，距离 1 米远按遥控器，进行遥控按键学习。
工作指示灯	控制器通电的时候闪一下 学习遥控器的时候常亮，学习成功就熄灭 发射红外信号会闪一下
发射头接口	需要注意接线的正负极，线序接错了会造成无法发射红外信号； 如果连接发射头，红线接+，黑线接-； 如果连接发射管，长脚接+，短脚接-； 红外光信号人眼无法看到，但是手机摄像头可以看到微弱的紫色光，所以有时候需要确认发射头是否能发射红外信号，可以用手机摄像头观察。

因为 MODBUS 通信协议开发稍微有点难度，所有有的用户可能不方便使用 MODBUS 协议进行控制，这时候可以选择简码通信协议进行控制。拆开控制器，板子上面有个焊锡短路点，默认没有焊接上，如果用焊锡焊接，控制器通电就会运行简码通信协议，协议的具体说明请参考说明书的【简码通信协议】部分。

特定情况下，拆开控制器可以进行一些额外操作。

1 测试按键	右图 1：长按测试按钮 4 秒松开，控制器进入按键 0 的红外学习状态，此时按遥控器可进行红外按键学习。 短按测试按钮就是发射按键 0 的红外信号。此按钮是方便没有电脑连接的情况下快速测试用。
2 内部扩展串口 TTL	控制器外接的有一个 485 接口，板子上还预留了一个串口 TTL 接口，5V 电平。 扩展接口和 485 接口是 2 个不同的通道，在特定的情况下部分用户可能用得上这个接口，比如在这里接蓝牙透传、WiFi 透传、4G 透传、Lora 透传、单片机…… 但请注意：485 接口和内部扩展接口允许 2 路都接设备进行通信，但不允许在同一时刻 2 路都有数据传输，会造成控制器放弃此次的通信数据包。
3 通信协议选择短路点	控制器默认没焊接短路点，运行的是 MODBUS-RTU 通信协议，如果焊接了这个短路点，控制器就运行简码通信协议。



1.4 产品应用场景

1.智能家居产品	如：多功能遥控器魔方、万能遥控器 等等 此应用场景推荐使用单片机+控制器，采用串口简码通信协议
2.工业控制产品	如：空调控制器，温湿度机控制器，恒温恒湿控制器
3.万能遥控器	如：将多款设备集合到一个遥控器里面
4.工控管理	如：档案管理室，远程的 ERP 控制设备；学校宿舍统一管理空调

1.5 控制器的功能简单说明

连接上位机	通过串口链接上位机软件，可以实现参数配置、键值学习、键值测试、通信协议调试、红外码库的导出、导入。方便批量的配置控制器工作参数和导入红外学习码库数据 控制器和上位机软件都支持 2 种通信协议
通信协议	通过串口控制设备的时候串口数据类型支持 2 种通信协议，由短路点决定 简码通信协议：55 AA 03 01 03（发射键值 01）通信协议比较简单，带有校验位，简单可靠，适合单片机控制 MODBUS 协议：采用标准的工业控制 MODBUS-RTU 协议，方便直接将产品应用到工业领域，无需过多开发
学习遥控键值	进入红外学习后，将遥控器对准红外接收头，距离 1 米，轻点需要学习的按键即可
测试遥控键值	对应的键值学习成功之后，可以通过上位机或者按钮测试对应的键值，将红外发射头对准设备，然后测试发射
测试按键	控制器上面有个按键，长按按键进入键值 0 的学习状态，短按按键则发射键值 0(前提是键值 0 已经学习成功)
485 地址	这个是 MODBUS 协议下有效，通信地址=软件配置的地址偏移量+拨码地址+1（地址有效范围 1-250）

1.6 上位机简介

“红外遥控学习管理器”是一款 windows 系统 PC 端的一款软件，方便用户更快速的上手使用红外学习控制器而研发的。通过该软件可以非常方便的测试，和配置红外学习控制器，还能很方便的批量下载红外学习码库，以实现高效率研发、生产产品。



2 应用场景

2.1 简述

由于控制器支持的接口多，可以应用于多种场景，不同的场景使用的方式不同，下面根据对应的场景进行说明。

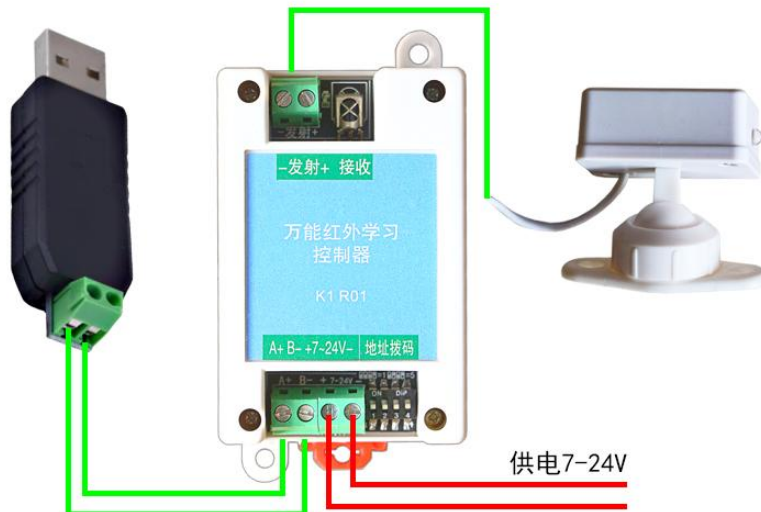
2.2 场景一：直接电脑端上位机调试测试

此应用场景通常用于控制器初期开发测试，需要自备一个 USB 转串口工具，推荐使用 CH340 相关工具，并进行接线。其中 CFG 引脚是用于选择通信协议方式的引脚，控制器支持 2 种通信协议，简码通信协议比较简单可靠，能通过单片机串口发送简单的指令就能有效控制；MODBUS 协议则是工业领域应用广泛的一种协议，但编程稍微复杂，可以根据需要选择合适自己的一套通信协议，并且需要修改上位机软件所处的协议状态。

CFG 低电平选择简码协议，高电平选择 MODBUS 协议。

红外学习上位机配置文件

使用USB转485工具进行连接，正确的连接好电线，如下图所示：



接线接好之后，将串口工具插上电脑 USB 口，并且安装好串口工具的驱动，选择对应的 COM 口和波特率，点击“打开串口”按钮，再点击“读取信息”按钮，如果能读取设备信息，则代表设备连接成功，不能读取则检查接线、驱动、参数相关是否有误。

机配置文件

在上位机软件界面，最底下有对应操作的串口指令首发具体数据内容

示例一：modbus 协议发送键值 0

TX:00 06 00 C8 00 01 C8 25

RX:05 06 00 C8 00 01 C8 70

上图中这一串码代表的是 MODBUS 协议下，电脑发送：

0x00 地址 00 设备（00 地址是广播地址）
 0x06 操作码 06，写保持寄存器
 0x00,0xC8 寄存器 200，对应的是发射红外信号键值 0
 0x00,0x01 数据内容 01
 0xC8,0x25 CRC-16-Modbus-Rtu 检验数据

之后控制器针对电脑发送：

0x05 地址 05 设备
 0x06 操作码 06，写保持寄存器
 0x00,0xC8 寄存器 200，对应的是发射红外信号键值 0
 0x00,0x01 数据内容 01
 0xC8,0x70 CRC-16-Modbus-Rtu 检验数据

示例二：简码协议发送键值 0

TX:0x55 0xAA 0x03 0x00 0x02 RX:0x55 0xAA 0x03 0x01 0x03

这一串码代表的是简码协议下，电脑发送：

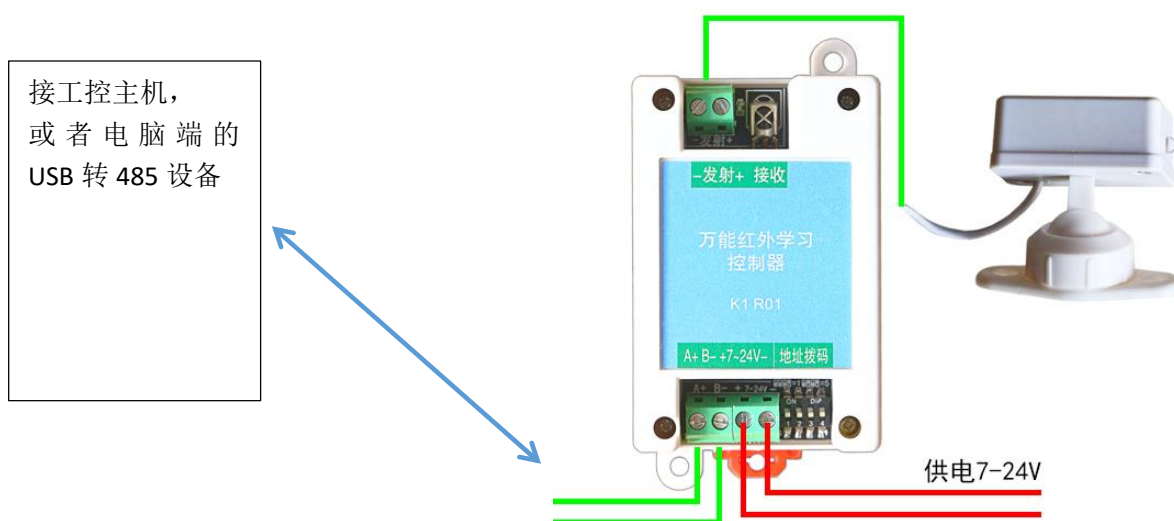
0x55,0xAA 固定的数据头
 0x03 操作码 03，发送红外键值
 0x00 键值 0
 0x02 累加和校验， $0x55+0xAA+0x03+0x00=0x102$ ，只保留一字节就是 0x02

之后控制器针对电脑发送：

0x55,0xAA 固定的数据头
 0x03 操作码 03，发送红外键值
 0x01 结果值：1 发送成功，0 发送失败
 0x02 累加和校验， $0x55+0xAA+0x03+0x01=0x103$ ，只保留一字节就是 0x03

2.3 场景二：控制器+RS485 电路用于工业领域

此场景常用于工控领域，比如远程控制空调、加湿机针对档案管理室、变电站、机房进行恒温恒湿控制。无需额外加单片机，不需要开发单片机代码，直接加电源电路、RS485 电路即可。



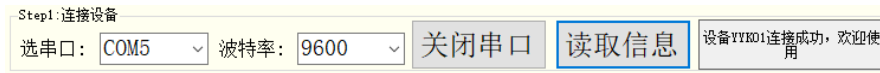
2.4 场景三：控制器+单片机实现复杂的功能

此应用场景常用于智能家居、工控领域，通过单片机挂载其他的传感器可以轻松实现复杂的功能。比如单片机挂载语音识别模块，再挂载控制器，则可实现语音控制空调，台灯，风扇……。

此模式下建议短接短路点，采用简码通信协议，开发简单。

3 上位机使用说明

使用 USB 转 485 工具进行连接，将 USB 插上电脑,打开资料包里面有驱动，安装完驱动之后打开上位机软件，选择对应的串口号，设置出厂默认的波特率 9600，点击打开按钮，之后点击读取信息，显示设备连接成功。



3.1 键值学习和测试

点击开始学习按钮，则控制器会进入红外学习的状态，此时指示灯亮，需要在 20 秒内将被学习的遥控器对准控制器的接收头，距离 1 米，轻点遥控器按键，学习成功之后控制器指示灯熄灭。之后将控制器的发射头对准被测的设备，点击测试键值，如果设备有相应的反应，则代表学习成功，若学习失败，请重试学习步骤，重新学习。



在上位机的最下方有串口通信数据收发信息显示（下面仅以简码协议做说明，Modbus 类似）

TX:55 AA 02 04 05

RX:55 AA 02 01 02

发:[0x55,0xAA 数据头]+[0x02 命令(进入红外学习)]+[0x04 键值 4]+[0x05 校验]

收:[0x55,0xAA 数据头]+[0x02 命令(进入红外学习)]+[0x01 操作结果有效]+[0x02 校验]

TX:55 AA 03 04 06

RX:55 AA 03 01 03

发:[0x55,0xAA 数据头]+[0x03 命令(发射红外信号)]+[0x04 键值 4]+[0x06 校验]

收:[0x55,0xAA 数据头]+[0x03 命令(发射红外信号)]+[0x01 操作结果有效]+[0x03 校验]

3.2 键值学习状态查询

点击“读取学习状态”按钮，可查询 48 个键值的学习状态，打钩的代表已经学习



在下方会有收发信息，55 AA 为数据头，01 是指令码，代表查询学习状态，后面 8 字节数据转换成 8 个 8 位，一共 64 位二进制数，代表：键值 63、键值 62、键值 61、键值 60……键值 2、键值 1、键值 0 共 64 个键值的学习状态。1 为学习成功，0 代表没有学习，因为 IR02 只有 48 个键，所以 8 字节数据只有低位的 6 个字节有效。

TX:55 AA 01 00

RX:55 AA 01 00 00 80 00 00 00 01 03 84

上图 8 个数据：00 00 80 00 00 00 01 03，取右边 6 字节 80 00 00 00 01 03 表示 48 键学习状态，转换成 2 进制是：1000 0000,0000 0000,0000 0000,0000 0000,0000 0001,0000 0011
代表键值 0、1、8、47 键值已经学习成功。

3.3 键值改名

为了方便使用红外控制器，上位机软件上面的键值名称可以被修改并保存

制热28度自动

☐ 已学习

开始学习

键值测试

制冷20度强力模式

☐ 已学习

开始学习

键值测试

除湿机关机

☐ 已学习

开始学习

键值测试

保存键值名称

1. 红外键值学习成功之后不会主动上报是

2. “键值n”这个可以修改并保存名称，根

3. 管理数据功能，可以将数据导出保存在

4. 修改波特率之后，需要重新以新的波特

3.4 导出导入红外学习码库文件

为了方便批量生产产品，免除反复的进行遥控学习，通过上位机的红外学习码库管理，可以轻松地批量下载码库文件。首先取一个控制器，将需要学习的键值全部学习并测试通过，之后点击“从设备导出所有的学习码库”按钮，等待导出完成，之后点击保存按钮，文件会以当前日期时间命名并且保存到软件所在的目录位置，之后可以自行修改文件名称，存档备用。然后可以更换成其他的控制器，再点击“导入学习码库到设备”按钮，进行其他控制器的快速配置。码库导入导出的时候可以将波特率设置最高，以加快速度，导完了之后改为低波特率。

step3:管理数据

从设备导出所有的学习码

保存学习码到本地文件

打开本地文件学习码

导入学习码到设备

4 简码通信协议

当焊接了短路点时，通信协议运行于简码协议。

4.1 通信协议格式

协定一套通信协议，为的是更可靠的将红外学习控制器与 MCU 控制器进行通信，协议码由:起始码(2 字节)+指令码(1 字节)+数据码(n 字节)+校验码(1 字节)组成。并且一条指令，需要在 100ms 内发送完成，超时未发送完的，视为无效指令。

起始码	0x55, 0xAA	
指令码	0x01	查询红外学习键值 TX: 0x55,0xAA,0x01,0x00 单片机发指令查询学习状态 RX: 0x55,0xAA,0x01,【0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,0xFF】,0xFD 控制器发送学习结果
	0x02	指定键值进入学习状态 TX: 0x55,0xAA,0x02,0x05,0x06 让键值 5 进入红外学习 RX: 0x55,0xAA,0x02,【0x01】,0x02 进入红外学习模式【1=成功/0=失败】 如果失败，可能对应的键值不存在
	0x03	指定键值发射红外信号，需要学习成功之后才能有效发送 TX: 0x55,0xAA,0x03,0x05,0x07 让键值 5 进入红外学习 RX: 0x55,0xAA,0x03,【0x01】,0x08 红外信号发射【1=成功/0=失败】 如果失败，可能对应的键值不存在或者没学习成功
	0x04	设置通信波特率 TX: 0x55,0xAA,0x04,0x02,0x05 修改波特率为 9600 RX: 0x55,0xAA,0x04,【0x01】,0x06 修改波特率成功【1=成功/0=失败】 如果失败，可能是不支持的波特率
数据码		不同的指令后面的数据码数量不同
校验码		校验码指的是从起始码开始，到最后一个数据码的 8 位累加和溢出值作为校验码，例如:0x55+0xAA+0x04+0x02+0x01 = 0x106，只取 8 位，就是 0x06

4.2 通信协议，命令详述

命令	说明	MCU/上位机 发送	控制器回码
1	查询红外学习结果	0x55,0xAA,0x01,0x00 无数据位	0x55,0xAA,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x13,0x13 8 字节数据位，转换成 64 位的二进制数，就代表着 64 个键值的学习状态（IR02 有 48 个键），1 已学习，0 未学习。从最右边的数据 0 位开始，至最边边的数据 63 位，分别代表键值 0……键值 63。0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x13 表示键值 0、1、4 已学习，其他的键值未学习
2	指定对应的键值开始学习红外	0x55,0xAA,0x02,0x05,0x06 1 字节的数据位，取值范围 0……47	0x55,0xAA,0x02,0x01,0x02 1 字节的数据位，若是 0 代表操作无效（可能没有这个键值），1 代表操作有效（不代表学习成功了）。进入学习状态需要在 20 秒内进行键值学习，超时视为学习失败。
3	发射键值	0x55,0xAA,0x03,0x05,0x07 1 字节数据位，数据取值范围 0……47	0x55,0xAA,0x03,0x01,0x03 1 字节的数据位，若是 0 代表操作无效（可能没有这个键值或没有学习），1 代表操作有效
4	设置通信波特率	0x55,0xAA,0x04,0x02,0x05 1 字节数据位 0:2400 1:4800 2:9600(出厂默认) 3:19200 4:115200	0x55,0xAA,0x04,0x01,0x04 1 字节数据位，若是 1 表示成功，若是 0 表示修改失败 修改波特率之后，断电重启都是以新的波特率在工作

5 modbus 通信协议

控制器默认没焊接短路线，通信协议运行于 Modbus-RTU 协议。

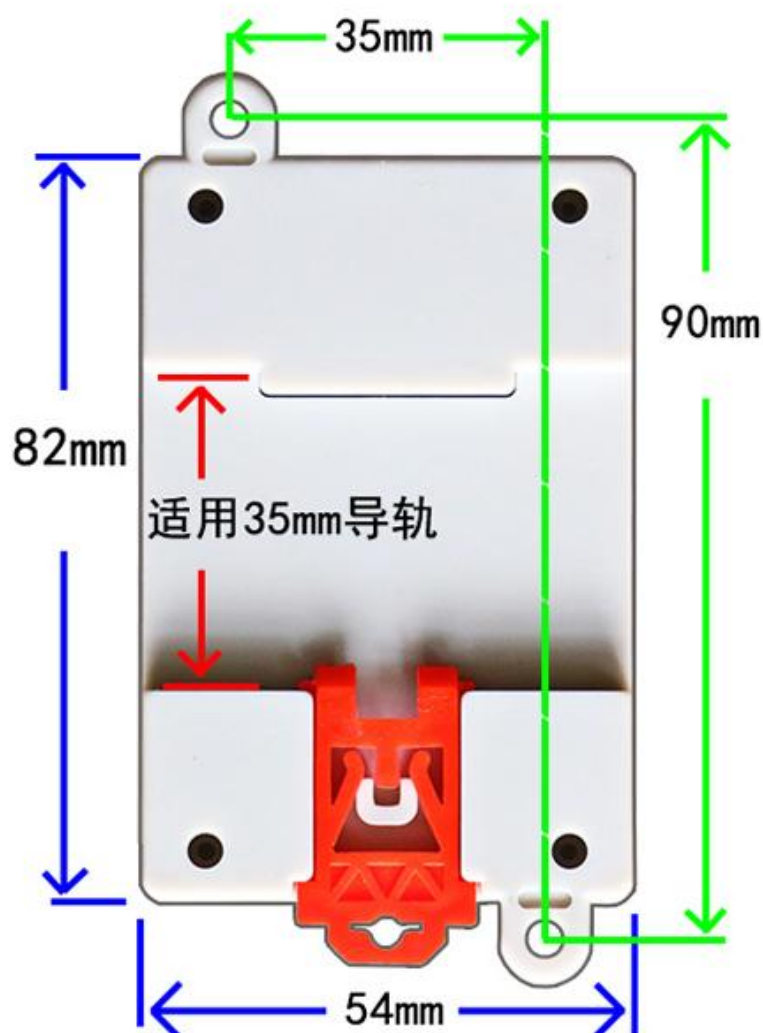
5.1 通信协议简述

采用标准 MODBUS-RTU 通信协议，能很方便的对接市面上大部分的工业设备，或者对接组态软件，甚至对接 PLC 设备。

寄存器地址	PLC 地址	操作码	说明
0	40001	03/06	485 通信地址偏移量
1	40002	03/06	485 通信波特率
100	40101	06	键值 0 进入红外学习
101	40102	06	键值 2 进入红外学习
……			
147	40248	06	键值 47 进入红外学习
200	40201	06	键值 0 发射红外信号
201	40202	06	键值 2 发射红外信号
……			
247	40248	06	键值 47 发射红外信号
300	40301	03	键值 15……0 学习状态参数

301	40302	03	键值 31……16 学习状态参数
302	40303	03	键值 47……32 学习状态参数
303	40304	03	键值 63……48 学习状态参数

6 尺寸图



控制器采用导轨或螺钉固定方式
发射头采用螺钉固定

7 附件

7.1 简码协议的校验计算方法，累加和校验

```
u8 UartCrc(u8 *buf, u8 lenth)
{
    u8 temp = 0;
    u8 i;
    for (i = 0; i < lenth; i++)
    {
        temp += buf[i];
    }
    return temp;
}

//-----
Uart1_Tx_Dat[0] = 0x55;
Uart1_Tx_Dat[1] = 0xAA;
Uart1_Tx_Dat[2] = 0x01;
Uart1_Tx_Dat[3] = UartCrc(Uart1_Tx_Dat, 3);
Uart_Sendstr_lenth(Uart1_Tx_Dat,4);
```

8 版本历史

V1.0	2022.07.09	修订
V1.1	2022.08.03	完善部分细节描述 更换部分图片 新增板载短路点焊接描述 新增简码通信协议的描述
V1.2	2023.02.18	新增控制器板载的串口 TTL 描述